ETSI GS CIM 013 V2.1.1 (2024-04)



Context Information Management (CIM); NGSI-LD Test Purposes Descriptions

The present document has been produced and approved by the cross-cutting Context Information Management (CIM) ETSI Industry Specification Group (ISG) and represents the views of those members who participated in this ISG.

It does not necessarily represent the views of the entire ETSI membership.

Reference RGS/CIM-0013v211 Keywords API, IoT, NGSI-LD, testing

ETSI

650 Route des Lucioles F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - APE 7112B Association à but non lucratif enregistrée à la Sous-Préfecture de Grasse (06) N° w061004871

Important notice

The present document can be downloaded from: https://www.etsi.org/standards-search

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be modified without the prior written authorization of ETSI. In case of any existing or perceived difference in contents between such versions and/or in print, the prevailing version of an ETSI deliverable is the one made publicly available in PDF format at www.etsi.org/deliver.

Users of the present document should be aware that the document may be subject to revision or change of status.

Information on the current status of this and other ETSI documents is available at https://portal.etsi.org/TB/ETSIDeliverableStatus.aspx

If you find errors in the present document, please send your comment to one of the following services: https://portal.etsi.org/People/CommiteeSupportStaff.aspx

If you find a security vulnerability in the present document, please report it through our Coordinated Vulnerability Disclosure Program:

https://www.etsi.org/standards/coordinated-vulnerability-disclosure

Notice of disclaimer & limitation of liability

The information provided in the present deliverable is directed solely to professionals who have the appropriate degree of experience to understand and interpret its content in accordance with generally accepted engineering or other professional standard and applicable regulations.

No recommendation as to products and services or vendors is made or should be implied.

No representation or warranty is made that this deliverable is technically accurate or sufficient or conforms to any law and/or governmental rule and/or regulation and further, no representation or warranty is made of merchantability or fitness for any particular purpose or against infringement of intellectual property rights.

In no event shall ETSI be held liable for loss of profits or any other incidental or consequential damages.

Any software contained in this deliverable is provided "AS IS" with no warranties, express or implied, including but not limited to, the warranties of merchantability, fitness for a particular purpose and non-infringement of intellectual property rights and ETSI shall not be held liable in any event for any damages whatsoever (including, without limitation, damages for loss of profits, business interruption, loss of information, or any other pecuniary loss) arising out of or related to the use of or inability to use the software.

Copyright Notification

No part may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm except as authorized by written permission of ETSI.

The content of the PDF version shall not be modified without the written authorization of ETSI.

The copyright and the foregoing restriction extend to reproduction in all media.

© ETSI 2024. All rights reserved.

Contents

Intelle	ectual Property Rights	5
Forew	ord	5
Modal	l verbs terminology	5
1	Scope	6
2	References	6
2.1	Normative references	
2.2	Informative references	6
3	Definition of terms, symbols and abbreviations	7
3.1	Terms	7
3.2	Symbols	
3.3	Abbreviations	7
4	Test Purposes Descriptions	8
4.1	jsonldContext	8
4.1.1	Consumption	
4.1.1.1		
4.1.1.2		
4.1.2	Provision	
4.1.2.1		
4.1.2.2		
4.2 4.2.1	CommonBehaviours	
4.2.1 4.2.1.1	•	
4.2.1.1 4.2.1.2		
4.2.1.3	·	
4.2.1.4		
4.2.1.5		
4.3	ContextSource	
4.3.1	Registration	
4.3.1.1	· · · · · · · · · · · · · · · · · · ·	
4.3.1.2	<u> </u>	
4.3.1.3	RegisterContextSource	56
4.3.2	RegistrationSubscription	65
4.3.2.1	β · · · · · · · · · · · · · · · · · · ·	
4.3.2.2	7 · · · · · · · · · · · · · · · · · · ·	
4.3.2.3		78
4.3.2.4		
4.3.2.5		
4.3.2.6		
4.3.3	Discovery	
4.3.3.1 4.3.3.2	e	
4.3.3.2 4.4	QueryContextSourceRegistrations ContextInformation	
4.4 4.4.1	Consumption	
4.4.1.1	•	
4.4.1.1	1	
4.4.1.1		
4.4.1.2		
4.4.1.2	· · · · · · · · · · · · · · · · · · ·	
4.4.1.2	J J1	
4.4.1.2		
4.4.1.2	• • • •	
4.4.1.2		
4.4.1.2	RetrieveAvailableEntityTypeInformation	140
4.4.1.3	Entity	141

4.4.1.3.1	QueryEn	tities	141
4.4.1.3.2	Retrievel	Entity	158
4.4.2	Subscription		167
4.4.2.1	UpdateSubso	cription	167
4.4.2.2	RetrieveSub	scription	174
4.4.2.3	Subscription	NotificationBehaviour	175
4.4.2.4	QuerySubsci	riptions	187
4.4.2.5	DeleteSubsc	ription	190
4.4.2.6	CreateSubsc	ription	191
4.4.3	Provision	•	195
4.4.3.1	TemporalEn	tity	195
4.4.3.1.1	UpdateTe	emporalRepresentationOfEntity	195
4.4.3.1.2	DeleteTe	mporalRepresentationOfEntity	196
4.4.3.1.3	CreateTe	mporalRepresentationOfEntity	198
4.4.3.2	BatchEntitie	S	201
4.4.3.2.1	CreateBa	tchOfEntities	201
4.4.3.2.2	DeleteBa	tchOfEntities	209
4.4.3.2.3	UpsertBa	tchOfEntities	212
4.4.3.2.4	UpdateBa	atchOfEntities	218
4.4.3.3	EntityAttribu	ıtes	222
4.4.3.3.1	DeleteEn	tityAttribute	222
4.4.3.3.2	PartialAt	tributeUpdate	227
4.4.3.3.3	UpdateEi	ntityAttributes	232
4.4.3.3.4	AppendE	ntity Attributes	237
4.4.3.4	Entities		243
4.4.3.4.1	CreateEn	tity	243
4.4.3.4.2	DeleteEn	tity	252
4.4.3.5	TemporalEn	tityAttributes	254
4.4.3.5.1	DeleteAt	tribute	254
4.4.3.5.2	DeleteAt	tributeInstance	259
4.4.3.5.3	AddAttri	butes	264
4.4.3.5.4	PartialUp	odateAttributeInstance	267
Annex A	(informative):	Bibliography	273
Annex B ((informative):	Change History	274
History			275

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The declarations pertaining to these essential IPRs, if any, are publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: "Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards", which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (https://ipr.etsi.org/).

Pursuant to the ETSI Directives including the ETSI IPR Policy, no investigation regarding the essentiality of IPRs, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

DECTTM, **PLUGTESTS**TM, **UMTS**TM and the ETSI logo are trademarks of ETSI registered for the benefit of its Members. **3GPP**TM and **LTE**TM are trademarks of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners. **oneM2M**TM logo is a trademark of ETSI registered for the benefit of its Members and of the oneM2M Partners. **GSM**[®] and the GSM logo are trademarks registered and owned by the GSM Association.

Foreword

This Group Specification (GS) has been produced by ETSI Industry Specification Group (ISG) cross-cutting Context Information Management (CIM).

Modal verbs terminology

In the present document "shall", "shall not", "should", "should not", "may", "need not", "will", "will not", "can" and "cannot" are to be interpreted as described in clause 3.2 of the <u>ETSI Drafting Rules</u> (Verbal forms for the expression of provisions).

"must" and "must not" are NOT allowed in ETSI deliverables except when used in direct citation.

1 Scope

The present document contains the description of each abstract test case using the Test Template [i.2] and using the Test Purposes Description Language identified in ETSI GS CIM 012 [2] and ETSI GR CIM 011 [i.1]. The test cases are described in tabular form. All clauses mentioned in tables of clause 4 of the present document are references to the NGSI-LD specification, hence to be found in [1].

2 References

2.1 Normative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

Referenced documents which are not found to be publicly available in the expected location might be found at https://docbox.etsi.org/Reference.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long-term validity.

The following referenced documents are necessary for the application of the present document.

- [1] ETSI GS CIM 009 (V1.5.1) (2021-11): "Context Information Management (CIM); NGSI-LD API".
- [2] <u>ETSI GS CIM 012 (V1.1.1) (2021-03)</u>: "Context Information Management (CIM); NGSI-LD Test Suite Structure".
- [3] Void.

2.2 Informative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long-term validity.

The following referenced documents are not necessary for the application of the present document but they assist the user with regard to a particular subject area.

- [i.1] ETSI GR CIM 011 (V1.1.1) (2021-04): "Context Information Management (CIM); NGSI-LD Testing Framework: Test Purposes Description Language (TPDL)".
- [i.2] ETSI GS CIM 016 (V1.1.1) (2021-04): "Context Information Management (CIM); NGSI-LD Testing Framework: Test Template".

3 Definition of terms, symbols and abbreviations

3.1 Terms

For the purposes of the present document, the following terms apply:

NOTE: The letters "NGSI-LD" were added to most terms to confirm that they are distinct from other terms of

similar/same name in use in other organizations, however, in the present document the letters "NGSI-LD"

are generally omitted for brevity.

NGSI-LD Central Broker: NGSI-LD Context Broker that only uses a local storage when serving NGSI-LD requests, without involving any external Context Sources

NGSI-LD Context Broker: architectural component that implements all the NGSI-LD interfaces

NGSI-LD Context Consumer: agent that uses the query and subscription functionality of NGSI-LD to retrieve context information

NGSI-LD Context Producer: agent that uses the NGSI-LD context provision and/or registration functionality to provide or announce the availability of its context information to an NGSI-LD Context Broker

NGSI-LD Context Registry: software functional element where Context Sources register the information that they can provide

NOTE: It is used by Distribution Brokers and Federation Brokers to find the appropriate Context Sources which can provide the information required for serving an NGSI-LD request.

NGSI-LD Context Source: source of context information which implements the NGSI-LD consumption and subscription (and possibly provision) interfaces defined by the present document

NOTE: It is usually registered with an NGSI-LD Registry so that it can announce what kind of information it can provide, when requested, to Context Consumers and Brokers.

NGSI-LD Distribution Broker: NGSI-LD Context Broker that uses both local context information and registration information from an NGSI-LD Context Registry, to access matching context information from a set of distributed Context Sources

NGSI-LD Federation Broker: Distribution Broker that federates information from multiple underlying NGSI-LD Context Brokers and across domains

3.2 Symbols

Void.

3.3 Abbreviations

For the purposes of the present document, the following abbreviations apply:

CF Config Identifier DEL DELETE

HTTP HperText Transfer Protocol JSON Java Script Object Notation

PICS Profile Implementation Conformance Statement

SUT System Under Test
TP Test Purpose

URI Unified Resource Identification
URL Unified Resource Location

4 Test Purposes Descriptions

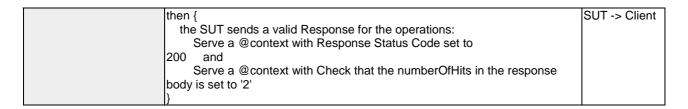
4.1 jsonldContext

4.1.1 Consumption

4.1.1.1 ServeContext

TP ld	TP/NGSI-LD/CTX/Cons/053_06		
Test objective	Check that the numberOfHits is increased after using a Cached context		
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.13.4		
Config Id	CF_01		
PICS Selection	PICS_SerOneSpeUse@co		
Initial conditions	with { the SUT containing a Cached @context created from a entity creation the downloading from external server. }	rough	
	Permutations		
TP Permutation id	TP/NGSI-LD/CTX/Cons/053_06_01		
Name	053_06_01 Check that the numberOfHits is increased after using a Cached co	ontext	
Parent Release	v1.5.1		
Description	Check that the numberOfHits is increased after using a Cached context		
Expected behaviour	Test events	Direction	
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/jsonldContexts/{contextId}' method set to 'GET' Serve a @context with contextID '\$ {uri}' and details 'true' }	SUT <- Client	
	then { the SUT sends a valid Response for the operations: Serve a @context with Response Status Code set to 200 and Serve a @context with Check that the numberOfHits in the response body is set to '2' }	SUT -> Client	

TP/NGSI-LD/CTX/Cons/053_04	
Check that one can get an increase numberOfHits after creation of a Hosted @context and	
using it	
ETSI GS CIM 009 V1.5.1 [1], clause 5.13.4	
CF_01	
PICS_SerOneSpeUse@co	
with { the SUT containing a Cached @context created from a entity creation th	rough
downloading from external server. }	-
Permutations	
TP/NGSI-LD/CTX/Cons/053_04_01	
053_04_01 Check that the numberOfHits is increased after using a Hosted context	
v1.5.1	
Check that the numberOfHits is increased after using a Hosted context	
Test events	Direction
when {	SUT <- Client
the SUT receives a Request from the client containing:	
URL set to '/	
ngsi-ld/v1/jsonldContexts/{contextId}'	
method set to 'GET'	
Serve a @context with contextID '\$	
}	
	using it ETSI GS CIM 009 V1.5.1 [1], clause 5.13.4 CF_01 PICS_SerOneSpeUse@co with { the SUT containing a Cached @context created from a entity creation th downloading from external server. } Permutations TP/NGSI-LD/CTX/Cons/053_04_01 053_04_01 Check that the numberOfHits is increased after using a Hosted cor v1.5.1 Check that the numberOfHits is increased after using a Hosted context Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/jsonIdContexts/{contextId}'

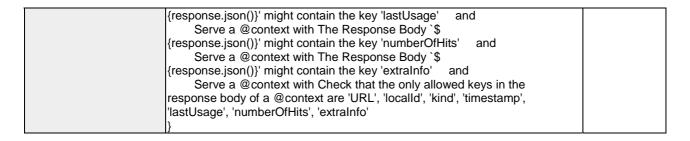


TP Id	TP/NGSI-LD/CTX/Cons/053_02	
Test objective	Check that one can serve a previous created @context with details equal to Ti	rue
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.13.4	
Config Id	CF_01	
PICS Selection	PICS_SerOneSpeUse@co	
Initial conditions	with { the SUT containing a Hosted @context and the default Cached Core Co	ontext. }
	Permutations	
TP Permutation id	TP/NGSI-LD/CTX/Cons/053_02_01	
Name	053_02_01 Serve a @context with details set to True	
Parent Release	v1.5.1	
Description	Check that one can serve a @context with details	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/jsonldContexts/{contextId}' method set to 'GET' Serve a @context with contextID '\$ {uri}' and details 'true' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Serve a @context with Response Status Code set to 200 and Serve a @context with Response reason set to 'OK' and Serve a @context with Response Header: Content- Type set to application/ json and Serve a @context with Check the Body of the response should contain a @context with 'URL' key not Empty and 'string' type, with 'localld' key not Empty and 'string' type, with 'kind' key not Empty, 'string' type, and value set to 'Hosted', and 'timestamp and 'DateTime' format and Serve a @context with The Response Body `\$ {response.json()}' might contain the key 'lastUsage' and Serve a @context with The Response Body `\$ {response.json()}' might contain the key 'numberOfHits' and Serve a @context with The Response Body `\$ {response.json()}' might contain the key 'extraInfo' and Serve a @context with Check that the only allowed keys in the response body of a @context are 'URL', 'localld', 'kind', 'timestamp', 'lastUsage', 'numberOfHits', 'extraInfo'	SUT -> Client

TP Id	TP/NGSI-LD/CTX/Cons/053_05		
Test objective	Check that the context served by a context server is still in the broker after a		
	ERROR_TYPE_LD_CONTEXT_NOT_AVAILABLE with details=true		
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.13.4		
Config Id	CF_01		
PICS Selection	PICS_SerOneSpeUse@co		
Initial conditions	with { the SUT containing a Cached @context created from a entity creation through		
	downloading from external server. }		
	Permutations		
TP Permutation id	TP/NGSI-LD/CTX/Cons/053_05_01		
Name	053_05_01 Check that the context served by a context server is still in the broker after a		
	ERROR_TYPE_LD_CONTEXT_NOT_AVAILABLE with details=true		
Parent Release	v1.5.1		

Description	Check that the context served by a context server is still in the broker after a	
	ERROR_TYPE_LD_CONTEXT_NOT_AVAILABLE with details=true	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/jsonldContexts/{contextId}'	
	method set to 'GET'	
	Serve a @context with contextID '\$	
	{uri}' and details 'true'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Serve a @context with Response Status Code set to	
	200 and	
	Serve a @context with Response reason set to 'OK' and	
	Serve a @context with Response Header: Content-	
	Type set to application/	
	json and	
	Serve a @context with Check the Body of the response should contain	
	a @context with 'URL' key not Empty and 'string' type, with 'localld' key not	
	Empty and 'string' type, with 'kind' key not Empty, 'string' type, and value set	
	to 'Cached', and 'timestamp and 'DateTime' format and Serve a @context with The Response Body `\$	
	{response.json()}' might contain the key 'lastUsage' and	
	Serve a @context with The Response Body `\$	
	{response.json()}' might contain the key 'numberOfHits' and	
	Serve a @context with The Response Body `\$	
	{response.json()}' might contain the key 'extraInfo' and	
	Serve a @context with Check that the only allowed keys in the	
	response body of a @context are 'URL', 'localld', 'kind', 'timestamp',	
	"lastUsage', 'numberOfHits', 'extraInfo'	
	}	
	U	1

TP ld	TD/NCCLLD/CTV/Comp/0F2 07	
	TP/NGSI-LD/CTX/Cons/053_07	
Test objective	Check that one can serve a ImplicitlyCreated @context with details set to true	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.13.4	
Config Id	CF_01	
PICS Selection	PICS_SerOneSpeUse@co	
Initial conditions	with { the SUT containing a ImplicitlyCreated @context created from a subscri	ption query. }
	Permutations	
TP Permutation id	TP/NGSI-LD/CTX/Cons/053_07_01	
Name	053_07_01 Check that one can serve a ImplicitlyCreated @context with details	s set to true
Parent Release	v1.5.1	
Description	Check that one can serve a ImplicitlyCreated @context with details set to true	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/jsonldContexts/{contextId}' method set to 'GET' Serve a @context with contextID '\$ {implicit_id}' and details 'true' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Serve a @context with Response Status Code set to 200 and Serve a @context with Response reason set to 'OK' and Serve a @context with Response Header: Content- Type set to application/ json and Serve a @context with Check the Body of the response should contain a @context with 'URL' key not Empty and 'string' type, with 'localld' key not Empty and 'string' type, with 'kind' key not Empty, 'string' type, and value set to 'ImplicitlyCreated', and 'timestamp and 'DateTime' format and Serve a @context with The Response Body `\$	SUT -> Client



TP Id	TP/NGSI-LD/CTX/Cons/053_03	
Test objective	Check that one gets an error when try to serve a @context	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.13.4	
Config Id	CF_01	
PICS Selection	PICS_SerOneSpeUse@co	
Initial conditions	with { the SUT containing a Hosted @context and the default Cached Core Co	ntext }
	Permutations	THOM: J
TP Permutation id	TP/NGSI-LD/CTX/Cons/053_03_01	
Name	053_03_01 Serve a @contexts with a wrong id and correct details	
Parent Release	V1.5.1	
Description	Check that an error is returned when one requests for a @context that does no	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/jsonldContexts/{contextId}' method set to 'GET' Serve a @context with contextID 'wrong_id_context' and details 'true' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Serve a @context with Response Status Code set to 404 and Serve a @context with Response reason set to 'Not Found' and Serve a @context with Response Body containing the type '\$ {ERROR_TYPE_RESOURCE_NOT_FOUND}' and Response Body Title is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Title is not equal to Response Body Detail	SUT -> Client
	}	
TP Permutation id	TP/NGSI-LD/CTX/Cons/053_03_02	
Name	053_03_02 Serve a @contexts with a valid id and incorrect details	
Parent Release	v1.5.1	
Description	Check that an error is returned when one requests for a @context that does no	ot exist
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/jsonldContexts/{contextId}' method set to 'GET' Serve a @context with contextID '\$ {uri}' and details 'other' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Serve a @context with Response Status Code set to 400 and Serve a @context with Response reason set to 'Bad Request' and Serve a @context with Response Body containing the type 'https:// uri.etsi.org/ngsi-Id/errors/ BadRequestData' and Response Body Title is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY}	SUT -> Client

	and Response Boty Title is not equal to Response Body Detail	
	}	
TP Permutation id	TP/NGSI-LD/CTX/Cons/053_03_03	
Name	053_03_03 Serve a Cached @contexts with details set to false	
Parent Release	v1.5.1	
Description	Check that an error is returned when one requests for a @context that does n	ot exist
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/jsonldContexts/{contextId}' method set to 'GET' Serve a @context with contextID 'https:// uri.etsi.org/ngsi-ld/v1/ngsi-ld-core-context- v1.6.jsonld' and details 'false' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Serve a @context with Response Status Code set to 422 and Serve a @context with Response reason set to 'Unprocessable Content' and Serve a @context with Response Body containing the type 'https:// uri.etsi.org/ngsi-Id/errors/ OperationNotSupported' and Response Body Title is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Title is not equal to Response Body Detail }	SUT -> Client

TP ld	TP/NGSI-LD/CTX/Cons/053_01	
Test objective	Check that one can serve a previous created @context	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.13.4	
Config Id	CF_01	
PICS Selection	PICS_SerOneSpeUse@co	
Initial conditions	with { the SUT containing a Hosted @context and the default Cached Core Co	ntext.}
	Permutations	,
TP Permutation id	TP/NGSI-LD/CTX/Cons/053_01_01	
Name	053_01_01 Serve a @context without details	
Parent Release	v1.5.1	
Description	Check that one can serve a @context with details equal to empty or false	
Expected behaviour	Test events	Direction SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/jsonldContexts/{contextId}' method set to 'GET' Serve a @context with contextID '\$ {uri}' and details '\${EMPTY}' }	SUT -> Client
	then { the SUT sends a valid Response for the operations: Serve a @context with Response Status Code set to 200 and Serve a @context with Response reason set to 'OK' and Serve a @context with Response Header: Content- Type set to application/ json and Serve a @context with Check the Body of the response should contain the @context `@context- minimal-valid.json' }	SUT -> Client
TP Permutation id	TP/NGSI-LD/CTX/Cons/053_01_02	
Name	053_01_02 Serve a @context with details equal to false	
Parent Release	v1.5.1	

Description	Check that one can serve a @context with details equal to empty or false	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/jsonldContexts/{contextId}' method set to 'GET' Serve a @context with contextID '\$ {uri}' and details 'false'	SUT <- Client
	then { the SUT sends a valid Response for the operations: Serve a @context with Response Status Code set to 200 and Serve a @context with Response reason set to 'OK' and Serve a @context with Response Header: Content- Type set to application/ json and Serve a @context with Check the Body of the response should contain the @context`@context- minimal-valid.json' }	SUT -> Client

TP Id	TP/NGSI-LD/CTX/Cons/053_08	
Test objective	Check that the numberOfHits is increased after using a ImplicitlyCreated conte	ext
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.13.4	
Config Id	CF_01	
PICS Selection	PICS_SerOneSpeUse@co	
Initial conditions	with { the SUT containing a ImplicitlyCreated @context created from a subscription	otion query. }
	Permutations	
TP Permutation id	TP/NGSI-LD/CTX/Cons/053_08_01	
Name	053_08_01 Check that the numberOfHits is increased after using a ImplicitlyC	reated context
Parent Release	v1.5.1	
Description	Check that the numberOfHits is increased after using a ImplicitlyCreated conte	ext
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/jsonldContexts/{contextId}' method set to 'GET' Serve a @context with contextID '\$ {implicit_id}' and details 'true' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Serve a @context with Response Status Code set to 200 and Serve a @context with Check that the numberOfHits in the response body is set to '2' }	SUT -> Client

4.1.1.2 ListContexts

TP Id	TP/NGSI-LD/CTX/Cons/052_06		
Test objective	Check that one can list all the @context available in the broker with no previous add		
	@ context		
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.13.3		
Config Id	CF_01		
PICS Selection	PICS_LisAllCac@co		
Initial conditions	with { the SUT containing a ImplicitlyCreated @context created from a subscription query. }		
	Permutations		
TP Permutation id	TP/NGSI-LD/CTX/Cons/052_06_01		
Name	052_06_01 List @contexts with neither details or kind and a created ImplicitlyCreated		
	@context		

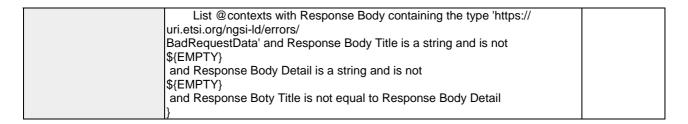
Parent Release	v1.5.1	
Description	Check that one can list @contexts	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-Id/v1/jsonIdContexts?\${params_as_string}' method set to 'GET'	SUT <- Client
	List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to '\${EMPTY}' }	
	then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and Query Parameter: response_body set to 'response.json()' and Query Parameter: expected_length set to '4' and Overy Parameter: list context to list context to list a context t	SUT -> Client
	Query Parameter: list_contexts set to 'list_contexts'	
TP Permutation id	TP/NGSI-LD/CTX/Cons/052_06_02	ı
Name	052_06_02 List @contexts with no details and kind set to hosted and a create ImplicitlyCreated @context	d
Parent Release	v1.5.1	
Description	Check that one can list @contexts	
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ngsi-Id/v1/jsonIdContexts?\${params_as_string}' method set to 'GET'	SUT <- Client
	List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'Hosted' }	
	then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and Query Parameter: response_body set to 'response.json()' and	SUT -> Client
	Query Parameter: expected_length set to 'lo' and Query Parameter: list_contexts set to 'list_contexts'	
TP Permutation id	} TP/NGSI-LD/CTX/Cons/052_06_03	
Name	052_06_03 List @contexts with no details and kind set to cached and a create ImplicitlyCreated @context	ed
Parent Release	v1.5.1	
Description	Check that one can list @contexts	Direction
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ngsi-Id/v1/jsonIdContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'Cached' }	Direction SUT <- Client
	then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and	SUT -> Client

	Query Parameter: response_body set to 'response.json()' and	
	Query Parameter: expected_length set to '3' and	
	Query Parameter: list_contexts set to 'list_contexts'	
TP Permutation id	TP/NGSI-LD/CTX/Cons/052_06_04	
Name	052_06_04 List @contexts with no details and kind set to implicitlycreated and	d a created
	ImplicitlyCreated @context	
Parent Release	v1.5.1	
Description	Check that one can list @contexts	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-Id/v1/jsonIdContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'ImplicitlyCreated'	SUT <- Client
	then {	SUT -> Client
	the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and Query Parameter: response_body set to 'response.json()' and Query Parameter: expected_length set to '1' and Query Parameter: list_contexts set to 'list_contexts'	Sor -> chem
	}	
TP Permutation id	TP/NGSI-LD/CTX/Cons/052_06_05	•
Name	052_06_05 List @contexts with details set to false and no kind and a created	
	ImplicitlyCreated @context	
Parent Release	v1.5.1	
Description Expected behaviour	Check that one can list @contexts	Direction
Expected behaviour	Test events when {	Direction SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ngsi-Id/v1/jsonIdContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to 'false' and Query Parameter: kind set to '\${EMPTY}'	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and Query Parameter: response_body set to 'response.json()' and Query Parameter: expected_length set to '4' and Query Parameter: list_contexts set to 'list_contexts' }	
TP Permutation id	TP/NGSI-LD/CTX/Cons/052_06_06	
Name	052_06_06 List @contexts with details set to false and kind equal to hosted a ImplicitlyCreated @context	nd a created
Parent Release	v1.5.1	
Description	Check that one can list @contexts	Direction
Expected behaviour	Test events	Direction SUT <- Client
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-Id/v1/jsonIdContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to 'false' and	SOT <- Client

	Query Parameter: kind set to 'Hosted'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	List @contexts with Response Status Code set to	
	200 and	
	List @contexts with Response reason set to 'OK' and	
	List @contexts with Check Context Response Body Containing a list of identifiers and	
	Query Parameter: response_body set to 'response.json()' and	
	Query Parameter: expected_length set to '0' and	
	Query Parameter: list_contexts set to 'list_contexts'	
	}	
TP Permutation id	TP/NGSI-LD/CTX/Cons/052_06_07	•
Name	052_06_07 List @contexts with details set to false and kind equal to cached a	and a created
	ImplicitlyCreated @context	
Parent Release	v1.5.1	
Description	Check that one can list @contexts	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}'	
	method set to 'GET'	
	List @contexts: and	
	Query Parameter: details set to 'false' and	
	Query Parameter: kind set to 'Cached'	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	301 -> Client
	List @contexts with Response Status Code set to	
	200 and	
	List @contexts with Response reason set to 'OK' and	
	List @contexts with Check Context Response Body Containing a list of	
	lidentifiers and	
	Query Parameter: response_body set to 'response.json()' and	
	Query Parameter: expected_length set to '3' and	
	Query Parameter: list_contexts set to 'list_contexts'	
	}	
TP Permutation id	TP/NGSI-LD/CTX/Cons/052_06_08	
Name	052_06_08 List @contexts with details set to false and kind equal to implicitly	created and a
D (D)	created ImplicitlyCreated @context	
Parent Release	v1.5.1	
Description	Check that one can list @contexts	Discotion
Expected behaviour	Test events	Direction Client
	when { the SUT receives a Request from the client containing:	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}'	
	method set to 'GET'	
	List @contexts: and	
	Query Parameter: details set to 'false' and	
	Query Parameter: kind set to 'ImplicitlyCreated'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	List @contexts with Response Status Code set to	
	200 and	
	List @contexts with Response reason set to 'OK' and	
	List @contexts with Check Context Response Body Containing a list of	
	identifiers and	
	Query Parameter: response_body set to 'response.json()' and	
	Query Parameter: expected_length set to '1' and	
	Query Parameter: list_contexts set to 'list_contexts'	
	l.	

TP ld	TP/NGSI-LD/CTX/Cons/052_04	
Test objective	Check that one gets an error when try to list @context with wrong details or	r kind
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.13.3	-
Config Id	CF_01	
PICS Selection	PICS_LisAllCac@co	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CTX/Cons/052_04_01	
Name	052_04_01 List @contexts with no details and kind set to other and not pre	viously created
	@context	
Parent Release	v1.5.1	
Description	Check that one can list @contexts	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'other' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 400 and List @contexts with Response reason set to 'Bad Request'	SUT -> Client
TD Downstation id	TD/NOOLLD/OTY/O/050, 04, 00	
TP Permutation id Name	TP/NGSI-LD/CTX/Cons/052_04_02 052_04_02 List @contexts with details set to other and no kind and not pre	viously created
Name	@context	viously created
Parent Release	V1.5.1	
Description	Check that one can list @contexts	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to 'other' and Query Parameter: kind set to '\${EMPTY}' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 400 and List @contexts with Response reason set to 'Bad Request' and List @contexts with Response Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/ BadRequestData' and Response Body Title is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Title is not equal to Response Body Detail	SUT -> Client
TP Permutation id	TP/NGSI-LD/CTX/Cons/052_04_03	l
Name	052_04_03 List @contexts with details set to other and kind set to other an	d not previously
Donant Dalasa	created @context	
Parent Release Description	v1.5.1 Check that one can list @contexts	
	IL DOCK TOST ONE CON LICT (II) CONTENTS	

when {	Expected behaviour	Test events	Direction
the SUT receives a Request from the client containing: URL set to 'ingslid-Vilsional/Gontexts'\$(params_as_string)' method set to 'GET' List '@contexts' and Query Parameter details set to 'other' and Query Parameter kind set to 'other') then {	Expedica Bellaviour		
URL set to 'ngsi-ldv1/sonidContexts?\$(params_as_string)' method set to 'GET List @contexts: and Ouery Parameter: kind set to 'other' and Ouery Parameter: kind set to 'other' } the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 400 and List @contexts with Response Body containing the type https:// url.elsi.org/ngsi-ld/errors/ BacRequestDatal and Response Body Title is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Title is not equal to Response Body Detail TP Permutation Id Name 052.04 A Use List @contexts with details set to true and kind set to other and not previously created @context Check that one can list @contexts Description Expected behaviour Test events Direction when (the SUT receives a Request from the client containing: URL set to 'ngsi-ld/d/1/son/id/Contexts'?(params_as_string)' method set to GET List @contexts: and Ouery Parameter clealis set to 'true' and Ouery Parameter set details set to 'true' and Ouery Parameter set set set of the parameter set of set set of the set of set			301 C Client
method set to "GET" List @contexts: and Query Parameter. details set to 'other' and Query Parameter. details set to 'other' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 400 and List @contexts with Response Body containing the type https:// unlests.org/ngsi-td/errors/ BadRequestData' and Response Body Title is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Title is not equal to Response Body Detail TPP/INGSI-LD/CTX/Cons/052_04_04 Name 052_04_04 List @contexts with details set to true and kind set to other and not previously created @context Test events When { the SUT receives a Request from the client containing: URL set to 'ngsi-id/v1/jsenidContexts'?\$(params_as_string)' method set to 'GET' List @contexts and Query Parameter, kind set to 'other' } URL set to 'other' the SUT sends a valid Response for the operations: List @contexts with Response Body Containing the type 'https:// unless long/nsi-id/v1/jsenidContexts'?\$(params_as_string)' method set to 'GET' List @contexts with Response Params and Interest to 'and List @contexts with Response Params and Interest to 'and Response Body Detail is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) SEMPTY) and Response Body Detail is a string and is not \$(EMPTY) SEMPTY) and Response Body Detail is a string and is not \$(EMPTY) SEMPTY) and Response Body Detail is a string and is not \$(EMPTY) SEMPTY) and Response Body Detail is a string and is not \$(EMPTY) SEMPTY) and Response Body Detail is a string and is not \$(EMPTY) SEMPTY) and Response Body Detail is a string and is not \$(EMPTY) SEMPTY S			
List @contexts; and Query Parameter ideals set to 'other' and Query Parameter ideals set to 'other' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 400 and List @contexts with Response Body containing the type 'https:// urt.esi.org/ingsl-id/errors' BadRequestData' and Response Body Detail is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Title is not equal to Response Body Detail } TP Permutation id Name 052 04 04 Usi @contexts with details set to true and kind set to other and not previously created @context vi.5.1 Description Check that one can list @contexts Test events Ust. set to 'ingsl-id/v1/sindldContexts'?\$(params_as_string)' method set to 'GET' List @contexts with Response Status Code set to 400 and List @contexts with Response Status Code set to 400 and List @contexts with Response Body Title is a string and is not \$(EMPTY) surfameter kind set to 'other' } then \$UT sends a valid Response for the operations: List @contexts with Response Body Containing the type 'https:// urt.esto.org/inschled/errors' BadRequestData' and Response Body Title is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Body Det			
Query Parameter: kind set to 'other' and Query Parameter: kind set to 'other' and Query Parameter: kind set to 'other'			
Cuery Parameter kind set to 'other' the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 400 and List @contexts with Response Body containing the type https:// uni.ests.org/ingsl-id/errors/ BadRequestData' and Response Body Containing the type https:// uni.ests.org/ingsl-id/errors/ BadRequestData' and Response Body Detail is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Title is not equal to Response Body Detail \$(EMPTY) and Response Body Title is not equal to Response Body Detail \$(EMPTY) and Response Body Detail is a string and kind set to other and not previously created @context \$(EMPTY) and Response Body Empty (India) \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Body Detail is a			
then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 400 and List @contexts with Response Body containing the type 'https:// uni.etsi.org/ngsi-ld/errors/ BadRequestData' and Response Body Containing the type 'https:// uni.etsi.org/ngsi-ld/errors/ BadRequestData' and Response Body Title is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Detail is not equal to Response Body Detail } TP Permutation id TP:NGSI-LD/CTX/Cons/052_04_04 TP:NGSI-LD/CTX/Cons/052_04_04 Tesponse Body Detail is a string and is not \${EMPTY} and Response Body Title is not equal to Response Body Detail Fepceted behaviour Test events When { the SUT receives a Request from the client containing: URL set to 'ngsi-ld/v1/sond/Contexts'?\${params_as_string}}' method set to 'GET List @contexts with Response for the operations: List @contexts with Response Body Containing the type 'https:// uni.etsi.org/ngsi-ld/errors/ BadRequestData' and Response Body Title is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Title is not equal to Response Body Detail TP:NGSI-LD/CTX/Cons/052_04_05 Name 052_04_05 List @contexts with details set to other and kind set to Hosted and not previously created @context Test events Direction Test events Direction When { the SUT receives a Request from the client containing: URL set to 'ngsi-ld/v1/sjond/Contexts'?\${params_as_string}}' method set to 'GET List @contexts: and Query Parameter details set to 'other' and Query Parameter details set to 'ot			
the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 400 and List @contexts with Response reason set to 'Bad Request' and List @contexts with Response Body containing the type 'https:// uri.etsi.org/ngsi-d/errors/ BadRequestData' and Response Body Title is a string and is not S[EMPTY] and Response Body Detail is a string and is not S[EMPTY] and Response Body Detail is a string and is not S[EMPTY] and Response Body Detail is not equal to Response Body Detail TP-Permutation id TP:/NGSI-LD/CTX/Cons/052_04_04 TP:/NGSI-LD/CTX/Cons/052_04_05 Test events Test events Test events Test events Test events Test events Direction SUT < Client URL set to 'Ingsi-Id/'1/spoind/Contexts?'\$(params_as_string)' method set to 'GET' List @contexts with Response For the operations: List @contexts with Response Body Containing the type 'https:// uri.etsi.org/ngsi-id/errors/ BadRequestData' and Response Body Title is a string and is not S[EMPTY] and Response Body Detail is a string and is not S[EMPTY] and Response Body Detail is a string and is not S[EMPTY] and Response Body Detail is a string and is not S[EMPTY] and Response Body Detail is a string and is not S[EMPTY] and Response Body Detail is a string and is not S[EMPTY] and Response Body Detail is a string and is not S[EMPTY] and Response Body Detail is a string and is not S[EMPTY] and Response Body Detail is a string and is not S[EMPTY] and Response Body Detail is a string and is not S[EMPTY] and Response Body Detail is a string and is not S[EMPTY] and Response Body Detail is a string and is not S[EMPTY] and Response Body Detail is a string and is not S[EMPTY] and Response Body Detail is a string and is not S[EMPTY] and Response Body Detail is a string and is not S[EMPTY] and Response Body Detail is a string and is not S[EMPTY] and Response Body Detail is a string and is not S[EMPTY] and Response Body Detail is a string and is not S[EMPTY] and Response Body Detail is a string and is not S[EMPTY] and Response Body Detail is a s		Query Parameter: kind set to 'other'	
the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 400 and List @contexts with Response reason set to 'Bad Request' and List @contexts with Response Body containing the type 'https:// uri.etsi.org/ngsi-d/errors/ BadRequestData' and Response Body Title is a string and is not S[EMPTY] and Response Body Detail is a string and is not S[EMPTY] and Response Body Detail is a string and is not S[EMPTY] and Response Body Detail is not equal to Response Body Detail TP-Permutation id TP:/NGSI-LD/CTX/Cons/052_04_04 TP:/NGSI-LD/CTX/Cons/052_04_05 Test events Test events Test events Test events Test events Test events Direction SUT < Client URL set to 'Ingsi-Id/'1/spoind/Contexts?'\$(params_as_string)' method set to 'GET' List @contexts with Response For the operations: List @contexts with Response Body Containing the type 'https:// uri.etsi.org/ngsi-id/errors/ BadRequestData' and Response Body Title is a string and is not S[EMPTY] and Response Body Detail is a string and is not S[EMPTY] and Response Body Detail is a string and is not S[EMPTY] and Response Body Detail is a string and is not S[EMPTY] and Response Body Detail is a string and is not S[EMPTY] and Response Body Detail is a string and is not S[EMPTY] and Response Body Detail is a string and is not S[EMPTY] and Response Body Detail is a string and is not S[EMPTY] and Response Body Detail is a string and is not S[EMPTY] and Response Body Detail is a string and is not S[EMPTY] and Response Body Detail is a string and is not S[EMPTY] and Response Body Detail is a string and is not S[EMPTY] and Response Body Detail is a string and is not S[EMPTY] and Response Body Detail is a string and is not S[EMPTY] and Response Body Detail is a string and is not S[EMPTY] and Response Body Detail is a string and is not S[EMPTY] and Response Body Detail is a string and is not S[EMPTY] and Response Body Detail is a string and is not S[EMPTY] and Response Body Detail is a string and is not S[EMPTY] and Response Body Detail is a s		}	
List @contexts with Response Status Code set to 400 and Ust @contexts with Response reason set to 'Bad Request' and List @contexts with Response Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/ BadRequestDtaid' and Response Body Title is a string and is not \${EMPTY}{} and Response Body Detail is a string and is not \${EMPTY}{} and Response Body Detail is a string and is not \${EMPTY}{} and Response Body Detail is a string and is not \${EMPTY}{} and Response Body Title is not equal to Response Body Detail } TP Permutation id TPNGSI-LD/CTX/Cons/052_04_04 Name 052_04_04 List @contexts with details set to true and kind set to other and not previously created @context V1.5.1 Description Expected behaviour Test events Direction When { the SUT receives a Request from the client containing: URL set to 'ingsi-ldu/fjsonldContexts'?\$[params_as_string]' method set to 'GET List @contexts: and Query Parameter: details set to 'true' and Query Parameter: details set to 'true' and Query Parameter: details set to 'true' and Query Parameter: kind set to 'other' } then { the SUT sends a valid Response for the operations: List @contexts with Responses Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/ BadRequestDtair' and Response Body Title is a string and is not \${EMPTY}{} and Response Body Detail is a string and is not \${EMPTY}{} and Response Body Detail is a string and is not \${EMPTY}{} and Response Body Detail is a string and is not \${EMPTY}{} and Response Body Detail is a string and is not \${EMPTY}{} and Response Body Detail is a string and is not \${EMPTY}{} and Response Body Detail is a string and is not \${EMPTY}{} and Response Body Detail is a string and is not \${EMPTY}{} and Response Body Detail is a string and is not \${EMPTY}{} and Response Body Detail is a string and is not \${EMPTY}{} and Response Body Detail is a string and is not \${EMPTY}{} and Response Body Detail is a string and is not \${EMPTY}{} and Response Body Detail is a string and is not \${EMPTY}{} and Response Body Detail is			SUT -> Client
400 and List @contexts with Response reason set to 'Bad Request' and List @contexts with Response Body containing the type 'https:// uri-eisto.gr/ngsi-ld/errors/ BadRequestData' and Response Body Title is a string and is not \${EMPTY} and Response Body Detail is a		the SUT sends a valid Response for the operations:	
List @contexts with Response Body containing the type 'https:// uri.etsi.org/ngsi-d/errors/ BadRequestData' and Response Body Title is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Detail is a string and kind set to other and not previously created @context V1.5.1 Description Check that one can list @contexts Parent Release V1.5.1 Description Check that one can list @contexts When { the \$UT receives a Request from the client containing:		List @contexts with Response Status Code set to	
List @contexts with Response Body containing the type 'https:// uni-etsi-org/ingsi-Id/errors/ BadRequestData' and Response Body Title is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Title is not equal to Response Body Detail TP Permutation id TP:NGSI-LD/CTX/Cons/052_04_04 Test events V1.5.1 Direction Test events When { the SUT receives a Request from the client containing: Urls set to 'ngsi-Id/v1/sponidContexts?\${params_as_string}} Unitests with Response Body Containing the type 'https:// unitests with Response Status Code set to 400 and List @contexts with Response Body Containing the type 'https:// unitests with Response Body Title is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Detail is } TP Permutation Id Name		· ·	
List @contexts with Response Body containing the type 'https:// uni-etsi-org/ingsi-Id/errors/ BadRequestData' and Response Body Title is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Title is not equal to Response Body Detail TP Permutation id TP:NGSI-LD/CTX/Cons/052_04_04 Test events V1.5.1 Direction Test events When { the SUT receives a Request from the client containing: Urls set to 'ngsi-Id/v1/sponidContexts?\${params_as_string}} Unitests with Response Body Containing the type 'https:// unitests with Response Status Code set to 400 and List @contexts with Response Body Containing the type 'https:// unitests with Response Body Title is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Detail is } TP Permutation Id Name		List @contexts with Response reason set to 'Bad Request' and	
Unit ests org/ingsi-ld/errors/			
BadRequestData' and Response Body Title is a string and is not \$\(\) \{EMPTY\\ \) and Response Body Detail is a string and is not \$\(\) \{EMPTY\\ \) and Response Body Detail is a string and is not \$\(\) \{EMPTY\\ \) and Response Boty Title is not equal to Response Body Detail \\ \text{Name} TP Permutation id			
SEMPTY and Response Body Detail is a string and is not SEMPTY and Response Body Detail TP/NGSI-LD/CTX/Cons/052_04_04 SEMPTY and Response Body Detail SUT -> Client SUT -> Client SUT -> Client SEMPTY and Response Body Detail SEMPTY and Response Body Detail SUT -> Client SUT -> Client SEMPTY SEMPTY SUT -> Client SUT -> Client SUT -> Client SEMPTY SUT -> Client S			
and Response Body Detail is a string and is not \${EKPMTY}; and Response Boty Title is not equal to Response Body Detail TP Permutation id TP/NGSI-LD/CTX/Cons/052_04_04 Name 052_04_04 List @contexts with details set to true and kind set to other and not previously created @context Parent Release V1.5.1 Description Check that one can list @contexts Test events When { the SUT receives a Request from the client containing:			
StEMPTY and Response Boty Title is not equal to Response Body Detail			
and Response Boty Title is not equal to Response Body Detail TP/NGSI-LD/CTX/Cons/052_04_04 Name			
TP Permutation id TP/NGSI-LD/CTX/Cons/052_04_04 Name 052_04_04 List @contexts with details set to true and kind set to other and not previously created @context Check that one can list @contexts Check that one can list @contexts Direction			
Different Release		and Response Boty Title is not equal to Response Body Detail	
Different Release	TD D	TRAIGNIA BIOTIVIO	
Created @context			
Parent Release v1.5.1 Check that one can list @contexts Test events Direction	Name		ot previously
Check that one can list @contexts Test events			
Test events			
When {			
the SUT receives a Request from the client containing:	Expected behaviour	Test events	
URL set to 'ngsi-Id/vi/jsonIdContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to 'true' and Query Parameter: kind set to 'other' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 400 and List @contexts with Response reason set to 'Bad Request' and List @contexts with Response Body containing the type 'https:// uri.etsi.org/ngsi-Id/errors/ BadRequestData' and Response Body Title is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Detail is not equal to Response Body Detail } TP Permutation id TP/NGSI-LD/CTX/Cons/052_04_05 Name 052_04_05 List @contexts with details set to other and kind set to Hosted and not previously created @context 71.5.1 Description Check that one can list @contexts When { the SUT receives a Request from the client containing: URL set to 'ngsi-Id/vi/jsonIdContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: kind set to 'Hosted' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 400 and			SUT <- Client
method set to 'GET' List @contexts: and Query Parameter: details set to 'true' and Query Parameter: kind set to 'other' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 400 and List @contexts with Response Body containing the type 'https:// uri.etsi.org/ngsi.ld/errors/ BadRequestData' and Response Body Title is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Title is not equal to Response Body Detail } TP Permutation Id TP/NGSI-LD/CTX/Cons/052_04_05 Name 052_04_05 List @contexts with details set to other and kind set to Hosted and not previously created @context V1.5.1 Description Expected behaviour Test events when { the SUT receives a Request from the client containing: URL set to 'ngsi-Id/v1/jsonIdContexts?\${params_as_string}}' method set to 'GET' List @contexts: and Query Parameter: kind set to 'thosted' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 400 and		the SUT receives a Request from the client containing:	
method set to 'GET' List @contexts: and Query Parameter: details set to 'true' and Query Parameter: kind set to 'other' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 400 and List @contexts with Response Body containing the type 'https:// uri.etsi.org/ngsi.ld/errors/ BadRequestData' and Response Body Title is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Title is not equal to Response Body Detail } TP Permutation id TP/NGSI-LD/CTX/Cons/052_04_05 Name 052_04_05 List @contexts with details set to other and kind set to Hosted and not previously created @context V1.5.1 Description Expected behaviour Test events when { the SUT receives a Request from the client containing:		URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}'	
Query Parameter: details set to 'true' and Query Parameter: kind set to 'other' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 400 and List @contexts with Response Body containing the type 'https:// uri.etsi.org/ngsi-Id/errors/ BadRequestData' and Response Body Title is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Detail to Response Body Detail } TP Permutation id TP/NGSI-LD/CTX/Cons/052_04_05 Name 052_04_05 List @contexts with details set to other and kind set to Hosted and not previously created @context Parent Release v1.5.1 Description Check that one can list @contexts Expected behaviour the SUT receives a Request from the client containing: URL set to 'ngsi-Id/v1/sonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to 'other' and Query Parameter:		method set to 'GET'	
Query Parameter: details set to 'true' and Query Parameter: kind set to 'other' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 400 and List @contexts with Response Body containing the type 'https:// uri.etsi.org/ngsi-Id/errors/ BadRequestData' and Response Body Title is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Detail to Response Body Detail } TP Permutation id TP/NGSI-LD/CTX/Cons/052_04_05 Name 052_04_05 List @contexts with details set to other and kind set to Hosted and not previously created @context Parent Release v1.5.1 Description Check that one can list @contexts Expected behaviour when { the SUT receives a Request from the client containing: URL set to 'ngsi-Id/v1/sonldContexts?\${params_as_string}}' method set to 'GET' List @contexts: and Query Parameter: details set to 'other' and		List @contexts: and	
Query Parameter: kind set to 'other' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 400 and List @contexts with Response reason set to 'Bad Request' and List @contexts with Response Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/ BadRequestData' and Response Body Title is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Boty Title is not equal to Response Body Detail } TP Permutation id TP/NGSI-LD/CTX/Cons/052_04_05 Name 052_04_05 List @contexts with details set to other and kind set to Hosted and not previously created @context Parent Release v1.5.1 Description Expected behaviour Test events Test events When { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonld/Contexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to 'other' and Query Parameter: details set to 'hosted' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 400 and			
then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 400 and List @contexts with Response reason set to 'Bad Request' and List @contexts with Response Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/ BadRequestData' and Response Body Title is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Boty Title is not equal to Response Body Detail } TP/NGSI-LD/CTX/Cons/052_04_05 Name 052_04_05 List @contexts with details set to other and kind set to Hosted and not previously created @context Parent Release v1.5.1 Description Check that one can list @contexts Expected behaviour Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET List @contexts: and Query Parameter: details set to 'other' and Query Parameter: kind set to 'Hosted' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 400 and			
the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 400 and List @contexts with Response reason set to 'Bad Request' and List @contexts with Response Body containing the type 'https:// uri.etsi.org/ngsi-id/errors/ BadRequestData' and Response Body Title is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Title is not equal to Response Body Detail **Presponse Body Detail is a string and is not **S{EMPTY} and Response Body Title is a string and is not **S{EMPTY} and Response Body Title is a string and is not **S{EMPTY} and Response Body Title is a string and is not **S{EMPTY} and Response Body Title is a string and is not **S{EMPTY} and Response Body Title is a string and is not **S{EMPTY} and Response Body Title is a string and is not **S{EMPTY} and Response Body Title is a string and is not **S{EMPTY} and Response Body Title is a string and is not **S{EMPTY} and Response Body Title is a string and is not **S{EMPTY} and Response Body Title is a string and is not **S{EMPTY} and Response Body Title is a string and is not **S{EMPTY} and Response Body Title is not **S{EMPTY} and		Real of the state	
the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 400 and List @contexts with Response reason set to 'Bad Request' and List @contexts with Response Body containing the type 'https:// uri.etsi.org/ngsi-id/errors/ BadRequestData' and Response Body Title is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Title is not equal to Response Body Detail **Presponse Body Detail is a string and is not **S{EMPTY} and Response Body Title is a string and is not **S{EMPTY} and Response Body Title is a string and is not **S{EMPTY} and Response Body Title is a string and is not **S{EMPTY} and Response Body Title is a string and is not **S{EMPTY} and Response Body Title is a string and is not **S{EMPTY} and Response Body Title is a string and is not **S{EMPTY} and Response Body Title is a string and is not **S{EMPTY} and Response Body Title is a string and is not **S{EMPTY} and Response Body Title is a string and is not **S{EMPTY} and Response Body Title is a string and is not **S{EMPTY} and Response Body Title is a string and is not **S{EMPTY} and Response Body Title is not **S{EMPTY} and		than /	SLIT -> Client
List @contexts with Response Status Code set to 400 and List @contexts with Response reason set to 'Bad Request' and List @contexts with Response Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/ BadRequestData' and Response Body Title is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Boty Title is not equal to Response Body Detail } TP Permutation id TP/NGSI-LD/CTX/Cons/052_04_05 Name 052_04_05 List @contexts with details set to other and kind set to Hosted and not previously created @context V1.5.1 Description Expected behaviour Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to 'other' and Query Parameter: kind set to 'Hosted' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 400 and			OUT -> Ollerit
400 and List @contexts with Response reason set to 'Bad Request' and List @contexts with Response Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/ BadRequestData' and Response Body Title is a string and is not \${EMPTY} and Response Body Detail \${EMPTY} and Respons			
List @contexts with Response reason set to 'Bad Request' and List @contexts with Response Body containing the type 'https:// uri.etsi.org/ngsi-Id/errors/ BadRequestData' and Response Body Title is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMP		·	
List @contexts with Response Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/ BadRequestData' and Response Body Title is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Boty Title is not equal to Response Body Detail } TP Permutation id TP/NGSI-LD/CTX/Cons/052_04_05 Name 052_04_05 List @contexts with details set to other and kind set to Hosted and not previously created @context Parent Release v1.5.1 Description Check that one can list @contexts Expected behaviour Test events Direction When {			
uri.etsi.org/ngsi-Id/errors/ BadRequestData' and Response Body Title is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Boty Title is not equal to Response Body Detail } TP Permutation id TP/NGSI-LD/CTX/Cons/052_04_05 Name 052_04_05 List @contexts with details set to other and kind set to Hosted and not previously created @context V1.5.1 Description Check that one can list @contexts Expected behaviour Test events Direction When { the SUT receives a Request from the client containing: URL set to '/ngsi-Id/v1/jsonIdContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to 'other' and Query Parameter: kind set to 'Hosted' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 400 and			
BadRequestData' and Response Body Title is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Body Detail is a string and is not \${EMPTY}\$ and Response Boty Title is not equal to Response Body Detail } TP Permutation id TP/NGSI-LD/CTX/Cons/052_04_05 Name 052_04_05 List @contexts with details set to other and kind set to Hosted and not previously created @context Parent Release v1.5.1 Description Check that one can list @contexts Expected behaviour Test events Direction when {			
\${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Detail is not equal to Response Body Detail } TP Permutation id TP/NGSI-LD/CTX/Cons/052_04_05 Name			
and Response Body Detail is a string and is not \${EMPTY} and Response Boty Title is not equal to Response Body Detail } TP/NGSI-LD/CTX/Cons/052_04_05 Name 052_04_05 List @contexts with details set to other and kind set to Hosted and not previously created @context V1.5.1 Description Check that one can list @contexts Expected behaviour Test events Direction when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to 'other' and Query Parameter: details set to 'Hosted' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 400 and			
\${EMPTY} and Response Boty Title is not equal to Response Body Detail } TP Permutation id TP/NGSI-LD/CTX/Cons/052_04_05 Name 052_04_05 List @contexts with details set to other and kind set to Hosted and not previously created @context Parent Release v1.5.1 Description Check that one can list @contexts Expected behaviour Test events Direction when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to 'other' and Query Parameter: kind set to 'Hosted' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 400 and			
and Response Boty Title is not equal to Response Body Detail TP Permutation id TP/NGSI-LD/CTX/Cons/052_04_05 Name 052_04_05 List @contexts with details set to other and kind set to Hosted and not previously created @context Parent Release V1.5.1 Description Check that one can list @contexts Expected behaviour Test events When { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to 'other' and Query Parameter: kind set to 'Hosted' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 400 and			
TP Permutation id TP/NGSI-LD/CTX/Cons/052_04_05 Name 052_04_05 List @contexts with details set to other and kind set to Hosted and not previously created @context Parent Release V1.5.1 Description Check that one can list @contexts Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to 'other' and Query Parameter: kind set to 'Hosted' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 400 and			
Name		and Response Boty Title is not equal to Response Body Detail	
Name		}	
previously created @context Parent Release v1.5.1 Description Check that one can list @contexts Expected behaviour Test events Direction when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to 'other' and Query Parameter: kind set to 'Hosted' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 400 and	TP Permutation id	TP/NGSI-LD/CTX/Cons/052_04_05	
previously created @context Parent Release v1.5.1 Description Check that one can list @contexts Expected behaviour Test events Direction when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to 'other' and Query Parameter: kind set to 'Hosted' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 400 and	Name	052 04 05 List @contexts with details set to other and kind set to Hosted and	not
Parent Release v1.5.1			
Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to 'other' and Query Parameter: kind set to 'Hosted' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 400 and	Parent Release	v1.5.1	
Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to 'other' and Query Parameter: kind set to 'Hosted' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 400 and	Description	Check that one can list @contexts	
when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to 'other' and Query Parameter: kind set to 'Hosted' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 400 and			Direction
the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to 'other' and Query Parameter: kind set to 'Hosted' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 400 and	•	when {	
URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to 'other' and Query Parameter: kind set to 'Hosted' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 400 and			
method set to 'GET' List @contexts: and Query Parameter: details set to 'other' and Query Parameter: kind set to 'Hosted' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 400 and			
List @contexts: and Query Parameter: details set to 'other' and Query Parameter: kind set to 'Hosted' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 400 and			
Query Parameter: details set to 'other' and Query Parameter: kind set to 'Hosted' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 400 and			
Query Parameter: kind set to 'Hosted' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 400 and			
then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 400 and			
the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 400 and		Query Farameter, kind Set to Hosted	
the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 400 and		} 	OUT OF I
List @contexts with Response Status Code set to 400 and			SUI -> Client
400 and			
		·	
List @contexts with Response reason set to 'Bad Request' and			
		List @contexts with Response reason set to 'Bad Request' and	



TP Id	TP/NGSI-LD/CTX/Cons/052_02	
Test objective	Check that one can list all the @context available in the broker with one add @	Doontoyt
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.13.3	ECOMEX
	CF_01	
Config Id		
PICS Selection	PICS_LisAllCac@co	
Initial conditions	with { the SUT containing a Hosted @context and the default Cached Core Co	ntext. }
	Permutations	
TP Permutation id	TP/NGSI-LD/CTX/Cons/052_02_01	
Name	052_02_01 List @contexts with neither details or kind and with previously one	add @context
Parent Release	v1.5.1	
Description	Check that one can list @contexts	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to '\${EMPTY}' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and Query Parameter: response_body set to 'response.json()' and Query Parameter: expected_length set to '2' and Query Parameter: list_contexts set to 'uri_list' and Query Parameter: kind set to '\$ {EMPTY}'	SUT <- Client SUT -> Client
	}	
TP Permutation id	TP/NGSI-LD/CTX/Cons/052_02_02	
Name	052_02_02 List @contexts with no details and kind equal to hosted and with p add @context	reviously one
Parent Release	v1.5.1	
Description	Check that one can list @contexts	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'Hosted' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and Query Parameter: response_body set to 'response.json()' and Query Parameter: expected_length set to '1' and Query Parameter: list_contexts set to 'uri_list' and	SUT -> Client

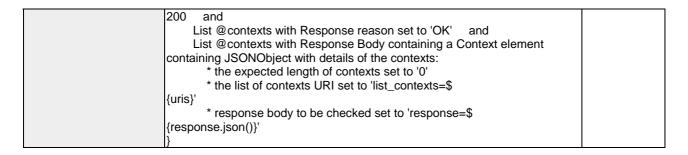
	Query Parameter: kind set to 'Hosted'	
TD Dames (atlantid	TD/NCOLL D/OTY/O/050, 00, 00	
TP Permutation id	TP/NGSI-LD/CTX/Cons/052_02_03	
Name	052_02_03 List @contexts with no details and kind equal to cached and with add @context	previously one
Parent Release	v1.5.1	
Description	Check that one can list @contexts	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}'	
	method set to 'GET'	
	List @contexts: and	
	Query Parameter: details set to '\${EMPTY}' and	
	Query Parameter: kind set to 'Cached'	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	301 -> Client
	List @contexts with Response Status Code set to	
	200 and	
	List @contexts with Response reason set to 'OK' and	
	List @contexts with Check Context Response Body Containing a list of	
	identifiers and	
	Query Parameter: response_body set to 'response.json()' and	
	Query Parameter: expected_length set to '1' and Query Parameter: list_contexts set to 'uri_list' and	
	Query Parameter: list_contexts set to dif_list and Query Parameter: kind set to 'Cached'	
	Query i arameter. Kind set to Cached	
TP Permutation id	TP/NGSI-LD/CTX/Cons/052_02_04	L
Name	052_02_04 List @contexts with no details and kind equal to implicitlycreated a	and with
	previously one add @context	
Parent Release	v1.5.1	
Description	Check that one can list @contexts	Discotion
	Check that one can list @contexts Test events	Direction Client
Description	Check that one can list @contexts Test events when {	Direction SUT <- Client
Description	Check that one can list @contexts Test events when { the SUT receives a Request from the client containing:	
Description	Check that one can list @contexts Test events when {	
Description	Check that one can list @contexts Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}'	
Description	Check that one can list @contexts Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and	
Description	Check that one can list @contexts Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and	
Description	Check that one can list @contexts Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'ImplicitlyCreated' }	SUT <- Client
Description	Check that one can list @contexts Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'ImplicitlyCreated' } then {	
Description	Check that one can list @contexts Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations:	SUT <- Client
Description	Check that one can list @contexts Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to	SUT <- Client
Description	Check that one can list @contexts Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and	SUT <- Client
Description	Check that one can list @contexts Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and	SUT <- Client
Description	Check that one can list @contexts Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and	SUT <- Client
Description	Check that one can list @contexts Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and Query Parameter: response_body set to 'response.json()' and	SUT <- Client
Description	Check that one can list @contexts Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and Query Parameter: response_body set to 'response.json()' and Query Parameter: expected_length set to '0' and	SUT <- Client
Description	Check that one can list @contexts Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and Query Parameter: response_body set to 'response.json()' and Query Parameter: expected_length set to '0' and Query Parameter: list_contexts set to 'uri_list' and	SUT <- Client
Description	Check that one can list @contexts Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and Query Parameter: response_body set to 'response.json()' and Query Parameter: expected_length set to '0' and	SUT <- Client
Description	Check that one can list @contexts Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and Query Parameter: response_body set to 'response.json()' and Query Parameter: expected_length set to '0' and Query Parameter: list_contexts set to 'uri_list' and	SUT <- Client
Description Expected behaviour	Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and Query Parameter: response_body set to 'response.json()' and Query Parameter: expected_length set to '0' and Query Parameter: list_contexts set to 'uri_list' and Query Parameter: kind set to 'ImplicitlyCreated' }	SUT <- Client SUT -> Client
Description Expected behaviour TP Permutation id Name	Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-Id/v1/jsonIdContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and Query Parameter: response_body set to 'response.json()' and Query Parameter: expected_length set to '0' and Query Parameter: list_contexts set to 'uri_list' and Query Parameter: kind set to 'ImplicitlyCreated' } TP/NGSI-LD/CTX/Cons/052_02_05 052_02_05 List @contexts with details equal to false and no kind and with preadd @context	SUT <- Client SUT -> Client
TP Permutation id Name Parent Release	Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and Query Parameter: response_body set to 'response.json()' and Query Parameter: expected_length set to '0' and Query Parameter: list_contexts set to 'uri_list' and Query Parameter: kind set to 'ImplicitlyCreated' } TP/NGSI-LD/CTX/Cons/052_02_05 052_02_05 List @contexts with details equal to false and no kind and with preadd @context v1.5.1	SUT <- Client SUT -> Client
TP Permutation id Name Parent Release Description	Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and	SUT <- Client SUT -> Client eviously one
TP Permutation id Name Parent Release	Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and Query Parameter: response_body set to 'response.json()' and Query Parameter: expected_length set to '0' and Query Parameter: list_contexts set to 'uri_list' and Query Parameter: kind set to 'ImplicitlyCreated' } TP/NGSI-LD/CTX/Cons/052_02_05 052_02_05 List @contexts with details equal to false and no kind and with preadd @context v1.5.1 Check that one can list @contexts Test events	SUT <- Client SUT -> Client Eviously one Direction
TP Permutation id Name Parent Release Description	Check that one can list @contexts Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and Query Parameter: response_body set to 'response.json()' and Query Parameter: expected_length set to '0' and Query Parameter: list_contexts set to 'uri_list' and Query Parameter: kind set to 'ImplicitlyCreated' } TP/NGSI-LD/CTX/Cons/052_02_05 052_02_05 List @contexts with details equal to false and no kind and with preadd @context v1.5.1 Check that one can list @contexts	SUT <- Client SUT -> Client eviously one
TP Permutation id Name Parent Release Description	Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-Id/v1/jsonIdContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and	SUT <- Client SUT -> Client Eviously one Direction
TP Permutation id Name Parent Release Description	Check that one can list @contexts Test events	SUT <- Client SUT -> Client Eviously one Direction
TP Permutation id Name Parent Release Description	Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-Id/v1/jsonIdContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and	SUT <- Client SUT -> Client Eviously one Direction

	Occasion Decreased and Line Location International	T
	Query Parameter: kind set to '\${EMPTY}'	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	301 -> Client
	List @contexts with Response Status Code set to	
	200 and	
	List @contexts with Response reason set to 'OK' and	
	List @contexts with Check Context Response Body Containing a list of	
	identifiers and	
	Query Parameter: response_body set to 'response.json()' and	
	Query Parameter: expected_length set to '2' and	
	Query Parameter: list_contexts set to 'uri_list' and	
	Query Parameter: kind set to '\$	
	{EMPTY}'	
TP Permutation id	TP/NGSI-LD/CTX/Cons/052_02_06	
Name	052_02_06 List @contexts with details equal to false and kind equal to hosted	l and with
Name	previously one add @context	and with
Parent Release	v1.5.1	
Description	Check that one can list @contexts	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}'	
	method set to 'GET'	
	List @contexts: and	
	Query Parameter: details set to 'false' and	
	Query Parameter: kind set to 'Hosted'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	List @contexts with Response Status Code set to	
	200 and	
	List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of	
	identifiers and	
	Query Parameter: response_body set to 'response.json()' and	
	Query Parameter: expected_length set to '1' and	
	Query Parameter: list_contexts set to 'uri_list' and	
	Query Parameter: kind set to 'Hosted'	
	}	
TP Permutation id	TP/NGSI-LD/CTX/Cons/052_02_07	
Name	052_02_07 List @contexts with details equal to false and kind equal to cache	d abd with
	previously one add @context	
Parent Release	v1.5.1	
Description	Check that one can list @contexts	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}'	
	method set to 'GET'	
	List @contexts: and	
	Query Parameter: details set to 'false' and	
	Query Parameter: kind set to 'Cached'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	List @contexts with Response Status Code set to	
	200 and	
	List @contexts with Response reason set to 'OK' and	
	List @contexts with Check Context Response Body Containing a list of	
	identifiers and	
	Query Parameter: response_body set to 'response.json()' and	
	Query Parameter: expected_length set to '1' and	
	Query Parameter: list_contexts set to 'uri_list' and	
	Query Parameter: kind set to 'Cached'	i .
	guery i arameter. Kind set to Gaeried	

TP Permutation id	TP/NGSI-LD/CTX/Cons/052_02_08	
Name	052_02_08 List @contexts with details equal to false and kind equal to implicit with previously one add @context	itlycreated and
Parent Release	v1.5.1	
Description	Check that one can list @contexts	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to 'false' and Query Parameter: kind set to 'ImplicitlyCreated' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and Query Parameter: response_body set to 'response.json()' and Query Parameter: expected_length set to '0' and Query Parameter: list_contexts set to 'uri_list' and Query Parameter: kind set to 'ImplicitlyCreated' }	SUT -> Client

TP Id	TP/NGSI-LD/CTX/Cons/052_05	1
Test objective	Check that one can list all the @context available in the broker with several ac	dd @ssatsyts
rest objective		ad @contexts
Reference	with details equal to true ETSI GS CIM 009 V1.5.1 [1], clause 5.13.3	
Config Id		
	CF_01	
PICS Selection Initial conditions	PICS_LisAllCac@co with { the SUT containing a set of three Hosted @contexts and the default Cached Core	
initial conditions		cned Core
	Context. } Permutations	
TP Permutation id	TP/NGSI-LD/CTX/Cons/052_05_01	
Name	052_05_01 List @contexts with details set to true and no kind and with previo	usly several
	add @contexts	u.s., 2213.u.
Parent Release	v1.5.1	
Description	Check that one can list @contexts	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}'	
	method set to 'GET'	
	List @contexts: and	
	Query Parameter: details set to 'true' and	
	Query Parameter: kind set to '\${EMPTY}'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	List @contexts with Response Status Code set to	
	200 and	
	List @contexts with Response reason set to 'OK' and	
	List @contexts with Response Body containing a Context element	
	containing JSONObject with details of the contexts:	
	* the expected length of contexts set to '4'	
	* the list of contexts URI set to 'list_contexts=\$	
	{uris}'	
	* response body to be checked set to 'response=\$	
	{response.json()}'	
	[]	
TP Permutation id	TP/NGSI-LD/CTX/Cons/052_05_02	
Name	052_05_02 List @contexts with details set to true and kind set to hosted and	with previously
	several add @contexts	

Parent Release	v1.5.1	
Description	Check that one can list @contexts	
Expected behaviour	Test events	Direction
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ngsi-Id/v1/jsonIdContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to 'true' and Query Parameter: kind set to 'Hosted' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Response Body containing a Context element	SUT -> Client
	containing JSONObject with details of the contexts: * the expected length of contexts set to '3' * the list of contexts URI set to 'list_contexts=\$ {uris}' * response body to be checked set to 'response=\$ {response.json()}' }	
TP Permutation id	TP/NGSI-LD/CTX/Cons/052_05_03	
Name	052_05_03 List @contexts with details set to true and kind set to cached a several add @contexts	pa with previously
Parent Release	v1.5.1	
Description	Check that one can list @contexts	
Expected behaviour	Test events	Direction
TP Permutation id	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to 'true' and Query Parameter: kind set to 'Cached' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Response Body containing a Context element containing JSONObject with details of the contexts: * the expected length of contexts set to '1' * the list of contexts URI set to 'list_contexts=\$ {uris}' * response body to be checked set to 'response=\$ {response.json()}' } TP/NGSI-LD/CTX/Cons/052_05_04	SUT -> Client
Name		ereated and with
Ivallie	052_05_04 List @contexts with details set to true and kind set to implicitly previously several add @contexts	neated and With
Parent Release	V1.5.1	
Description	Check that one can list @contexts	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to 'true' and Query Parameter: kind set to 'ImplicitlyCreated' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to	SUT -> Client



TP Id	TP/NGSI-LD/CTX/Cons/052_03	1
Test objective	Check that one can list all the @context available in the broker with several ac	dd @contexts
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.13.3	dd @contexts
Config Id	CF 01	
PICS Selection	PICS_LisAllCac@co	
Initial conditions	with { the SUT containing a set of three Hosted @contexts and the default Ca	chod Coro
	Context. }	ched Core
	Permutations	
TP Permutation id	TP/NGSI-LD/CTX/Cons/052_03_01	
Name	052_03_01 List @contexts with neither details or kind and with previously sev	eral add
Teame	@context	Ciai add
Parent Release	v1.5.1	
Description	Check that one can list @contexts	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	OOT COMOTIC
	URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}'	
	method set to 'GET'	
	List @contexts: and	
	Query Parameter: details set to '\${EMPTY}' and	
	Query Parameter: kind set to '\${EMPTY}'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	List @contexts with Response Status Code set to	
	200 and	
	List @contexts with Response reason set to 'OK' and	
	List @contexts with Check Context Response Body Containing a list of	
	identifiers and	
	Query Parameter: response_body set to 'response.json()' and	
	Query Parameter: expected_length set to '4' and	
	Query Parameter: list_contexts set to 'uri_list' and	
	Query Parameter: kind set to '\$	
	{EMPTY}'	
TD D (TD/N/00/1/ D/07///0 //070 00 00	
TP Permutation id	TP/NGSI-LD/CTX/Cons/052_03_02	
Name	052_03_02 List @contexts with no details and kind equal to hosted and with p	previously
5	several add @context	
Parent Release	V1.5.1	
Description	Check that one can list @contexts	Direction
Expected behaviour	Test events	Direction SUT <- Client
	the SUT receives a Request from the client containing:	SUT <- Client
	URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}'	
	method set to 'GET'	
	List @contexts: and	
	Query Parameter: details set to '\${EMPTY}' and	
	Query Parameter: kind set to 'Hosted'	
	3	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	OUT -> CHEIR
	List @contexts with Response Status Code set to	
	200 and	
	List @contexts with Response reason set to 'OK' and	
	List @contexts with Check Context Response Body Containing a list of	
	List @contexts with Check Context Nesponse body Containing a list of	

Ī	identificate and	
	identifiers and	
	Query Parameter: response_body set to 'response.json()' and	
	Query Parameter: expected_length set to '3' and	
	Query Parameter: list_contexts set to 'uri_list' and	
	Query Parameter: kind set to 'Hosted'	
TP Permutation id	TP/NGSI-LD/CTX/Cons/052_03_03	
Name	052_03_03 List @contexts with no details and kind equal to cached and with several add @context	previously
Parent Release	v1.5.1	
Description	Check that one can list @contexts	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET'	
	List @contexts: and	
	Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'Cached'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and	
	List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of	
	identifiers and	
	Query Parameter: response_body set to 'response.json()' and	
	Query Parameter: expected_length set to '1' and	
	Query Parameter: list_contexts set to 'uri_list' and	
	Query Parameter: kind set to 'Cached'	
TD Downsutation id		
TP Permutation id	TP/NGSI-LD/CTX/Cons/052_03_04	1 20
Name	052_03_04 List @contexts with no details and kind equal to implicitlycreated a	and with
Parent Release	previously several add @context	
Parent Release		
	v1.5.1	
Description	Check that one can list @contexts	
	Check that one can list @contexts Test events	Direction
Description	Check that one can list @contexts Test events when {	Direction SUT <- Client
Description	Check that one can list @contexts Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}'	
Description	Check that one can list @contexts Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET'	
Description	Check that one can list @contexts Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and	
Description	Check that one can list @contexts Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and	
Description	Check that one can list @contexts Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and	
Description	Check that one can list @contexts Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-Id/v1/jsonIdContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'ImplicitlyCreated' }	SUT <- Client
Description	Check that one can list @contexts Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'ImplicitlyCreated' } then {	
Description	Check that one can list @contexts Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations:	SUT <- Client
Description	Check that one can list @contexts Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to	SUT <- Client
Description	Check that one can list @contexts Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and	SUT <- Client
Description	Check that one can list @contexts Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and	SUT <- Client
Description	Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of	SUT <- Client
Description	Check that one can list @contexts Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and	SUT <- Client
Description	Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and Query Parameter: response_body set to 'response.json()' and	SUT <- Client
Description	Check that one can list @contexts Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and Query Parameter: response_body set to 'response.json()' and Query Parameter: expected_length set to '0' and	SUT <- Client
Description	Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and Query Parameter: response_body set to 'response.json()' and Query Parameter: expected_length set to '0' and Query Parameter: list_contexts set to 'uri_list' and	SUT <- Client
Description	Check that one can list @contexts Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and Query Parameter: response_body set to 'response.json()' and Query Parameter: expected_length set to '0' and	SUT <- Client
Description Expected behaviour	Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and Query Parameter: response_body set to 'response.json()' and Query Parameter: expected_length set to '0' and Query Parameter: list_contexts set to 'uri_list' and Query Parameter: kind set to 'ImplicitlyCreated' }	SUT <- Client
Description Expected behaviour TP Permutation id	Check that one can list @contexts Test events	SUT -> Client
Description Expected behaviour	Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and Query Parameter: response_body set to 'response.json()' and Query Parameter: expected_length set to '0' and Query Parameter: list_contexts set to 'uri_list' and Query Parameter: kind set to 'ImplicitlyCreated' } TP/NGSI-LD/CTX/Cons/052_03_05 052_03_05 List @contexts with details equal to false and no kind and with preserved.	SUT -> Client
Description Expected behaviour TP Permutation id Name	Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and Query Parameter: response_body set to 'response.json()' and Query Parameter: expected_length set to '0' and Query Parameter: list_contexts set to 'uri_list' and Query Parameter: kind set to 'ImplicitlyCreated' } TP/NGSI-LD/CTX/Cons/052_03_05 052_03_05 List @contexts with details equal to false and no kind and with preseveral add @context	SUT -> Client
TP Permutation id Name Parent Release	Check that one can list @contexts Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-Id/v1/jsonIdContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and Query Parameter: response_body set to 'response.json()' and Query Parameter: expected_length set to '0' and Query Parameter: list_contexts set to 'uri_list' and Query Parameter: kind set to 'ImplicitlyCreated' } TP/NGSI-LD/CTX/Cons/052_03_05 052_03_05 List @contexts with details equal to false and no kind and with preseveral add @context v1.5.1	SUT -> Client
Description Expected behaviour TP Permutation id Name	Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and Query Parameter: response_body set to 'response.json()' and Query Parameter: expected_length set to '0' and Query Parameter: list_contexts set to 'uri_list' and Query Parameter: kind set to 'ImplicitlyCreated' } TP/NGSI-LD/CTX/Cons/052_03_05 052_03_05 List @contexts with details equal to false and no kind and with preseveral add @context	SUT -> Client

Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	OG 1 4 OHOIR
	URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}'	
	method set to 'GET'	
	List @contexts: and	
	Query Parameter: details set to 'false' and	
	Query Parameter: kind set to '\${EMPTY}'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	List @contexts with Response Status Code set to	
	200 and	
	List @contexts with Response reason set to 'OK' and	
	List @contexts with Check Context Response Body Containing a list of	
	identifiers and	
	Query Parameter: response_body set to 'response.json()' and	
	Query Parameter: expected_length set to '4' and	
	Query Parameter: list_contexts set to 'uri_list' and	
	Query Parameter: kind set to '\$	
	{EMPTY}'	
	1	
TP Permutation id	TP/NGSI-LD/CTX/Cons/052_03_06	
Name	052_03_06 List @contexts with details equal to false and kind equal to hosted	and with
	previously several add @context	
Parent Release	v1.5.1	
Description	Check that one can list @contexts	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}'	
	method set to 'GET'	
	List @contexts: and	
	Query Parameter: details set to 'false' and	
	Query Parameter: kind set to 'Hosted'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	List @contexts with Response Status Code set to	
	200 and	
	List @contexts with Response reason set to 'OK' and	
	List @contexts with Check Context Response Body Containing a list of	
	identifiers and	
	Query Parameter: response_body set to 'response.json()' and	
	Query Parameter: expected_length set to '3' and	
	Query Parameter: list_contexts set to 'uri_list' and	
	Query Parameter: kind set to 'Hosted'	
TD Down of ation in	TD/NCCLLD/CTV/Cons/052 02 07	
TP Permutation id Name	TP/NGSI-LD/CTX/Cons/052_03_07 052_03_07 List @contexts with details equal to false and kind equal to cache	d abd with
Ivaille	previously several add @context	abu willi
Parent Release	v1.5.1	
Description	Check that one can list @contexts	
Expected behaviour	Test events	Direction
Expedica beliavioui	when {	SUT <- Client
	the SUT receives a Request from the client containing:	John V. Ollerit
	URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}'	
	method set to 'GET'	
	List @contexts: and	
	Query Parameter: details set to 'false' and	
	Query Parameter: kind set to 'Cached'	
	The state of the s	1
	}	

	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	List @contexts with Response Status Code set to	
	200 and	
	List @contexts with Response reason set to 'OK' and	
	List @contexts with Check Context Response Body Containing a list of	
	identifiers and	
	Query Parameter: response_body set to 'response.json()' and	
	Query Parameter: expected_length set to '1' and	
	Query Parameter: list_contexts set to 'uri_list' and	
	Query Parameter: kind set to 'Cached'	
	3	
TP Permutation id	TP/NGSI-LD/CTX/Cons/052_03_08	I
Name	052_03_08 List @contexts with details equal to false and kind equal to implici	tlycroated and
Name		llycreated and
D	with previously several add @context	
Parent Release	v1.5.1	
Description	Check that one can list @contexts	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}'	
	method set to 'GET'	
	List @contexts: and	
	List @contexts: and Query Parameter: details set to 'false' and	
	List @contexts: and	
	List @contexts: and Query Parameter: details set to 'false' and Query Parameter: kind set to 'ImplicitlyCreated' }	
	List @contexts: and Query Parameter: details set to 'false' and Query Parameter: kind set to 'ImplicitlyCreated' } then {	SUT -> Client
	List @contexts: and Query Parameter: details set to 'false' and Query Parameter: kind set to 'ImplicitlyCreated' }	SUT -> Client
	List @contexts: and Query Parameter: details set to 'false' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations:	SUT -> Client
	List @contexts: and Query Parameter: details set to 'false' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to	SUT -> Client
	List @contexts: and Query Parameter: details set to 'false' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to and	SUT -> Client
	List @contexts: and Query Parameter: details set to 'false' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to and List @contexts with Response reason set to 'OK' and	SUT -> Client
	List @contexts: and Query Parameter: details set to 'false' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of	SUT -> Client
	List @contexts: and Query Parameter: details set to 'false' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and	SUT -> Client
	List @contexts: and Query Parameter: details set to 'false' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and Query Parameter: response_body set to 'response.json()' and	SUT -> Client
	List @contexts: and Query Parameter: details set to 'false' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and Query Parameter: response_body set to 'response.json()' and Query Parameter: expected_length set to '0' and	SUT -> Client
	List @contexts: and Query Parameter: details set to 'false' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and Query Parameter: response_body set to 'response.json()' and	SUT -> Client
	List @contexts: and Query Parameter: details set to 'false' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and Query Parameter: response_body set to 'response.json()' and Query Parameter: expected_length set to '0' and Query Parameter: list_contexts set to 'uri_list' and	SUT -> Client
	List @contexts: and Query Parameter: details set to 'false' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and Query Parameter: response_body set to 'response.json()' and Query Parameter: expected_length set to '0' and	SUT -> Client

TP ld	TD/NCCLLD/CTV/Cons/052, 04	
	TP/NGSI-LD/CTX/Cons/052_01	
Test objective	Check that one can list all the @context available in the broker with no previous	is add
	@context	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.13.3	
Config Id	CF_01	
PICS Selection	PICS_LisAllCac@co	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CTX/Cons/052_01_01	
Name	052_01_01 List @contexts with neither details or kind and not previously creat	ted @context
Parent Release	v1.5.1	
Description	Check that one can list @contexts	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}'	
	method set to 'GET'	
	List @contexts: and	
	Query Parameter: details set to '\${EMPTY}' and	
	Query Parameter: kind set to '\${EMPTY}'	
	Query r drameter. Kind set to \(\psi_{\text{Livil 1.1}} \)	
	}	SUT -> Client
	then {	SUT -> Client
	then { the SUT sends a valid Response for the operations:	SUT -> Client
	then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to	SUT -> Client
	then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to and	SUT -> Client
	then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to	SUT -> Client

	identifiers and	
	Query Parameter: response_body set to 'response.json()' and Query Parameter: expected_length set to '1'	
	Query Parameter, expected_length set to 1	
TD David College		
TP Permutation id	TP/NGSI-LD/CTX/Cons/052_01_02	
Name	052_01_02 List @contexts with no details and kind equal to hosted and not provide the context of	reviously
	created @context	
Parent Release	v1.5.1	
Description	Check that one can list @contexts	
Expected behaviour	Test events	Direction
-	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}'	
	method set to 'GET'	
	List @contexts: and	
	Query Parameter: details set to '\${EMPTY}' and	
	Query Parameter: kind set to 'Hosted'	
	Read of the section o	
	then {	SUT -> Client
	· ·	SUT -> CITETIL
	the SUT sends a valid Response for the operations:	
	List @contexts with Response Status Code set to	
	200 and	
	List @contexts with Response reason set to 'OK' and	
	List @contexts with Check Context Response Body Containing a list of	
	identifiers and	
	Query Parameter: response_body set to 'response.json()' and	
	Query Parameter: expected_length set to '0'	
	}	
TP Permutation id	TP/NGSI-LD/CTX/Cons/052_01_03	
Name	052_01_03 List @contexts with no details and kind equal to cached and not p	reviously
	created @context	,
Parent Release	v1.5.1	
Description	Check that one can list @contexts	
		Direction
F x nected behaviour	I est events	
Expected behaviour	Test events	
Expected benaviour	when {	SUT <- Client
Expected benaviour	when { the SUT receives a Request from the client containing:	
Expected benaviour	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}'	
Expected benaviour	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET'	
Expected benaviour	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and	
Expected benaviour	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and	
Expected benaviour	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and	
Expected benaviour	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'Cached' }	SUT <- Client
Expected benaviour	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'Cached' } then {	
Expected benaviour	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'Cached' } then { the SUT sends a valid Response for the operations:	SUT <- Client
Expected benaviour	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'Cached' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to	SUT <- Client
Expected benaviour	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'Cached' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and	SUT <- Client
Expected benaviour	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'Cached' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to and List @contexts with Response reason set to 'OK' and	SUT <- Client
Expected benaviour	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'Cached' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and	SUT <- Client
Expected benaviour	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'Cached' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to and List @contexts with Response reason set to 'OK' and	SUT <- Client
Expected benaviour	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'Cached' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of	SUT <- Client
Expected benaviour	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'Cached' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and	SUT <- Client
Expected benaviour	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'Cached' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and Query Parameter: response_body set to 'response.json()' and	SUT <- Client
TP Permutation id	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'Cached' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and Query Parameter: response_body set to 'response.json()' and	SUT <- Client
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-Id/v1/jsonIdContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'Cached' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and Query Parameter: response_body set to 'response.json()' and Query Parameter: expected_length set to '1' } TP/NGSI-LD/CTX/Cons/052_01_04	SUT <- Client SUT -> Client
TP Permutation id	when { the SUT receives a Request from the client containing: URL set to '/ngsi-Id/v1/jsonIdContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'Cached' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and Query Parameter: response_body set to 'response.json()' and Query Parameter: expected_length set to '1' } TP/NGSI-LD/CTX/Cons/052_01_04 052_01_04 List @contexts with no details and kind equal to implicitlycreated and contexts with the client containing and contexts with contexts with contexts and kind equal to implicitlycreated and contexts with contexts with contexts with contexts and kind equal to implicitlycreated and contexts with contexts with contexts with contexts with contexts with contexts with contexts and kind equal to implicitlycreated and contexts with contexts and kind equal to implicitlycreated and contexts with contexts wi	SUT <- Client SUT -> Client
TP Permutation id Name	when { the SUT receives a Request from the client containing: URL set to '/ngsi-Id/v1/jsonIdContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'Cached' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and Query Parameter: response_body set to 'response.json()' and Query Parameter: expected_length set to '1' } TP/NGSI-LD/CTX/Cons/052_01_04	SUT <- Client SUT -> Client
TP Permutation id Name Parent Release	when { the SUT receives a Request from the client containing: URL set to '/ngsi-Id/v1/jsonIdContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'Cached' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and	SUT <- Client SUT -> Client
TP Permutation id Name Parent Release Description	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'Cached' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and Query Parameter: response_body set to 'response.json()' and Query Parameter: expected_length set to '1' } TP/NGSI-LD/CTX/Cons/052_01_04 052_01_04 List @contexts with no details and kind equal to implicitlycreated a previously created @context v1.5.1 Check that one can list @contexts	SUT <- Client SUT -> Client and not
TP Permutation id Name Parent Release	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'Cached' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and	SUT <- Client SUT -> Client and not Direction
TP Permutation id Name Parent Release Description	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'Cached' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and	SUT <- Client SUT -> Client and not
TP Permutation id Name Parent Release Description	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'Cached' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and	SUT <- Client SUT -> Client and not Direction
TP Permutation id Name Parent Release Description	when { the SUT receives a Request from the client containing: URL set to '/ngsi-Id/v1/jsonIdContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'Cached' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and	SUT <- Client SUT -> Client and not Direction
TP Permutation id Name Parent Release Description	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'Cached' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and	SUT <- Client SUT -> Client and not Direction
TP Permutation id Name Parent Release Description	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'Cached' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and	SUT <- Client SUT -> Client and not Direction
TP Permutation id Name Parent Release Description	when { the SUT receives a Request from the client containing: URL set to '/ngsi-Id/v1/jsonIdContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'Cached' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and Query Parameter: response_body set to 'response.json()' and Query Parameter: expected_length set to '1' } TP/NGSI-LD/CTX/Cons/052_01_04 052_01_04 List @contexts with no details and kind equal to implicitlycreated a previously created @context v1.5.1 Check that one can list @contexts	SUT <- Client SUT -> Client and not Direction
TP Permutation id Name Parent Release Description	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'Cached' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and	SUT <- Client SUT -> Client and not Direction
TP Permutation id Name Parent Release Description	when { the SUT receives a Request from the client containing: URL set to '/ngsi-Id/v1/jsonIdContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to '\${EMPTY}' and Query Parameter: kind set to 'Cached' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and Query Parameter: response_body set to 'response.json()' and Query Parameter: expected_length set to '1' } TP/NGSI-LD/CTX/Cons/052_01_04 052_01_04 List @contexts with no details and kind equal to implicitlycreated a previously created @context v1.5.1 Check that one can list @contexts	SUT <- Client SUT -> Client and not Direction

	Tu ,	OUT OF
	then { the SUT sends a valid Response for the operations:	SUT -> Client
	List @contexts with Response Status Code set to	
	200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of	
	identifiers and Query Parameter: response_body set to 'response.json()' and Query Parameter: expected_length set to '0'	
	}	
TP Permutation id	TP/NGSI-LD/CTX/Cons/052_01_05	
Name	052_01_05 List @contexts with details equal to false and no kind and not prevent of the context.	iously created
Parent Release	v1.5.1	
Description	Check that one can list @contexts	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ngsi-Id/v1/jsonIdContexts?\${params_as_string}' method set to 'GET' List @contexts: and	
	Query Parameter: details set to 'false' and Query Parameter: kind set to '\${EMPTY}'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and	
	List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of	
	identifiers and Query Parameter: response_body set to 'response.json()' and Query Parameter: expected_length set to '1'	
	}	
TP Permutation id	TP/NGSI-LD/CTX/Cons/052_01_06	
Name	052_01_06 List @contexts with details equal to false and kind equal to hosted previously created @context	and not
Parent Release	v1.5.1	
Description	Check that one can list @contexts	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-Id/v1/jsonIdContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to 'false' and Query Parameter: kind set to 'Hosted'	SUT <- Client
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to	
	200 and	
	List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and	
	Query Parameter: response_body set to 'response.json()' and Query Parameter: expected_length set to '0'	
TP Permutation id	TP/NGSI-LD/CTX/Cons/052_01_07	1
Name	052_01_07 List @contexts with details equal to false and kind equal to cached previously created @context	d and not
Parent Release	v1.5.1	
Description	Check that one can list @contexts	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}'	

	I	I
	method set to 'GET'	
	List @contexts: and	
	Query Parameter: details set to 'false' and	
	Query Parameter: kind set to 'Cached'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	List @contexts with Response Status Code set to	
	200 and	
	List @contexts with Response reason set to 'OK' and	
	List @contexts with Check Context Response Body Containing a list of	
	lidentifiers and	
	Query Parameter: response_body set to 'response.json()' and	
	Query Parameter: response_body set to response.json() and Query Parameter: expected_length set to '1'	
	Query Parameter, expected_length set to 1	
TP Permutation id]} TP/NGSI-LD/CTX/Cons/052	
Name	052_01_08 List @contexts with details equal to false and kind equal to implici	thuaraatad and
Name		llycreated and
Daniel Dalassa	not previously created @context	
Parent Release	v1.5.1	
Description	Check that one can list @contexts	Discotion
Description Expected behaviour	Test events	Direction
	Test events when {	Direction SUT <- Client
	Test events when { the SUT receives a Request from the client containing:	
	Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}'	
	Test events when { the SUT receives a Request from the client containing:	
	Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and	
	Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and	
	Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to 'false' and	
	Test events when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and	
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-Id/v1/jsonIdContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to 'false' and Query Parameter: kind set to 'ImplicitlyCreated' }	
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-Id/v1/jsonIdContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to 'false' and Query Parameter: kind set to 'ImplicitlyCreated' } then {	SUT <- Client
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-Id/v1/jsonIdContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to 'false' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations:	SUT <- Client
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to 'false' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to	SUT <- Client
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to 'false' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and	SUT <- Client
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to 'false' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to and List @contexts with Response reason set to 'OK' and	SUT <- Client
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to 'false' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of	SUT <- Client
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to 'false' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and	SUT <- Client
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to 'false' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and Query Parameter: response_body set to 'response.json()' and	SUT <- Client
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to 'false' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: List @contexts with Response Status Code set to 200 and List @contexts with Response reason set to 'OK' and List @contexts with Check Context Response Body Containing a list of identifiers and	SUT <- Client

4.1.2 Provision

4.1.2.1 AddContext

		1
TP ld	TP/NGSI-LD/CTX/Prov/050_02	
Test objective	Check that one receives a 400 Bad Request creating a @context if the content	t is incorrect
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.13.2	
Config Id	CF_01	
PICS Selection	PICS_AddUse@coTheInt	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CTX/Prov/050_02_01	
Name	050_02_01 Checking Incorrect Payload	
Parent Release	v1.5.1	
Description	Check that the payload include "@context"	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to /	
	ngsi-ld/v1/jsonldContexts'	
	method set to 'POST'	
	Add a new @context based on the payload defined in the file	
	@context-	
	incorrect.ison	
	1	
	1}	

	than (SUT -> Client
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Add a new @context with Response Status Code set to	
	400 and	
	Add a new @context with Response reason set to 'Bad Request' and	
	Add a new @context with Response Body containing the type 'https://	
	uri.etsi.org/ngsi-ld/errors/	
	InvalidRequest' and Response Body Title is a string and is not	
	\${EMPTY}	
	and Response Body Detail is a string and is not	
	\${EMPTY}	
	and Response Boty Title is not equal to Response Body Detail	
	}	
TP Permutation id	TP/NGSI-LD/CTX/Prov/050_02_02	
Name	050_02_02 Checking Wrong JSON	
Parent Release	v1.5.1	
Description	Check that the payload include "@context"	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/jsonldContexts'	
	method set to 'POST'	
	Add a new @context based on the payload defined in the file	
	@context-	
	wrong-json.json	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Add a new @context with Response Status Code set to	
	400 and	
	Add a new @context with Response reason set to 'Bad Request' and	
	Add a new @context with Response Body containing the type 'https://	
	uri.etsi.org/ngsi-ld/errors/	
	InvalidRequest' and Response Body Title is a string and is not	
	\${EMPTY}	
	and Response Body Detail is a string and is not	
	\${EMPTY}	
	and Response Boty Title is not equal to Response Body Detail	
	Land Responde Body Fillions for equal to Responde Body Botton	
	}	

TP ld	TP/NGSI-LD/CTX/Prov/050_04	
Test objective	Check that one can add a hosted @context with list of URIs and each of them @coxtexts	are cached
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.13.2	
Config Id	CF_01	
PICS Selection	PICS_AddUse@coTheInt	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CTX/Prov/050_04_01	
Name	050_04_01 Add a valid hosted @context with URIs and check that the URIs are Cached @contexts	
Parent Release	v1.5.1	
Description	Check that one can add a @context	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/jsonldContexts' method set to 'POST' Add a new @context based on the payload defined in the file @context- cached-valid.json }	SUT <- Client

then {	SUT -> Client
the SUT sends a valid Response for the operations:	
Add a new @context with Response Status Code set to	
201 and	
Add a new @context with Response does not contain a body and Add a new @context with Response reason set to 'Created' and Serve a @context with Check that the Kind of the @context is set to 'Hosted' and Notification with Check that for each @context in the response body,	
they are of the type 'Cached' }	

TP Id	TD/NCCL D/CTV/Draw/050 04	
	TP/NGSI-LD/CTX/Prov/050_01	
Test objective	Check that one can add a hosted @context	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.13.2	
Config Id	CF_01	
PICS Selection	PICS_AddUse@coTheInt	
Initial conditions	with { the SUT containing an initial state }	
TD Democratical in the	Permutations	
TP Permutation id	TP/NGSI-LD/CTX/Prov/050_01_01	
Name	050_01_01 Add a valid hosted @context from key=value	
Parent Release	V1.5.1	
Description	Check that one can add a @context	T 51 //
Expected behaviour	Test events	Direction SUT <- Client
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/jsonIdContexts' method set to 'POST' Add a new @context based on the payload defined in the file @context- minimal-valid.json } then { the SUT sends a valid Response for the operations: Add a new @context with Response Status Code set to 201 and Add a new @context with Response does not contain a body and Add a new @context with Response reason set to 'Created' and Serve a @context with Check that the Kind of the @context is set to	SUT -> Client
TP Permutation id	'Hosted' } TP/NGSI-LD/CTX/Prov/050_01_02 050_01_02 Add a valid cached @context from URI	
Parent Release	v1.5.1	
Description	Check that one can add a @context	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/jsonldContexts' method set to 'POST' Add a new @context based on the payload defined in the file @context- cached-valid.json }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Add a new @context with Response Status Code set to 201 and Add a new @context with Response does not contain a body and Add a new @context with Response reason set to 'Created' and Serve a @context with Check that the Kind of the @context is set to 'Hosted' }	SUT -> Client

TP ld	TP/NGSI-LD/CTX/Prov/050_03	
Test objective	Check that one can create a implicitlycreated @context through creating a sub	scription
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.13.2	
Config Id	CF_01	
PICS Selection	PICS_AddUse@coTheInt	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CTX/Prov/050_03_01	
Name	050_03_01 Check the creation of ImplicitelyCreted @context	
Parent Release	v1.5.1	
Description	Check that one can create a subscription	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts?\${params_as_string}' method set to 'GET' List @contexts: and Query Parameter: details set to 'true' and Query Parameter: kind set to 'ImplicitlyCreated' } then { the SUT sends a valid Response for the operations: Create Subscription with Response Status Code set to 200 and Notification with Check that the Kind of the @context is set to	SUT <- Client SUT -> Client
	'ImplicitlyCreated' }	

4.1.2.2 DeleteContext

TP Id	TP/NGSI-LD/CTX/Prov/051_09	
Test objective	Check that one gets an error when try to delete the core @context	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.13.5	
Config Id	CF_01	
PICS Selection	PICS_DelOneSpe@coInt	
Initial conditions	with { the SUT containing a core context and it has been deleted with reload s	set to true. }
	Permutations	
TP Permutation id	TP/NGSI-LD/CTX/Prov/051_09_01	
Name	051_09_01 Delete a core @contexts with reload set to true and check if the c	ontext has
	been download again by the broker	
Parent Release	v1.5.1	
Description	Delete a core @contexts with reload set to true and check downloaded core of	context
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	the SUT receives a Request from the client containing: URL set to '/	
	URL set to '/	
	URL set to '/ ngsi-ld/v1/jsonldContexts/{contextId}'	
	URL set to '/ ngsi-ld/v1/jsonldContexts/{contextId}' method set to 'GET'	
	URL set to '/ ngsi-ld/v1/jsonldContexts/{contextId}' method set to 'GET' Serve a @context with contextID 'https://	
	URL set to '/ ngsi-ld/v1/jsonldContexts/{contextId}' method set to 'GET' Serve a @context with contextID 'https:// uri.etsi.org/ngsi-ld/v1/ngsi-ld-core-context-	
	URL set to '/ ngsi-ld/v1/jsonldContexts/{contextId}' method set to 'GET' Serve a @context with contextID 'https:// uri.etsi.org/ngsi-ld/v1/ngsi-ld-core-context-	SUT -> Client
	URL set to '/ ngsi-ld/v1/jsonldContexts/{contextId}' method set to 'GET' Serve a @context with contextID 'https:// uri.etsi.org/ngsi-ld/v1/ngsi-ld-core-context- v1.6.jsonld' and details 'true' }	SUT -> Client
	URL set to '/ ngsi-ld/v1/jsonldContexts/{contextId}' method set to 'GET' Serve a @context with contextID 'https:// uri.etsi.org/ngsi-ld/v1/ngsi-ld-core-context- v1.6.jsonld' and details 'true' } then {	SUT -> Client
	URL set to '/ ngsi-ld/v1/jsonldContexts/{contextId}' method set to 'GET' Serve a @context with contextID 'https:// uri.etsi.org/ngsi-ld/v1/ngsi-ld-core-context- v1.6.jsonld' and details 'true' } then { the SUT sends a valid Response for the operation:	SUT -> Client

TP Id	TP/NGSI-LD/CTX/Prov/051_08
Test objective	Check that one gets an error when try to delete the core @context
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.13.5
Config Id	CF_01
PICS Selection	PICS_DelOneSpe@coInt
Initial conditions	with { the SUT containing an initial state }

	Permutations	
TP Permutation id	TP/NGSI-LD/CTX/Prov/051_08_01	
Name	051_08_01 Delete a core @contexts and return an error with no reload	
Parent Release	v1.5.1	
Description	Check that one gets an error when try to delete the core @context	1
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/jsonldContexts/{contextId}' method set to 'DELETE' Delete a @context whose 'URI' set to 'https:// uri.etsi.org/ngsi-ld/v1/ngsi-ld-core-context-v1.6.jsonld' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Delete a @context with Response Status Code set to 400 and Delete a @context with Response reason set to 'Bad Request' and Delete a @context with Response Body containing the type 'https:// uri.etsi.org/ngsi-Id/errors/ BadRequestData' and Response Body Title is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Title is not equal to Response Body Detail }	SUT -> Client
TP Permutation id	TP/NGSI-LD/CTX/Prov/051_08_02	
Name	051_08_02 Delete a core @contexts and return an error with reload set to false	Se Se
Parent Release	v1.5.1	50
Description	Check that one gets an error when try to delete the core @context	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/jsonldContexts/{contextId}' method set to 'DELETE' Delete a @context whose 'URI' set to 'https:// uri.etsi.org/ngsi-ld/v1/ngsi-ld-core-context-v1.6.jsonld' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Delete a @context with Response Status Code set to 400 and Delete a @context with Response reason set to 'Bad Request' and Delete a @context with Response Body containing the type 'https:// uri.etsi.org/ngsi-Id/errors/ BadRequestData' and Response Body Title is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Title is not equal to Response Body Detail }	SUT -> Client

TP ld	TP/NGSI-LD/CTX/Prov/051_02
Test objective	Check that one gets an error when trying to delete an unknown @context identifier
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.13.5
Config Id	CF_01
PICS Selection	PICS_DelOneSpe@coInt
Initial conditions	with { the SUT containing an initial state }
	Permutations
TP Permutation id	TP/NGSI-LD/CTX/Prov/051_02_01
Name	051_02_01 Delete a @context with unknown @context identifier
Parent Release	v1.5.1
Description	Check that an error message is obtained in the response when one tries to delete a @context with unknonwn id

Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/jsonldContexts/{contextId}'	
	method set to 'DELETE'	
	Delete a @context whose 'URI' set to '\$	
	{random_url}'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Delete a @context with Response Status Code set to	
	404 and	
	Delete a @context with Response reason set to 'Not Found' and	
	Delete a @context with Response Body containing the type '\$	
	{ERROR_TYPE_RESOURCE_NOT_FOUND}' and Response Body Title is a	
	string and is not	
	\${EMPTY}	
	and Response Body Detail is a string and is not	
	\${EMPTY}	
	and Response Boty Title is not equal to Response Body Detail	
	<u> }</u>	

TP Id	TP/NGSI-LD/CTX/Prov/051_04	
Test objective	Check that one gets an error when try to delete a @context	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.13.5	
Config Id	CF_01	
PICS Selection	PICS_DelOneSpe@coInt	
Initial conditions	with { the SUT containing a Hosted @context and the default Cached Core Co	ontext. }
	Permutations	
TP Permutation id	TP/NGSI-LD/CTX/Prov/051_04_01	
Name	051_04_01 Delete a @contexts with a wrong id and reload set to true	
Parent Release	v1.5.1	
Description	Check that one can delete a hosted @context	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/jsonIdContexts/{contextId}' method set to 'DELETE' Delete a @context whose 'URI' set to 'wrong_id_context' } then { the SUT sends a valid Response for the operations: Delete a @context with Response Status Code set to 404 and Delete a @context with Response reason set to 'Not Found' and Delete a @context with Response Body containing the type '\$ {ERROR_TYPE_RESOURCE_NOT_FOUND}' and Response Body Title is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Boty Title is not equal to Response Body Detail }	SUT <- Client SUT -> Client
TP Permutation id	TP/NGSI-LD/CTX/Prov/051_04_02	•
Name	051_04_02 Delete a @contexts with a wrong id and reload set to false	
Parent Release	v1.5.1	
Description	Check that one can delete a hosted @context	•
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/jsonIdContexts/{contextId}' method set to 'DELETE'	SUT <- Client

	Doloto a @contact whose II IDII act to lumana id contact	I
	Delete a @context whose 'URI' set to 'wrong_id_context'	
	then { the SUT sends a valid Response for the operations: Delete a @context with Response Status Code set to	SUT -> Client
	404 and Delete a @context with Response reason set to 'Not Found' and Delete a @context with Response Body containing the type '\$ {ERROR_TYPE_RESOURCE_NOT_FOUND}' and Response Body Title is a	
	string and is not \${EMPTY}	
	and Response Body Detail is a string and is not \${EMPTY} and Response Boty Title is not equal to Response Body Detail	
TP Permutation id	TP/NGSI-LD/CTX/Prov/051_04_03	
Name	051_04_03 Delete a @contexts with a wrong id and no reload value	
Parent Release	v1.5.1	
Description	Check that one can delete a hosted @context	
Expected behaviour	Test events	Direction
•	when {	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/	COT Y CHOTH
	ngsi-ld/v1/jsonldContexts/{contextId}' method set to 'DELETE'	
	Delete a @context whose 'URI' set to 'wrong_id_context'	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	301 -> Client
	Delete a @context with Response Status Code set to	
	404 and Delete a @context with Response reason set to 'Not Found' and	
	Delete a @context with Response Body containing the type '\$	
	{ERROR_TYPE_RESOURCE_NOT_FOUND}' and Response Body Title is a	
	string and is not	
	\${EMPTY}	
	and Response Body Detail is a string and is not \${EMPTY}	
	and Response Boty Title is not equal to Response Body Detail	
TP Permutation id	TP/NGSI-LD/CTX/Prov/051_04_04	
Name	051_04_04 Delete a @contexts with a wrong id and wrong reload value	
Parent Release	v1.5.1	
Description	Check that one can delete a hosted @context	
		Direction
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/	SUT <- Client
	ngsi-ld/v1/jsonldContexts/{contextId}'	
	method set to 'DELETE' Delete a @context whose 'URI' set to 'wrong_id_context'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Delete a @context with Response Status Code set to 404 and	
	Delete a @context with Response reason set to 'Not Found' and	
	Delete a @context with Response Body containing the type '\$	
	[ERROR_TYPE_RESOURCE_NOT_FOUND]' and Response Body Title is a string and is not	
	\${EMPTY}	
	and Response Body Detail is a string and is not	
	\$\{EMPTY\} and Response Boty Title is not equal to Response Body Detail	
	Tana Nooponoo Doly Tillo lo nol oqual lo Nooponoo Douy Dolan	Ì
	}	
TP Permutation id	TP/NGSI-LD/CTX/Prov/051_04_05	
TP Permutation id Name Parent Release	}	

Description	Check that one can delete a hosted @context	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts/{contextId}' method set to 'DELETE' Delete a @context whose 'URI' set to '\${uri}' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Delete a @context with Response Status Code set to 400 and Delete a @context with Response reason set to 'Bad Request' and Delete a @context with Response Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/ BadRequestData' and Response Body Title is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Title is not equal to Response Body Detail }	SUT -> Client

TP Id	TP/NGSI-LD/CTX/Prov/051_06	
Test objective	Delete a @context whose kind is ImplicitlyCreated without reload param	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.13.5	
Config Id	CF_01	
PICS Selection	PICS_DelOneSpe@coInt	
Initial conditions	with { the SUT containing a ImplicitlyCreated @context created from a subscrip	otion query. }
	Permutations	
TP Permutation id	TP/NGSI-LD/CTX/Prov/051_06_01	
Name	051_06_01 Delete a @context whose kind is ImplicitlyCreated without reload p	param
Parent Release	v1.5.1	
Description	Check that one can delete a ImplicitlyCreated @context	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/jsonldContexts/{contextId}' method set to 'DELETE' Delete a @context whose 'URI' set to '\$ {implicit_id}' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Delete a @context with Response Status Code set to 204 and Delete a @context with Response reason set to 'No Content' and Delete a @context with Response does not contain a body }	SUT -> Client

TP ld	TP/NGSI-LD/CTX/Prov/051_01	
Test objective	Check that one can delete a previous created hosted @context without reload param	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.13.5	
Config Id	CF_01	
PICS Selection	PICS_DelOneSpe@coInt	
Initial conditions	with { the SUT containing a Hosted @context and the default Cached Core Context. }	
Permutations		
TP Permutation id	TP/NGSI-LD/CTX/Prov/051_01_01	
Name	051_01_01 Delete a @context whose kind is hosted without reload param	
Parent Release	v1.5.1	
Description	Check that one can delete a hosted @context	

Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/jsonldContexts/{contextId}' method set to 'DELETE'	
	Delete a @context whose 'URI' set to '\${uri}'	
	}	
	then { the SUT sends a valid Response for the operations:	SUT -> Client
	Delete a @context with Response Status Code set to 204 and	
	Delete a @context with Response reason set to 'No Content' and Delete a @context with Response does not contain a body	

TP Id	TP/NGSI-LD/CTX/Prov/051_03	
Test objective	Check that one can delete a previous created cached @context without reload	param
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.13.5	
Config Id	CF_01	
PICS Selection	PICS_DelOneSpe@coInt	
Initial conditions	with { the SUT containing a Cached @context added from a URL. }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CTX/Prov/051_03_01	
Name	051_03_01 Delete a @context whose kind is cached without reload param	
Parent Release	v1.5.1	-
Description	Check that one can delete a cached @context	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/jsonldContexts/{contextId}' method set to 'DELETE' Delete a @context whose 'URI' set to '\$ {contextId}' } then { the SUT sends a valid Response for the operations: Delete a @context with Response Status Code set to	SUT <- Client SUT -> Client
	204 and Delete a @context with Response reason set to 'No Content' and Delete a @context with Response does not contain a body }	

I · ·		1
TP ld	TP/NGSI-LD/CTX/Prov/051_07	
Test objective	Check that one cannot delete a ImplicitlyCreated @context with reload set to t	rue
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.13.5	
Config Id	CF_01	
PICS Selection	PICS_DelOneSpe@coInt	
Initial conditions	with { the SUT containing a ImplicitlyCreated @context created from a subscription	otion query. }
	Permutations	
TP Permutation id	TP/NGSI-LD/CTX/Prov/051_07_01	
Name	051_07_01 Delete a ImplicitlyCreated @contexts with a valid id and reload set to true	
Parent Release	v1.5.1	
Description	Check that one cannot delete a ImplicitlyCreated @context with reload set to t	rue
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/jsonldContexts/{contextId}'	
	method set to 'DELETE'	
	Delete a @context whose 'URI' set to '\$	
	{implicit_id}'	
	}	

1	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Delete a @context with Response Status Code set to	
4	400 and	
	Delete a @context with Response reason set to 'Bad Request' and	
	Delete a @context with Response Body containing the type 'https://	
	uri.etsi.org/ngsi-ld/errors/	
	BadRequestData' and Response Body Title is a string and is not	
	\${EMPTY}	
	and Response Body Detail is a string and is not	
	\${EMPTY}	
	and Response Boty Title is not equal to Response Body Detail	
	}	

TP Id	TP/NGSI-LD/CTX/Prov/051_05	
Test objective	Check that one gets an error if one created an entity with a context (Cached context)	ontext) and
	one tries to delete it with reload=true	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.13.5	
Config Id	CF_01	
PICS Selection	PICS_DelOneSpe@coInt	
Initial conditions	with { the SUT containing a Cached @context created from a entity creation the	rough
	downloading from external server. }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CTX/Prov/051_05_01	
Name	051_05_01 Delete and Reload a Cached @context with no communication wi	th the Context
	Server	
Parent Release	v1.5.1	
Description	Check that one gets an error if one tries to reload a cached context with no co	mmunication
	with the context server	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ngsi-ld/v1/jsonldContexts/{contextId}'	
	method set to 'DELETE'	
	Delete a @context whose 'URI' set to '\${uri}'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Delete a @context with Response Status Code set to	
	503 and	
	Delete a @context with Response reason set to 'Service Unavailable'	
	and	
	Delete a @context with Response Body containing the type 'https://	
	uri.etsi.org/ngsi-ld/errors/	
	LdContextNotAvailable' and Response Body Title is a string and is not \${EMPTY}	
	and Response Body Detail is a string and is not \${EMPTY}	
	and Response Boty Title is not equal to Response Body Detail	

4.2 CommonBehaviours

4.2.1 CommonResponses

4.2.1.1 VerifyNotAcceptableMediaType

TP ld	TP/NGSI-LD/CB/HTTP/049_01
Test objective	Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Accept" header
	does not imply "application/json" nor "application/ld+json"
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 6.3.4
Config Id	CF_01

Initial conditions	PICS Selection	PICS_HTTP_PRECOND	
Permutation id TP/NCSI-LD/CB/HTTP/04g_01_01 Parent Release O49_01_01 Endpoint get /entities/(entityld) Parent Release O49_01_01 Endpoint get /entityle/graphication/spon* (or "or "application/spon" (get /entities/(entityld)) Parent Release O49_01_01 Endpoint get /esourceRegistrations/			
TPPermutation id TP/NCSI-LD/CB/HTTP/049_01_02			
Name	TP Permutation id		
Parent Release Description Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Accept" header does not imply "application/ison" nor "application/de-json" (get /entities/(entityd/)) Expected behaviour When {	•		
Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Accept" header does not imply "application/son" nor "application/de-json" (get /entities/(entityd))			
does not imply "application/json" nor "application/d-json" (get /entities/(entities/continus) when (cept" header
Test events	Decempnen.		
when { the SUT receives a Request from the client containing: URL set to '/ngsi-id/v1/enities/(id)' method set to 'GET' Get Entity Request: and Query Parameter: id set to 'Stentity_id)' and Query Parameter: accept set to 'application/xml' } hen { the SUT sends a valid Response for the operation: Query Entity with Response Status Code set to 406 } TP Permutation id TP/NGSI-LD/CB/HTTP/049_01_02 Name	Expected behaviour		
the SUT receives a Request from the client containing: URL set to 'ngs-lid/n1/entities/(id)' method set to 'GET' Get Entity Request: and Query Parameter: dise to 'S(entity, id)' and Query Parameter: accept set to 'application/xml'] then { the SUT sends a valid Response for the operation: Query Entity with Response Status Code set to 406 } TP Permutation id TP/NGSI-LD/CB/HTTP/049_01_02 Name 040 10 22 Endpoint get /subscriptions/(subscriptionid) Parent Release V1.3.1 Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the 'Accept' header does not imply 'application/son' nor 'application/ld+json' (get /subscriptions/subscriptioned)) Expected behaviour When { the SUT receives a Request from the client containing: URL set to '/ngsi-Id/n1/subscriptions/(sid)' method set to 'GET' Subscription Retrieve with the following data: and Query Parameter: dises to 'S(id)' and Query Parameter: disest to 'S(id)' and Query Parameter: dises to 'S(id)' and Query Par			
URL set to 'ngsi-Id/n/Ientities/(id)' method set to 'GET' Get Entity Request: and Query Parameter: id set to 'S(entity_id)' and Query Entity with Response for the operation: Query Entity with Response Status Code set to 406 TP Permutation id TP/NGSI-LD/CB/HTTP/049 01_02 Name Q49_01_02 Endpoint get /subscriptions/(subscriptions/d) V1.3.1 Description Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the 'Accept' header does not imply 'application/ison' nor 'application/id+json' (get /subscriptions/s/subscriptions/f/d) Expected behaviour When { the SUT receives a Request from the client containing: URL set to 'ngsi-Id/n/1/subscriptions/f/d)' method set to 'GET' Subscription Retrieve with the following data: and Query Parameter: id set to '\$(id)' and Query Parameter: accept set to 'application/xml' } Then { the SUT sends a valid Response for the operation: Retrieve Subscription with Response Status Code set to 406 TP Permutation id			
Gel Entity Request: and Query Parameter: dose to 'S[entity_id]' and Query Parameter: accept set to 'application/xml' } then { the SUT sends a valid Response for the operation: Query Entity with Response Status Code set to 406 } } TPNGSI-LD/CB/HTTP/049_01_02 Name			
Query Parameter: id set to "s[entity_id]" and Query Parameter: accept set to 'application/xml" Tent		method set to 'GET'	
Cuery Parameter: accept set to 'application/xml' then {		Get Entity Request: and	
then { the SUT sends a valid Response for the operation: Query Entity with Response Status Code set to 406 } TP Permutation id TP/NGSI-LD/CB/HTTP/049_01_02 Name 049_01_02 Endpoint get /subscriptions/(subscription) Parent Release 1.3.1 Description Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Accept" header dose not imply "application/json" nor "application/id-pson" (get /subscriptions/(subscriptions/subscriptions/))		Query Parameter: id set to '\${entity_id}' and	
the SUT sends a valid Response for the operation: Query Entity with Response Status Code set to 406 TP Permutation id		Query Parameter: accept set to 'application/xml'	
the SUT sends a valid Response for the operation: Query Entity with Response Status Code set to 406 TP Permutation id		}	
Query Entity with Response Status Code set to 406 1		then {	SUT -> Client
TP Permutation id Name 049_01_02 Endpoint get /subscriptions/(subscriptionId) Parent Release 10.3.1 Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Accept" header does not imply "application/son" nor "application/ld+json" (get /subscriptions/(subscriptions/))		the SUT sends a valid Response for the operation:	
TP Permutation id TP/NGSI-LD/CB/HTTP/049_01_02 Name 049_01_02 Endpoint get /subscriptions/{subscriptionld}} Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Accept" header does not imply "application/spon" nor "application/ld+json" (get /subscriptions/{subscriptions/{subscriptions//subscriptions/subscriptions//subscriptions//subscriptions/subscriptions//subscriptions/subscr		Query Entity with Response Status Code set to	
Name 049 01 02 Endpoint get /subscriptions/(subscriptionId)		406	
Name 049 01 02 Endpoint get /subscriptions/(subscriptionId)		}	
Parent Release V1.3.1 Description Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Accept" header does not imply 'application/json' nor 'application/ld+json' (get 'subscriptions/(subscription			
Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Accept" header does not imply "application/json" nor "application/ld+json" (get /subscriptions/{subscriptions/{subscriptions}}. Test events	Name		
does not imply "application/ son" nor "application/ d+ son" (get /subscriptions/(subscriptions/(subscriptions/)	Parent Release		
Subscriptions/(subscriptions/d) Test events Direction	Description		cept" header
Test events			
when { the SUT receives a Request from the client containing: URL set to '/ngsi-Id/v1/subscriptions/(id)' method set to 'GET' Subscription Retrieve with the following data: and Query Parameter: id set to \$\{id\}' and Query Parameter: accept set to 'application/xml' } } then { the SUT sends a valid Response for the operation: Retrieve Subscription with Response Status Code set to 406 } TP Permutation id TP/NGSI-LD/CB/HTTP/049_01_03 Name			
the SUT receives a Request from the client containing:	Expected behaviour		
URL set to 'ngsi-Id/n'/subscriptions/[id]' method set to 'GET' Subscription Retrieve with the following data: and Query Parameter: id set to '\${id}' and Query Parameter: accept set to 'application/xml' } then { the SUT sends a valid Response for the operation: Retrieve Subscription with Response Status Code set to 406 } TP Permutation id Name 049_01_03 Endpoint get /csourceRegistrations/ Parent Release V1.3.1 Description Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Accept" header does not imply "application/json" nor "application/Id+json" (get /csourceRegistrations/) Expected behaviour Test events When { the SUT receives a Request from the client containing: URL set to 'ngsi-Id/n'/csourceRegistrations' method set to 'GET' Request a Context Source Registration with Return Entity Type set to 'Building' Header['Accept'] set to 'application/xml' } then { the SUT sends a valid Response for the operation: Query Context Source Registrations With Return with Response Status Code set to 406 } TP Permutation id TP/NGSI-LD/CB/HTTP/049_01_04 Name 049_01_04 Endpoint get /csourceSubscriptions/ Parent Release V1.3.1 Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Accept" header			SUT <- Client
method set to 'GET' Subscription Retrieve with the following data: and Query Parameter: id set to '\$(id)' and Query Parameter: accept set to 'application/xml' } then { the SUT sends a valid Response for the operation: Retrieve Subscription with Response Status Code set to 406 } TP Permutation id TP/NGSI-LD/CB/HTTP/049_01_03 Name 049_01_03 Endpoint get /csourceRegistrations/ Parent Release V1.3.1 Expected behaviour When { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/csourceRegistrations' Mend Sur receives a Request from the client containing: URL set to '/ngsi-ld/v1/csourceRegistrations' method set to 'GET' Request a Context Source Registration with Return Entity Type set to 'Building' Header['Accept'] set to 'application/xml' } then { the SUT sends a valid Response for the operation: Query Context Source Registrations With Return with Response Status Code set to 406 } TP Permutation id TP/NGSI-LD/CB/HTTP/049_01_04 Name 049_01_04 Endpoint get /csourceSubscriptions/ Parent Release V1.3.1 Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Accept" header V1.3.1 Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Accept" header			
Subscription Retrieve with the following data: and Query Parameter: id set to '\$f(id)' and Query Parameter: id set to '\$f(id)' and Query Parameter: accept set to 'application/xml' } then { the SUT sends a valid Response for the operation: Retrieve Subscription with Response Status Code set to 406 } TP/NGSI-LD/CB/HTTP/049_01_03 Name 049_01_03 Endpoint get /csourceRegistrations/ Parent Release V1.3.1 Description Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Accept" header does not imply "application/json" nor "application/d+json" (get /csourceRegistrations/) Expected behaviour Test events Direction When { the SUT receives a Request from the client containing: URL set to '/ngsi-Id/v1/csourceRegistrations' method set to 'GET' Request a Context Source Registration with Return Entity Type set to 'Building' Header['Accept'] set to 'application/xml' } then { the SUT sends a valid Response for the operation:			
Query Parameter: id set to '\$\{id\}' and Query Parameter: accept set to 'application/xml' \} \			
Query Parameter: accept set to 'application/xml' } then { the SUT sends a valid Response for the operation: Retrieve Subscription with Response Status Code set to 406 } TP Permutation id			
then { the SUT sends a valid Response for the operation: Retrieve Subscription with Response Status Code set to 406 } TP Permutation id TP/NGSI-LD/CB/HTTP/049_01_03 Name 049_01_03 Endpoint get /csourceRegistrations/ Parent Release v1.3.1 Description Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Accept" header does not imply "application/json" nor "application/ld+json" (get /csourceRegistrations/) Expected behaviour Test events Direction When { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/csourceRegistrations' method set to 'GET' Request a Context Source Registration with Return Entity Type set to 'application/xml' } then { the SUT sends a valid Response for the operation: Query Context Source Registrations With Return with Response Status Code set to 406 } TP Permutation id TP/NGSI-LD/CB/HTTP/049_01_04 Name 049_01_04 Endpoint get /csourceSubscriptions/ Parent Release v1.3.1 Description Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Accept" header			
the SUT sends a valid Response for the operation: Retrieve Subscription with Response Status Code set to 406 } TP Permutation id TP/NGSI-LD/CB/HTTP/049_01_03 Name 049_01_03 Endpoint get /csourceRegistrations/ Parent Release V1.3.1 Description Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Accept" header does not imply "application/json" nor "application/ld+json" (get /csourceRegistrations/) Expected behaviour Test events Direction When { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/csourceRegistrations' method set to 'GET' Request a Context Source Registration with Return Entity Type set to 'Building' Header['Accept'] set to 'application/xml' } then { the SUT sends a valid Response for the operation: Query Context Source Registrations With Return with Response Status Code set to 406 } TP Permutation id TP/NGSI-LD/CB/HTTP/049_01_04 Name 049_01_04 Endpoint get /csourceSubscriptions/ Parent Release V1.3.1 Description Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Accept" header		Query Parameter, accept set to application/xmi	
the SUT sends a valid Response for the operation: Retrieve Subscription with Response Status Code set to 406 } TP Permutation id TP/NGSI-LD/CB/HTTP/049_01_03 Name 049_01_03 Endpoint get /csourceRegistrations/ Parent Release V1.3.1 Description Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Accept" header does not imply "application/json" nor "application/ld+json" (get /csourceRegistrations/) Expected behaviour Test events Direction When { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/csourceRegistrations' method set to 'GET' Request a Context Source Registration with Return Entity Type set to 'Building' Header['Accept'] set to 'application/xml' } then { the SUT sends a valid Response for the operation: Query Context Source Registrations With Return with Response Status Code set to 406 } TP Permutation id TP/NGSI-LD/CB/HTTP/049_01_04 Name 049_01_04 Endpoint get /csourceSubscriptions/ Parent Release V1.3.1 Description Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Accept" header		than (CLIT . Client
Retrieve Subscription with Response Status Code set to 406 } TP/Permutation id TP/NGSI-LD/CB/HTTP/049_01_03 Name 049_01_03 Endpoint get /csourceRegistrations/ Parent Release v1.3.1 Description Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Accept" header does not imply "application/json" nor "application/ld+json" (get /csourceRegistrations/) Expected behaviour Test events When {			SUT -> Client
TP Permutation id TP/NGSI-LD/CB/HTTP/049_01_03 Name O49_01_03 Endpoint get /csourceRegistrations/ Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Accept" header does not imply "application/json" nor "application/ld+json" (get /csourceRegistrations/) Expected behaviour Test events Direction When {			
TP Permutation id TP/NGSI-LD/CB/HTTP/049_01_03 Name 049_01_03 Endpoint get /csourceRegistrations/ V1.3.1 Description Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Accept" header does not imply "application/json" nor "application/ld+json" (get /csourceRegistrations/) Expected behaviour Test events When {			
Name		1	
Name	TP Permutation id	TP/NGSI-I D/CB/HTTP/049_01_03	
Parent Release V1.3.1 Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Accept" header does not imply "application/json" nor "application/ld+json" (get /csourceRegistrations/) Expected behaviour Test events Direction When { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/csourceRegistrations' method set to 'GET' Request a Context Source Registration with Return Entity Type set to 'Building' Header['Accept'] set to 'application/xml' } then { the SUT sends a valid Response for the operation: Query Context Source Registrations With Return with Response Status Code set to 406 } TP Permutation id TP/NGSI-LD/CB/HTTP/049_01_04 Name 049_01_04 Endpoint get /csourceSubscriptions/ Parent Release V1.3.1 Description Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Accept" header			
Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Accept" header does not imply "application/json" nor "application/ld+json" (get /csourceRegistrations/) Expected behaviour Test events Direction			
does not imply "application/json" nor "application/Id+json" (get /csourceRegistrations/) Test events			cept" header
### Test events When { the SUT receives a Request from the client containing: URL set to '/ngsi-Id/v1/csourceRegistrations' method set to 'GET' Request a Context Source Registration with Return Entity Type set to 'Building' Header['Accept'] set to 'application/xml' } then { the SUT sends a valid Response for the operation: Query Context Source Registrations With Return with Response Status Code set to 406 } TP Permutation id TP/NGSI-LD/CB/HTTP/049_01_04 Name 049_01_04 Endpoint get /csourceSubscriptions/ Parent Release v1.3.1 Description Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Accept" header			
when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/csourceRegistrations' method set to 'GET' Request a Context Source Registration with Return Entity Type set to 'Building' Header['Accept'] set to 'application/xml' } then { the SUT sends a valid Response for the operation: Query Context Source Registrations With Return with Response Status Code set to 406 } TP Permutation id TP/NGSI-LD/CB/HTTP/049_01_04 Name 049_01_04 Endpoint get /csourceSubscriptions/ Parent Release V1.3.1 Description Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Accept" header	Expected behaviour		
the SUT receives a Request from the client containing:			
URL set to '/ngsi-Id/v1/csourceRegistrations' method set to 'GET' Request a Context Source Registration with Return Entity Type set to 'Building' Header['Accept'] set to 'application/xml' } then { the SUT sends a valid Response for the operation: Query Context Source Registrations With Return with Response Status Code set to 406 } TP Permutation id TP/NGSI-LD/CB/HTTP/049_01_04 Name 049_01_04 Endpoint get /csourceSubscriptions/ Parent Release v1.3.1 Description Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Accept" header			
method set to 'GET' Request a Context Source Registration with Return Entity Type set to 'Building' Header['Accept'] set to 'application/xml' } then { the SUT sends a valid Response for the operation: Query Context Source Registrations With Return with Response Status Code set to 406 } TP Permutation id TP/NGSI-LD/CB/HTTP/049_01_04 Name 049_01_04 Endpoint get /csourceSubscriptions/ Parent Release v1.3.1 Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Accept" header			
Entity Type set to 'Building' Header['Accept'] set to 'application/xml' } then { the SUT sends a valid Response for the operation: Query Context Source Registrations With Return with Response Status Code set to 406 } TP Permutation id TP/NGSI-LD/CB/HTTP/049_01_04 Name 049_01_04 Endpoint get /csourceSubscriptions/ Parent Release v1.3.1 Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Accept" header			
Header['Accept'] set to 'application/xml' } then {		Request a Context Source Registration with Return	
then { the SUT sends a valid Response for the operation: Query Context Source Registrations With Return with Response Status Code set to 406 } TP Permutation id TP/NGSI-LD/CB/HTTP/049_01_04 Name 049_01_04 Endpoint get /csourceSubscriptions/ Parent Release v1.3.1 Description Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Accept" header		Entity Type set to 'Building'	
the SUT sends a valid Response for the operation: Query Context Source Registrations With Return with Response Status Code set to 406 } TP Permutation id TP/NGSI-LD/CB/HTTP/049_01_04 Name 049_01_04 Endpoint get /csourceSubscriptions/ Parent Release v1.3.1 Description Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Accept" header		Header['Accept'] set to 'application/xml'	
the SUT sends a valid Response for the operation: Query Context Source Registrations With Return with Response Status Code set to 406 } TP Permutation id TP/NGSI-LD/CB/HTTP/049_01_04 Name 049_01_04 Endpoint get /csourceSubscriptions/ Parent Release v1.3.1 Description Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Accept" header		}	
Query Context Source Registrations With Return with Response Status Code set to 406 } TP Permutation id TP/NGSI-LD/CB/HTTP/049_01_04 Name 049_01_04 Endpoint get /csourceSubscriptions/ Parent Release v1.3.1 Description Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Accept" header			SUT -> Client
Code set to 406 } TP Permutation id TP/NGSI-LD/CB/HTTP/049_01_04 Name 049_01_04 Endpoint get /csourceSubscriptions/ Parent Release v1.3.1 Description Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Accept" header			
## 406 } TP Permutation id			
TP Permutation id TP/NGSI-LD/CB/HTTP/049_01_04			
Name 049_01_04 Endpoint get /csourceSubscriptions/ Parent Release v1.3.1 Description Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Accept" header		406	
Name 049_01_04 Endpoint get /csourceSubscriptions/ Parent Release v1.3.1 Description Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Accept" header		}	
Parent Release v1.3.1 Description Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Accept" header	TP Permutation id		
Description Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Accept" header	Name		
	Parent Release		
does not imply "application/json" nor "application/ld+json" (get /csourceSubscriptions/)	Description		
		Idoes not imply "application/json" nor "application/ld+json" (get /csourceSubsci	riptions/)

Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ngsi-ld/v1/csourceSubscriptions'	
	method set to 'GET'	
	Query Context Source Registration Subscriptions and	
	Query Parameter: accept set to 'application/xml'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operation:	
	Query Context Source Registration Subscriptions with Response Status	
	Code set to	
	406	
TP Permutation id]} TP/NGSI-LD/CB/HTTP/049_01_05	
Name	049_01_05 Endpoint get /temporal/entities	
Parent Release	v1.3.1	
Description	Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Acc	cept" header
•	does not imply "application/json" nor "application/ld+json" (get /temporal/entitie	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ngsi-ld/v1/temporal/entities'	
	method set to 'GET'	
	Query Temporal Representation of Entities and	
	Query Parameter: entity_types set to '\${entity_types_to_be_retrieved}'	
	and Company of the second of t	
	Query Parameter: timerel set to 'after' and	
	Query Parameter: timeAt set to '2020-08-01T12:05:00Z' and	
	Query Parameter: accept set to 'application/xml'	
	then {	SUT -> Client
	the SUT sends a valid Response for the operation:	OO1 > OIIOIII
	Query Temporal Representation Of Entities with Response Status	
	Code set to	
		1
	1406	

TP Id	TP/NGSI-LD/CB/HTTP/049_02	
Test objective	Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Ac is "application/geo+json" for operations different than "Retrieve Entity" and "Qu	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 6.3.4	
Config Id	CF_01	
PICS Selection	PICS_HTTP_PRECOND	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CB/HTTP/049_02_01	
Name	049_02_01 Retrieve subscription by id	
Parent Release	v1.3.1	
Description	Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Accept" header is "application/geo+json" for operations different than "Retrieve Entity" and "Query Entity" (get /subscriptions/{subscriptionId})	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/subscriptions/{id}' method set to 'GET' Subscription Retrieve with the following data: and Query Parameter: id set to '\${id}' and Query Parameter: accept set to 'application/geo+json' }	SUT <- Client
	then { the SUT sends a valid Response for the operation: Retrieve Subscription with Response Status Code set to 406	SUT -> Client

TP Permutation id	TP/NGSI-LD/CB/HTTP/049_02_02	
Name	049_02_02 Query temporal entities	
Parent Release	v1.3.1	
Description	Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Acis "application/geo+json" for operations different than "Retrieve Entity" and "Q (get /temporal/entities)	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/temporal/entities' method set to 'GET' Query Temporal Representation of Entities and Query Parameter: entity_types set to '\${entity_types_to_be_retrieved}' and Query Parameter: timerel set to 'after' and Query Parameter: timeAt set to '2020-08-01T12:05:00Z' and Query Parameter: accept set to 'application/geo+json' }	SUT <- Client
	then { the SUT sends a valid Response for the operation: Query Temporal Representation Of Entities with Response Status Code set to 406 }	SUT -> Client
TP Permutation id	TP/NGSI-LD/CB/HTTP/049_02_03	•
Name	049_02_03 Query context source registration	
Parent Release	v1.3.1	
Description	Verify throwing 406 HTTP status code (Not Acceptable Media Type) if the "Ac is "application/geo+json" for operations different than "Retrieve Entity" and "Q (get /csourceRegistrations)	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/csourceRegistrations' method set to 'GET' Request a Context Source Registration with Return Entity Type set to 'Building' Header['Accept'] set to 'application/geo+json' }	SUT <- Client
	then { the SUT sends a valid Response for the operation: Query Context Source Registrations With Return with Response Status Code set to 406 }	SUT -> Client

4.2.1.2 VerifyLdContextNotAvailable

TP ld	TP/NGSI-LD/CB/HTTP/043_01	
Test objective	Verify receiving 503 - LdContextNotAvailable error if remote JSON-LD @conte	ext cannot be
	retrieved	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.2.2	
Config Id	CF_01	
PICS Selection	PICS_IGNORESYSPROPS_AND_SYSPROPS_IFASKED	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CB/HTTP/043_01_01	
Name	043_01_01 Create entity	
Parent Release	v1.3.1	
Description	Verify receiving 503 - LdContextNotAvailable error if remote JSON-LD @conte	ext cannot be
	retrieved (Create entity)	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	

	111/4/ (9) //	1
	ngsi-ld/v1/entities/	
	method set to 'POST'	
	Request Header['Content-	
	Type'] set to 'application/ld+json' and	
	payload defined in file: 'building-	
	unretrievable-context-sample.jsonld'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Create Entity Selecting Content Type with Response Status Code set	
	to	
	503 and	
	Create Entity Selecting Content Type with Response Body containing	
	the type 'https://	
	uri.etsi.org/ngsi-ld/errors/	
	LdContextNotAvailable' and	
	Create Entity Selecting Content Type with Response body containing	
	'title' element	
	}	
TP Permutation id	TP/NGSI-LD/CB/HTTP/043_01_02	
Name	043_01_02 Create subscription	
Parent Release	v1.3.1	
Description	Verify receiving 503 - LdContextNotAvailable error if remote JSON-LD @cont	ext cannot be
Description	retrieved (Create subscription)	ext carriot be
Expected behaviour	Test events	Direction
Expected behaviour		Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/subscriptions/	
	method set to 'POST'	
	Request Header['Content-	
	Type'] set to 'application/ld+json' and	
	payload defined in file: 'subscriptions/	
	subscription-unretrievable-context-sample.jsonld'	
	\	
	then {	SUT -> Client
		301 -> Client
	the SUT sends a valid Response for the operations:	
	Create Subscription with Response Status Code set to	
	503 and	
	Create Subscription with Response Body containing the type 'https://	
	uri.etsi.org/ngsi-ld/errors/	
	LdContextNotAvailable' and	
	Create Subscription with Response body containing 'title' element	
	}	
TP Permutation id	TP/NGSI-LD/CB/HTTP/043_01_03	,
Name	043_01_03 Create Temporal Representation of Entities	
Parent Release	v1.3.1	
	Verify receiving 503 - LdContextNotAvailable error if remote JSON-LD @cont	ovt connot bo
Description		ext carriot be
Francisco de la la calcada de	retrieved (Create Temporal Representation of Entities)	Discotion
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/temporal/entities'	
	method set to 'POST'	
	Create or Update Temporal Representation of Entity Selecting Content	
	Type: and	
	Query Parameter: temporal_entity_representation_id set to '\$	
	{temporal_entity_representation_id}' and	
	Query Parameter: filename set to 'bus-	
	temporal-representation-unretrievable-context-sample.jsonld' and	
	Query Parameter: content_type set to 'application/	
	ld+json' and	
	Query Parameter: accept set to '\$	
	{EMPTY}'	
]	<u> </u>

the art	OUT Oliese
	SUT -> Client
the SUT sends a valid Response for the operations:	
Create Or Update Temporal Representation Of Entity Selecting Content	
Type with Response Status Code set to	
503 and	
Create Or Update Temporal Representation Of Entity Selecting Content	
Type with Response Body containing the type 'https://	
uri.etsi.org/ngsi-ld/errors/	
LdContextNotAvailable' and	
Create Or Update Temporal Representation Of Entity Selecting Content	
Type with Response body containing 'title' element	
}	
Permutation id TP/NGSI-LD/CB/HTTP/043_01_04	
me 043_01_04 Batch entity create	
rent Release v1.3.1	
scription Verify receiving 503 - LdContextNotAvailable error if remote JSON-LD @context	kt cannot be
retrieved (Batch entity create)	
pected behaviour Test events	Direction
	SUT <- Client
the SUT receives a Request from the client containing:	
URL set to '/	
ngsi-ld/v1/entityOperations/create'	
method set to 'POST'	
Request Header['Content-	
Type'] set to 'application/ld+json' and	
payload set to a list of entities to be created	
}	
then {	SUT -> Client
the SUT sends a valid Response for the operations:	
Batch Create Entities with Response Status Code set to	
503 and	
Batch Create Entities with Response Body containing the type 'https://	
uri.etsi.org/ngsi-ld/errors/	
LdContextNotAvailable and	
Batch Create Entities with Response body containing 'title' element	
batch Cleate Entitles with Response body containing title element	
Permutation id TP/NGSI-I D/CB/HTTP/043 01 05	
Permutation id TP/NGSI-LD/CB/HTTP/043_01_05 me 043_01_05 Create context source registration	
me 043_01_05 Create context source registration	
me 043_01_05 Create context source registration rent Release v1.3.1	et cannot he
me 043_01_05 Create context source registration rent Release v1.3.1 scription Verify receiving 503 - LdContextNotAvailable error if remote JSON-LD @context	kt cannot be
me 043_01_05 Create context source registration rent Release v1.3.1 verify receiving 503 - LdContextNotAvailable error if remote JSON-LD @context retrieved (Create context source registration)	
me 043_01_05 Create context source registration rent Release v1.3.1 verify receiving 503 - LdContextNotAvailable error if remote JSON-LD @context retrieved (Create context source registration) pected behaviour Test events	Direction
me 043_01_05 Create context source registration rent Release v1.3.1 Verify receiving 503 - LdContextNotAvailable error if remote JSON-LD @contex retrieved (Create context source registration) pected behaviour Test events when {	
me 043_01_05 Create context source registration rent Release v1.3.1 Verify receiving 503 - LdContextNotAvailable error if remote JSON-LD @contextretrieved (Create context source registration) pected behaviour Test events when { the SUT receives a Request from the client containing:	Direction
me 043_01_05 Create context source registration rent Release v1.3.1 Verify receiving 503 - LdContextNotAvailable error if remote JSON-LD @contextretrieved (Create context source registration) pected behaviour Test events when { the SUT receives a Request from the client containing: URL set to '/	Direction
me 043_01_05 Create context source registration rent Release v1.3.1 Verify receiving 503 - LdContextNotAvailable error if remote JSON-LD @contextretrieved (Create context source registration) pected behaviour Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations'	Direction
me 043_01_05 Create context source registration rent Release v1.3.1 Verify receiving 503 - LdContextNotAvailable error if remote JSON-LD @context retrieved (Create context source registration) pected behaviour Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'POST'	Direction
me 043_01_05 Create context source registration rent Release v1.3.1 Verify receiving 503 - LdContextNotAvailable error if remote JSON-LD @context retrieved (Create context source registration) pected behaviour Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'POST' Request Header['Content-	Direction
me 043_01_05 Create context source registration rent Release v1.3.1 Verify receiving 503 - LdContextNotAvailable error if remote JSON-LD @context retrieved (Create context source registration) pected behaviour Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'POST' Request Header['Content- Type'] set to 'application/ld+json' and	Direction
me 043_01_05 Create context source registration rent Release v1.3.1 Verify receiving 503 - LdContextNotAvailable error if remote JSON-LD @context retrieved (Create context source registration) pected behaviour Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'POST' Request Header['Content-	Direction
rent Release v1.3.1 Verify receiving 503 - LdContextNotAvailable error if remote JSON-LD @context retrieved (Create context source registration) pected behaviour Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'POST' Request Header['Content- Type'] set to 'application/ld+json' and payload defined in file: '\${updated_payload}' }	Direction SUT <- Client
me 043_01_05 Create context source registration v1.3.1 Verify receiving 503 - LdContextNotAvailable error if remote JSON-LD @context retrieved (Create context source registration) pected behaviour Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'POST' Request Header['Content- Type'] set to 'application/ld+json' and payload defined in file: '\${updated_payload}' } then {	Direction
rent Release v1.3.1 Verify receiving 503 - LdContextNotAvailable error if remote JSON-LD @context retrieved (Create context source registration) pected behaviour Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'POST' Request Header['Content- Type'] set to 'application/ld+json' and payload defined in file: '\${updated_payload}' } then { the SUT sends a valid Response for the operation:	Direction SUT <- Client
rent Release v1.3.1 Verify receiving 503 - LdContextNotAvailable error if remote JSON-LD @context retrieved (Create context source registration) pected behaviour Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'POST' Request Header['Content- Type'] set to 'application/ld+json' and payload defined in file: '\${updated_payload}' } then { the SUT sends a valid Response for the operation: Create Context Source Registration With Return with Response Status	Direction SUT <- Client
D43_01_05 Create context source registration	Direction SUT <- Client
rent Release v1.3.1 Verify receiving 503 - LdContextNotAvailable error if remote JSON-LD @context retrieved (Create context source registration) pected behaviour Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'POST' Request Header['Content- Type'] set to 'application/ld+json' and payload defined in file: '\${updated_payload}' } then { the SUT sends a valid Response for the operation: Create Context Source Registration With Return with Response Status	Direction SUT <- Client

4.2.1.3 VerifyGETWithoutAccept

TP Id	TP/NGSI-LD/CB/HTTP/045_01
Test objective	Verify that on a GET HTTP request if nothing is specified on the Accept header,
	"application/json" is assumed
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 6.3.4
Config Id	CF_01
PICS Selection	PICS_HTTP_PRECOND

Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CB/HTTP/045_01_01	
Name	045_01_01 Endpoint /entities/{entityId}	
Parent Release	v1.3.1	
Description	Verify that on a GET HTTP request if nothing is specified on the Accept heade	er,
•	"application/json" is assumed (/entities/{entityId})	•
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	COT CHOIL
	URL set to '/	
	ngsi-ld/v1/entities/{id}'	
	method set to 'GET'	
	Get Entity Request: and	
	Query Parameter: id set to '\$	
	(id)' and	
	Query Parameter: Link set to '<	
	\$https://forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/	
	resources/jsonld-contexts/ngsi-ld-test-suite-	
	compound.jsonld>; rel="http://www.w3.org/ns/json-	
	ld#context";type="application/ld+json" and	
	Query Parameter: accept set to '*/	
	* ¹	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Create Entity Selecting Content Type with Response Status Code set	
	lto	
	201 and	
	Create Entity Selecting Content Type with Response Status Code set	
	lto	
	200 and	
	Create Entity Selecting Content Type with Response Header: Content-	
	Type set to application/	
	json and	
	Notification with Response Header: Link is not Empty	
	}	
TP Permutation id	TP/NGSI-LD/CB/HTTP/045_01_02	1
Name	045_01_02 Endpoint /subscriptions/{subscriptionId}	
Parent Release	v1.3.1	
Description	Verify that on a GET HTTP request if nothing is specified on the Accept heads	\r_
Description		ŧI,
Expected behaviour	"application/json" is assumed (/subscriptions/{subscriptionId})	Direction
Expected behaviour	Test events	
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ngsi-ld/v1/subscriptions/{id}'	
	method set to 'GET'	
	Subscription Retrieve with the following data: and	
	Query Parameter: id set to '\${id}' and	
	Query Parameter: accept set to '*/*'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Create Subscription with Response Status Code set to	
	201 and	
	Create Subscription with Response Status Code set to	
	200 and	
	Create Subscription with Response Header: Content-	
	Type set to application/	
	json and	
	Notification with Response Header: Link is not Empty	
	}	
TP Permutation id	TP/NGSI-LD/CB/HTTP/045_01_03	•
Name	045_01_03 Endpoint /csourceRegistrations/	
Parent Release	v1.3.1	
Description	Verify that on a GET HTTP request if nothing is specified on the Accept heads	<u></u>
Description	"application/json" is assumed (/csourceRegistrations/)	,,
	application/journ to accumed (/occurrence) attailurio/j	

Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ngsi-ld/v1/csourceRegistrations'	
	method set to 'GET'	
	Request a Context Source Registration with Return	
	Entity Type set to 'Vehicle'	
	Header['Accept'] set to '*/*'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Create Context Source Registration With Return with Response Status	
	Code set to	
	201 and	
	Create Context Source Registration With Return with Response Status	
	Code set to	
	200 and	
	Create Context Source Registration With Return with Response	
	Header: Content-	
	Type set to application/	
	json and	
	Notification with Response Header: Link is not Empty	
TP Permutation id	} TD/NCCLLD/CD/UTTD/045_04_04	
Name	TP/NGSI-LD/CB/HTTP/045_01_04	
	045_01_04 Endpoint /temporal/entities	
Parent Release	V1.3.1	\ w
Description	Verify that on a GET HTTP request if nothing is specified on the Accept heade "application/json" is assumed (/temporal/entities)	÷1,
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/temporal/entities'	
	method set to 'GET'	
	Request a Temporal Representation of Entities with Return	
	}	
	} then {	SUT -> Client
	then { the SUT sends a valid Response for the operations:	SUT -> Client
		SUT -> Client
	the SUT sends a valid Response for the operations:	SUT -> Client
	the SUT sends a valid Response for the operations: Create Or Update Temporal Representation Of Entity Selecting Content Type with Response Status Code set to 201 and	SUT -> Client
	the SUT sends a valid Response for the operations: Create Or Update Temporal Representation Of Entity Selecting Content Type with Response Status Code set to 201 and Create Or Update Temporal Representation Of Entity Selecting Content	SUT -> Client
	the SUT sends a valid Response for the operations: Create Or Update Temporal Representation Of Entity Selecting Content Type with Response Status Code set to 201 and Create Or Update Temporal Representation Of Entity Selecting Content Type with Response Status Code set to	SUT -> Client
	the SUT sends a valid Response for the operations: Create Or Update Temporal Representation Of Entity Selecting Content Type with Response Status Code set to 201 and Create Or Update Temporal Representation Of Entity Selecting Content Type with Response Status Code set to 200 and	SUT -> Client
	the SUT sends a valid Response for the operations: Create Or Update Temporal Representation Of Entity Selecting Content Type with Response Status Code set to 201 and Create Or Update Temporal Representation Of Entity Selecting Content Type with Response Status Code set to 200 and Create Or Update Temporal Representation Of Entity Selecting Content	SUT -> Client
	the SUT sends a valid Response for the operations: Create Or Update Temporal Representation Of Entity Selecting Content Type with Response Status Code set to 201 and Create Or Update Temporal Representation Of Entity Selecting Content Type with Response Status Code set to 200 and Create Or Update Temporal Representation Of Entity Selecting Content Type with Response Header: Content-	SUT -> Client
	the SUT sends a valid Response for the operations: Create Or Update Temporal Representation Of Entity Selecting Content Type with Response Status Code set to 201 and Create Or Update Temporal Representation Of Entity Selecting Content Type with Response Status Code set to 200 and Create Or Update Temporal Representation Of Entity Selecting Content Type with Response Header: Content- Type set to application/	SUT -> Client
	the SUT sends a valid Response for the operations: Create Or Update Temporal Representation Of Entity Selecting Content Type with Response Status Code set to 201 and Create Or Update Temporal Representation Of Entity Selecting Content Type with Response Status Code set to 200 and Create Or Update Temporal Representation Of Entity Selecting Content Type with Response Header: Content- Type with Response Header: Content- Type set to application/ json and	SUT -> Client
	the SUT sends a valid Response for the operations: Create Or Update Temporal Representation Of Entity Selecting Content Type with Response Status Code set to 201 and Create Or Update Temporal Representation Of Entity Selecting Content Type with Response Status Code set to 200 and Create Or Update Temporal Representation Of Entity Selecting Content Type with Response Header: Content- Type set to application/	SUT -> Client

4.2.1.4 VerifyUnsupportedMediaType

TP ld	TP/NGSI-LD/CB/HTTP/048_01
Test objective	Verify throwing 415 HTTP status code (Unsupported Media Type) if "Content-Type" header
	is not "application/json" or "application/ld+json"
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 6.3.4
Config Id	CF_01
PICS Selection	PICS_HTTP_PRECOND
Initial conditions	with { the SUT containing an initial state }
	Permutations
TP Permutation id	TP/NGSI-LD/CB/HTTP/048_01_01
Name	048_01_01 Endpoint patch /entities/{entityId}/attrs/{attrId}
Parent Release	v1.3.1
Description	Verify throwing 415 HTTP status code (Unsupported Media Type) if "Content-Type" header
	is not "application/json" or "application/ld+json" (patch /entities/{entityId}/attrs/{attrId})

Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/{entityld}/attrs/{attributeld}'	
	method set to 'PATCH'	
	Request Partial Update Entity Attributes and	
	Query Parameter: entityId set to '\$	
	{entity_id}' and	
	Query Parameter: Attributeld set to 'speed' and Query Parameter: fragment_filename set to 'vehicle-	
	brandname-fragment.jsonId' and	
	Query Parameter: content_type set to 'application/	
	xml	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operation:	
	Partial Update Entity Attributes with Response Status Code set to	
	415	
TD Down of a tile.	}]
TP Permutation id	TP/NGSI-LD/CB/HTTP/048_01_02	
Name Parent Release	048_01_02 Endpoint patch /subscriptions/{subscriptionId}	
Description	v1.3.1 Verify throwing 415 HTTP status code (Unsupported Media Type) if "Content-	Typo" booder
Description	is not "application/json" or "application/ld+json" (patch /subscriptions/{subscriptions/	
Expected behaviour	Test events	Direction
Expected beliavious	when {	SUT <- Client
	the SUT receives a Request from the client containing:	COT CHOIL
	URL set to '/ngsi-ld/v1/subscriptions/{subscription_id}'	
	method set to 'PATCH'	
	Request Update Subscription and	
	Header['Content-Type'] set to 'application/xml' and	
	Payload defined in file 'csourceSubscriptions/fragments/subscription-update-	
	sample.jsonld'	
	<u>}</u>	OLIT OF 1
	then {	SUT -> Client
	the SUT sends a valid Response for the operation: Update Subscription with Response Status Code set to	
	415	
	}	
TP Permutation id	TP/NGSI-LD/CB/HTTP/048_01_03	
Name	048_01_03 Endpoint post /entities/	
Parent Release	v1.3.1	
Description	Verify throwing 415 HTTP status code (Unsupported Media Type) if "Content-	Type" header
	is not "application/json" or "application/ld+json" (post /entities/)	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/' method set to 'POST'	
	Request Header['Content-	
	Type'] set to 'application/xml' and	
	payload defined in file: 'building-simple-	
	attributes-sample.jsonld'	
	}	<u> </u>
	then {	SUT -> Client
	the SUT sends a valid Response for the operation:	
	Create Entity Selecting Content Type with Response Status Code set	
	to	
	415	
TD Downstation is	} TD/NCCLLD/CD/UTTD/040, C4, C4	1
TP Permutation id	TP/NGSI-LD/CB/HTTP/048_01_04	
Name	048_01_04 Endpoint post /subscriptions/	
Parent Release	v1.3.1 Verify throwing 415 HTTP status code (Unsupported Media Type) if "Content-	Typo" boods:
Description		rype neader
	is not "application/json" or "application/ld+json" (post /subscriptions/)	

Expected behaviour	Test events	Direction
Expected beliavious	when {	SUT <- Client
	the SUT receives a Request from the client containing:	OOT < Ollerit
	URL set to '/	
	ngsi-ld/v1/subscriptions/	
	method set to 'POST'	
	Request Header['Content-	
	Type'] set to 'application/xml' and	
	payload defined in file: 'csourceSubscriptions/	
	subscription-sample.jsonId'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operation:	
	Create Subscription with Response Status Code set to	
	415	
	}	
TP Permutation id	TP/NGSI-LD/CB/HTTP/048_01_05	
Name	048_01_05 Endpoint post /entityOperations/create	
Parent Release	v1.3.1	
Description	Verify throwing 415 HTTP status code (Unsupported Media Type) if "Content-	Type" header
	is not "application/json" or "application/ld+json" (post /entityOperations/create)	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entityOperations/create'	
	method set to 'POST'	
	Request Header['Content-	
	Type'] set to 'application/xml' and	
	payload set to a list of entities to be created	
	}	OLIT OF 1
	then {	SUT -> Client
	the SUT sends a valid Response for the operation:	
	Batch Create Entities with Response Status Code set to 415	
	1415	
TP Permutation id	TP/NGSI-LD/CB/HTTP/048_01_06	
Name	048_01_06 Endpoint post /temporal/entities/	
Parent Release	v1.3.1	
Description	Verify throwing 415 HTTP status code (Unsupported Media Type) if "Content-	Tyne" header
Description	is not "application/json" or "application/ld+json" (post /temporal/entities/)	Type Headel
Expected behaviour	Test events	Direction
Expected beliavious	when {	SUT <- Client
	the SUT receives a Request from the client containing:	SOT <= Client
	URL set to '/	
	ngsi-ld/v1/temporal/entities'	
	method set to 'POST'	
	Create or Update Temporal Representation of Entity Selecting Content	
	Type: and	
	Query Parameter: temporal_entity_representation_id set to '\$	
	{temporal_entity_representation_id}' and	
	Query Parameter: filename set to 'vehicle-	
	temporal-representation-sample.jsonld' and	
	Query Parameter: content_type set to 'application/	
	xml' and	
	Query Parameter: accept set to '\$	
	{EMPTY}'	
	· · · · · · · · · · · · · · · · · · ·	
	then {	SUT -> Client
	the SUT sends a valid Response for the operation:	
	Create Or Update Temporal Representation Of Entity Selecting Content	
	Type with Response Status Code set to	
	415	
	}	

4.2.1.5 VerifyMergePatchJson

TP ld	TP/NGSI-LD/CB/HTTP/044_02	
Test objective	Verify that PATCH HTTP requests can be done with "application/merge-patc	n+json" as
	Content-Type	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 6.3.4	
Config Id	CF_01	
PICS Selection	PICS_HTTP_PRECOND	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing an initial Sul	scription
	\${subscription} with an id set to \${subscription_id} }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CB/HTTP/044_02_01	
Name	044_02_01 Endpoint /subscriptions/{subscriptionId}	
Parent Release	v1.3.1	
Description	Verify that PATCH HTTP requests can be done with "application/merge-patc	n+json" as
	Content-Type	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/subscriptions/{subscription_id}'	
	method set to 'PATCH'	
	Request Update Subscription and	
	Header['Link'] contain the context 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and	
	Header['Content-Type'] set to 'application/merge-	
	patch+json' and	
	Payload defined in file 'subscriptions/fragments/	
	subscription-update-sample.json'	
	}	01.17
	then {	SUT -> Client
	the SUT sends a valid Response for the operation:	
	Update Subscription with Response Status Code set to	
	204	
	 }	

		
TP ld	TP/NGSI-LD/CB/HTTP/044_01	
Test objective	Verify that PATCH HTTP requests can be done with "application/merge-pate	ch+json" as
	Content-Type	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 6.3.4	
Config Id	CF_01	
PICS Selection	PICS_HTTP_PRECOND	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing an initial Er	ntity \${entity} with
	an id set to \${entityId} }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CB/HTTP/044_01_01	
Name	044_01_01 endpoint /entities/{entityId}/attrs/{attrId}	
Parent Release	v1.3.1	
Description	Verify that PATCH HTTP requests can be done with "application/merge-pate	ch+json" as
-	Content-Type	•
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}'	
	method set to 'PATCH'	
	Request Partial Update Entity Attributes and	
	Query Parameter: entityId set to '\$	
	{entity_id}' and	
	Query Parameter: AttributeId set to 'brandName' and	
	Query Parameter: fragment_filename set to 'vehicle-	
	brandname-fragment.json' and	
	Query Parameter: content_type set to 'application/	
	merge-patch+json' and	

Query Parameter: context set to 'https://	
forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
jsonld-contexts/ngsi-ld-test-suite-compound.jsonld'	
}	
then {	SUT -> Client
the SUT sends a valid Response for the operation:	
Partial Update Entity Attributes with Response Status Code set to	
204	
}	

4.3 ContextSource

4.3.1 Registration

4.3.1.1 DeleteContextSourceRegistration

TP ld	TP/NGSI-LD/CS/REG/035_03	
Test objective	Check that one cannot delete a context source registration by id if the id is not	t known to the
	system	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.9.4	
Config Id	CF_05	
PICS Selection	PICS_CsoRegDel	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/REG/035_03_01	
Name	035_03_01 Delete a context source registration by id	
Parent Release	v1.3.1	
Description	Check that one cannot delete a context source registration by id if the id is not	t known to the
	system	T = -
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations/{registration_id}' method set to 'DELETE' Delete Context Source Registration with id set to '\$ {registration_id}' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Delete Context Source Registration With Return with Response Status Code set to 404 and Delete Context Source Registration With Return with Response body containing 'title' element }	SUT -> Client

TP Id	TP/NGSI-LD/CS/REG/035_01
Test objective	Check that one can delete a context source registration by id
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.9.4
Config Id	CF_05
PICS Selection	PICS_CsoRegDel
Initial conditions	with { the SUT containing an initial state }
	Permutations
TP Permutation id	TP/NGSI-LD/CS/REG/035_01_01
Name	035_01_01 Delete a context source registration by id
Parent Release	v1.3.1
Description	Check that one can delete a context source registration by id

Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/csourceRegistrations/{registration_id}'	
	method set to 'DELETE'	
	Delete Context Source Registration with id set to '\$	
	{registration_id}'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Create Context Source Registration With Return with Response Status	
	Code set to	
	201 and	
	Delete Context Source Registration With Return with Response Status	
	Code set to	
	204 and	
	Retrieve Context Source Registration with Response Status Code set	
	to	
	\${response1.status_code}	
]}	

TP Id	TP/NGSI-LD/CS/REG/035_02	
Test objective	Check that one cannot delete a context source registration under some condit	ions
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.9.4	
Config Id	CF_05	
PICS Selection	PICS_CsoRegDel	
Initial conditions	with { the SUT being in the "initial state" and the SUT contains a Context Sour	ce
	Registration with id equal to \${context_source_registration_id} and payload se	t to
	\${context_source_registration_payload_file_path}}	
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/REG/035_02_01	
Name	035_02_01 Delete a Context Source Registration if the Id is not present	
Parent Release	v1.3.1	
Description	Check that one cannot delete a context source registration under some condit	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/csourceRegistrations/{registration_id}'	
	method set to 'DELETE'	
	Delete Context Source Registration with id set to '\$	
	{EMPTY}'	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	SUT -> CITETIL
	Delete Context Source Registration With Return with Response Status	
	Code set to	
	405 and	
	Delete Context Source Registration With Return with Response Body	
	containing the type '\$	
	{EMPTY}' and	
	Delete Context Source Registration With Return with Response body	
	containing 'title' element	
	}	
TP Permutation id	TP/NGSI-LD/CS/REG/035_02_02	
Name	035_02_02 Delete a Context Source Registration if the Id is not a valid URI	
Parent Release	v1.3.1	
Description	Check that one cannot delete a context source registration under some condit	ions

Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/csourceRegistrations/{registration_id}'	
	method set to 'DELETE'	
	Delete Context Source Registration with id set to 'invalidURI'	
	than	SUT -> Client
	then { the SUT sends a valid Response for the operations:	SUT-> Client
	Delete Context Source Registration With Return with Response Status	
	Code set to	
	400 and	
	Delete Context Source Registration With Return with Response Body	
	containing the type 'https://	
	uri.etsi.org/ngsi-ld/errors/	
	BadRequestData' and	
	Delete Context Source Registration With Return with Response body	
	containing 'title' element	
	}	

4.3.1.2 UpdateContextSourceRegistration

TP ld	TP/NGSI-LD/CS/REG/034_05	
Test objective	Check that one can update a context source registration by id	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.9.3	
Config Id	CF 05	
PICS Selection	PICS_CsoRegUpd	
Initial conditions		
initial conditions	with { the SUT being in the "initial state" and the SUT contains a Context Source Registration with id equal to \${context_source_registration_id} and payload se	
	\${context_source_registration_payload_file_path} }	1 10
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/REG/034 05 01	
Name	034_05_01 Update a context source registration to never expire	
Parent Release	v1.3.1	
Description	Check that one can update a context source registration by id	
Expected behaviour	Test events	Direction
Expected beliavious	when {	SUT <- Client
	the SUT receives a Request from the client containing:	301 C- Client
	URL set to /	
	ngsi-ld/v1/csourceRegistrations/{registration_id}'	
	method set to 'PATCH'	
	Update Context Source Registration with id set to '\$	
	{registration_id}' and registration update fragment set to '\$	
	{registration_update_fragment}' and content-	
	type set to 'application/json'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Update Context Source Registration With Return with Response Status	
	Code set to	
	204 and	
	Lo i alia	
	Retrieve Context Source Registration with Check that response body does not contain the key '\$.expiresAt'	

TP Id	TP/NGSI-LD/CS/REG/034_03
Test objective	Check that one cannot update a context source registration by id if the id is not known to the
	system
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.9.3
Config Id	CF_05
PICS Selection	PICS_CsoRegUpd
Initial conditions	with { the SUT containing an initial state }

Permutations		
TP Permutation id	TP/NGSI-LD/CS/REG/034_03_01	
Name	034_03_01 Update a context source registration by id if the id is not known to	the system
Parent Release	v1.3.1	
Description	Check that one cannot update a context source registration by id if the id is no	t known to the
Expected behaviour	system Test events	Direction
Expected Bellavious	when {	SUT <- Client
	the SUT receives a Request from the client containing:	SOT < SHELL
	URL set to '/	
	ngsi-ld/v1/csourceRegistrations/{registration_id}'	
	method set to 'PATCH'	
	Update Context Source Registration with id set to '\$	
	{registration_id}' and registration update fragment set to '\$	
	{fragment_with_id}' and content-	
	type set to 'application/ld+json'	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Update Context Source Registration With Return with Response Status	
	Code set to	
	404 and	
	Update Context Source Registration With Return with Response reason set to 'Not Found' and	
	Update Context Source Registration With Return with Response Body containing the type 'https://	
	uri.etsi.org/ngsi-ld/errors/	
	ResourceNotFound' and Response Body Title is a string and is not	
	\${EMPTY}	
	and Response Body Detail is a string and is not \${EMPTY}	
	and Response Boty Title is not equal to Response Body Detail }	

TP Id	TP/NGSI-LD/CS/REG/034_01	
Test objective	Check that one can update a context source registration by id	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.9.3	
Config Id	CF_05	
PICS Selection	PICS_CsoRegUpd	
Initial conditions	with { the SUT being in the "initial state" and the SUT contains a Context Source	се
	Registration with id equal to \${context_source_registration_id} and payload se	t to
	\${context_source_registration_payload_file_path}}	
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/REG/034_01_01	
Name	034_01_01 Update a context source registration by id	
Parent Release	v1.3.1	
Description	Check that one can update a context source registration by id	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to /	
	ngsi-ld/v1/csourceRegistrations/{registration_id}'	
	method set to 'PATCH'	
	Update Context Source Registration with id set to '\$	
	{registration_id}' and registration update fragment set to '\$	
	{registration_update_fragment}' and content-	
	type set to 'application/json'	
]}	

then {	SUT -> Client
the SUT sends a valid Response for the operations:	
Update Context Source Registration With Return with Response Status	
Code set to	
204 and	
Retrieve Context Source Registration with Response Body containing	
the key '['endpoint']', with the value 'http://	
my.new.csource.org:1026'	
 }	

TP Id	TP/NGSI-LD/CS/REG/034_06	
Test objective	Check that one cannot update a context source registration if the ld is not pres	sent
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.9.3	
Config Id	CF_05	
PICS Selection	PICS_CsoRegUpd	
Initial conditions	with { the SUT being in the "initial state" and the SUT contains a Context Source	
	Registration with id equal to \${context_source_registration_id} and payload se	t to
	\${context_source_registration_payload_file_path} }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/REG/034_06_01	
Name	034_06_01 Update a context source registration by id if the ld is not present	
Parent Release	v1.3.1	
Description	Check that one cannot update a context source registration under some condi	
Expected behaviour	Test events	Direction SUT <- Client
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations/{registration_id}' method set to 'PATCH' Update Context Source Registration with id set to '\$ {EMPTY}' and registration update fragment set to '\$ {fragment_with_id}' and content- type set to 'application/ld+json' }	
	then { the SUT sends a valid Response for the operations: Update Context Source Registration With Return with Response Status Code set to 405 and Update Context Source Registration With Return with Response reason set to 'Method Not Allowed' and Update Context Source Registration With Return with Response does not contain a body }	SUT -> Client

TP Id	TP/NGSI-LD/CS/REG/034_04	
Test objective	Check that one cannot update a context source registration under some condit	tions
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.9.3	
Config Id	CF_05	
PICS Selection	PICS_CsoRegUpd	
Initial conditions	with { the SUT being in the "initial state" and the SUT contains a Context Source	e
	Registration with id equal to \${context_source_registration_id} and payload se	t to
	\${context_source_registration_payload_file_path} }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/REG/034_04_01	
Name	034_04_01 Update a context source registration if the request body is invalid	
Parent Release	v1.3.1	
Description	Check that one cannot update a context source registration if the request body is invalid	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/csourceRegistrations/{context_source_registration_id}'	
	method set to 'PATCH'	
	Update Context Source Registration with CSR Id set to '\$	

{registration_id}' and subscription update from file 'context-source-registration-invalid-json-sample.json' }	
then { the SUT sends a valid Response for the operations: Update Context Source Registration From File with Response Status Code set to 400 and Update Context Source Registration From File with Response Body containing the type 'https:// uri.etsi.org/ngsi-Id/errors/ BadRequestData' and Update Context Source Registration From File with Response body containing 'title' element }	SUT -> Client

TP Id TP/NGSI-LD/CS/REG/034_02 Test objective Check that one cannot update a context source registration under some conditions Reference ETSI GS CIM 009 V1.5.1 [1], clause 5.9.3 Config Id CF_05 PICS_CsoRegUpd Initial conditions with { the SUT being in the "initial state" and the SUT contains a Context Source Registration with id equal to \${context_source_registration_id}\$ and payload set to \${context_source_registration_jayload_file_path}}\$ Permutations TP Permutation id TP/NGSI-LD/CS/REG/034_02_01 Name 034_02_01 Update a context source registration by id if the Id is not a valid URI Parent Release v1.3.1 Description Check that one cannot update a context source registration under some conditions Expected behaviour Test events Direction When { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceRegistrations/{registration_id}' method set to 'PATCH' Update Context Source Registration with id set to 'invalidURI' and registration update fragment set to '\$
Reference ETSI GS CIM 009 V1.5.1 [1], clause 5.9.3 Config Id CF_05 PICS Selection PICS_CsoRegUpd with { the SUT being in the "initial state" and the SUT contains a Context Source Registration with id equal to \${context_source_registration_id} and payload set to \${context_source_registration_payload_file_path} } Permutations TP/NGSI-LD/CS/REG/034_02_01 Name 034_02_01 Update a context source registration by id if the Id is not a valid URI Parent Release v1.3.1 Description Check that one cannot update a context source registration under some conditions Expected behaviour Test events Direction When {
Config Id CF_05 PICS Selection PICS_CsoRegUpd with { the SUT being in the "initial state" and the SUT contains a Context Source Registration with id equal to \${context_source_registration_id} and payload set to \${context_source_registration_id} and payload set to \${context_source_registration_payload_file_path} }\$ Permutations TP/NGSI-LD/CS/REG/034_02_01 Name 034_02_01 Update a context source registration by id if the Id is not a valid URI V1.3.1 Description Check that one cannot update a context source registration under some conditions Expected behaviour Test events Direction When {
PICS_CsoRegUpd
with { the SUT being in the "initial state" and the SUT contains a Context Source Registration with id equal to \${context_source_registration_id} and payload set to \${context_source_registration_payload_file_path} }
Registration with id equal to \${context_source_registration_id} and payload set to \${context_source_registration_payload_file_path}}\$ Permutations TP Permutation id TP/NGSI-LD/CS/REG/034_02_01 Name 034_02_01 Update a context source registration by id if the ld is not a valid URI Parent Release v1.3.1 Description Check that one cannot update a context source registration under some conditions Expected behaviour Test events Direction when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations/{registration_id}' method set to 'PATCH' Update Context Source Registration with id set to 'invalidURI' and registration update fragment set to '\$
\$\ \text{source_registration_payload_file_path}\} \ \text{Permutations} \ \text{TP Permutation id} \tag{TP/NGSI-LD/CS/REG/034_02_01} \\ \text{Name} \tag{034_02_01 Update a context source registration by id if the Id is not a valid URI} \\ \text{Parent Release} \tag{v1.3.1} \\ \text{Description} \tag{Check that one cannot update a context source registration under some conditions} \\ \text{Expected behaviour} \tag{Test events} \tag{Direction} \\ \text{When } \{ \tag{SUT <- Clieflet} \\ \text{URL set to '/ ngsi-Id/v1/csourceRegistrations/{registration_id}' \\ \text{method set to 'PATCH' Update Context Source Registration with id set to 'invalidURI' and registration update fragment set to '\}
TP Permutation id TP/NGSI-LD/CS/REG/034_02_01 Name 034_02_01 Update a context source registration by id if the ld is not a valid URI Parent Release v1.3.1 Description Check that one cannot update a context source registration under some conditions Expected behaviour Test events Direction when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations/{registration_id}' method set to 'PATCH' Update Context Source Registration with id set to 'invalidURI' and registration update fragment set to '\$
TP Permutation id TP/NGSI-LD/CS/REG/034_02_01 034_02_01 Update a context source registration by id if the Id is not a valid URI Parent Release v1.3.1 Check that one cannot update a context source registration under some conditions Expected behaviour Test events When { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceRegistrations/{registration_id}' method set to 'PATCH' Update Context Source Registration with id set to 'invalidURI' and registration update fragment set to '\$
Name 034_02_01 Update a context source registration by id if the ld is not a valid URI
Parent Release v1.3.1 Description Check that one cannot update a context source registration under some conditions Expected behaviour Test events Direction When {
Check that one cannot update a context source registration under some conditions
Expected behaviour Test events When { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations/{registration_id}' method set to 'PATCH' Update Context Source Registration with id set to 'invalidURI' and registration update fragment set to '\$
when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations/{registration_id}' method set to 'PATCH' Update Context Source Registration with id set to 'invalidURI' and registration update fragment set to '\$
the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations/{registration_id}' method set to 'PATCH' Update Context Source Registration with id set to 'invalidURI' and registration update fragment set to '\$
URL set to '/ ngsi-ld/v1/csourceRegistrations/{registration_id}' method set to 'PATCH' Update Context Source Registration with id set to 'invalidURI' and registration update fragment set to '\$
ngsi-ld/v1/csourceRegistrations/{registration_id}' method set to 'PATCH' Update Context Source Registration with id set to 'invalidURI' and registration update fragment set to '\$
method set to 'PATCH' Update Context Source Registration with id set to 'invalidURI' and registration update fragment set to '\$
Update Context Source Registration with id set to 'invalidURI' and registration update fragment set to '\$
registration update fragment set to '\$
{fragment_with_id}' and content-
type set to 'application/ld+json'
}
then { SUT -> Clie
the SUT sends a valid Response for the operations:
Update Context Source Registration With Return with Response Status
Code set to
400 and
Update Context Source Registration With Return with Response reason
set to 'Bad Request' and
Update Context Source Registration With Return with Response Body
containing the type 'https://
uri.etsi.org/ngsi-ld/errors/
BadRequestData' and Response Body Title is a string and is not
\${EMPTY}
and Response Body Detail is a string and is not
\${EMPTY}
and Response Boty Title is not equal to Response Body Detail
}
TP Permutation id TP/NGSI-LD/CS/REG/034_02_02
Name 034_02_02 Update a context source registration if the request body is not of the same data
type
Parent Release v1.3.1
Description Check that one cannot update a context source registration under some conditions
Expected behaviour Test events Direction
when { SUT <- Clie
the SUT receives a Request from the client containing:
URL set to '/
ngsi-ld/v1/csourceRegistrations/{registration_id}'

		1
	method set to 'PATCH'	
	Update Context Source Registration with id set to '\$	
	{valid_registration_id}' and registration update fragment set to '\$	
	{fragment_with_id}' and content-	
	type set to 'application/ld+json'	
	\frac{1}{2}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Update Context Source Registration With Return with Response Status	
	Code set to	
	400 and	
	Update Context Source Registration With Return with Response reason	
	Update Context Source Registration With Return with Response Body	
	containing the type 'https://	
	uri.etsi.org/ngsi-ld/errors/	
	BadRequestData' and Response Body Title is a string and is not	
	\${EMPTY}	
	and Response Body Detail is a string and is not	
	\${EMPTY}	
	and Response Boty Title is not equal to Response Body Detail	
	}	
TP Permutation id	TP/NGSI-LD/CS/REG/034_02_03	
Name	034_02_03 Update a context source registration if one attempts to remove a n	nandatory
	property	
Parent Release	v1.3.1	
Description	Check that one cannot update a context source registration under some condi	tions
Expected behaviour	Test events	Direction
Expected beliavious	lest events	
Expected beliaviour	when {	SUT <- Client
Expected beliavious	when {	
Expected beliavious		
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/	
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations/{registration_id}'	
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations/{registration_id}' method set to 'PATCH'	
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceRegistrations/{registration_id}' method set to 'PATCH' Update Context Source Registration with id set to '\$	
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations/{registration_id}' method set to 'PATCH' Update Context Source Registration with id set to '\$ {valid_registration_id}' and registration update fragment set to '\$	
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations/{registration_id}' method set to 'PATCH' Update Context Source Registration with id set to '\$ {valid_registration_id}' and registration update fragment set to '\$ {fragment_with_id}' and content-	
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations/{registration_id}' method set to 'PATCH' Update Context Source Registration with id set to '\$ {valid_registration_id}' and registration update fragment set to '\$	
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations/{registration_id}' method set to 'PATCH' Update Context Source Registration with id set to '\$ {valid_registration_id}' and registration update fragment set to '\$ {fragment_with_id}' and content- type set to 'application/ld+json' }	SUT <- Client
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations/{registration_id}' method set to 'PATCH' Update Context Source Registration with id set to '\$ {valid_registration_id}' and registration update fragment set to '\$ {fragment_with_id}' and content- type set to 'application/ld+json' } then {	
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations/{registration_id}' method set to 'PATCH' Update Context Source Registration with id set to '\$ {valid_registration_id}' and registration update fragment set to '\$ {fragment_with_id}' and content- type set to 'application/ld+json' } then { the SUT sends a valid Response for the operations:	SUT <- Client
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations/{registration_id}' method set to 'PATCH' Update Context Source Registration with id set to '\$ {valid_registration_id}' and registration update fragment set to '\$ {fragment_with_id}' and content- type set to 'application/Id+json' } then { the SUT sends a valid Response for the operations: Update Context Source Registration With Return with Response Status	SUT <- Client
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations/{registration_id}' method set to 'PATCH' Update Context Source Registration with id set to '\$ {valid_registration_id}' and registration update fragment set to '\$ {fragment_with_id}' and content- type set to 'application/Id+json' } then { the SUT sends a valid Response for the operations: Update Context Source Registration With Return with Response Status Code set to	SUT <- Client
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations/{registration_id}' method set to 'PATCH' Update Context Source Registration with id set to '\$ {valid_registration_id}' and registration update fragment set to '\$ {fragment_with_id}' and content- type set to 'application/ld+json' } then { the SUT sends a valid Response for the operations: Update Context Source Registration With Return with Response Status Code set to 400 and	SUT <- Client
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations/{registration_id}' method set to 'PATCH' Update Context Source Registration with id set to '\$ {valid_registration_id}' and registration update fragment set to '\$ {fragment_with_id}' and content- type set to 'application/Id+json' } then { the SUT sends a valid Response for the operations: Update Context Source Registration With Return with Response Status Code set to 400 and Update Context Source Registration With Return with Response reason	SUT <- Client
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations/{registration_id}' method set to 'PATCH' Update Context Source Registration with id set to '\$ {valid_registration_id}' and registration update fragment set to '\$ {fragment_with_id}' and content- type set to 'application/Id+json' } then { the SUT sends a valid Response for the operations: Update Context Source Registration With Return with Response Status Code set to 400 and Update Context Source Registration With Return with Response reason set to 'Bad Request' and	SUT <- Client
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations/{registration_id}' method set to 'PATCH' Update Context Source Registration with id set to '\$ {valid_registration_id}' and registration update fragment set to '\$ {fragment_with_id}' and content- type set to 'application/ld+json' } then { the SUT sends a valid Response for the operations: Update Context Source Registration With Return with Response Status Code set to 400 and Update Context Source Registration With Return with Response reason set to 'Bad Request' and Update Context Source Registration With Return with Response Body	SUT <- Client
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations/{registration_id}' method set to 'PATCH' Update Context Source Registration with id set to '\$ {valid_registration_id}' and registration update fragment set to '\$ {fragment_with_id}' and content- type set to 'application/ld+json' } then { the SUT sends a valid Response for the operations: Update Context Source Registration With Return with Response Status Code set to 400 and Update Context Source Registration With Return with Response reason set to 'Bad Request' and Update Context Source Registration With Return with Response Body containing the type 'https://	SUT <- Client
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations/{registration_id}' method set to 'PATCH' Update Context Source Registration with id set to '\$ {valid_registration_id}' and registration update fragment set to '\$ {fragment_with_id}' and content- type set to 'application/ld+json' } then { the SUT sends a valid Response for the operations: Update Context Source Registration With Return with Response Status Code set to 400 and Update Context Source Registration With Return with Response reason set to 'Bad Request' and Update Context Source Registration With Return with Response Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/	SUT <- Client
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations/{registration_id}' method set to 'PATCH' Update Context Source Registration with id set to '\$ {valid_registration_id}' and registration update fragment set to '\$ {fragment_with_id}' and content- type set to 'application/ld+json' } then { the SUT sends a valid Response for the operations: Update Context Source Registration With Return with Response Status Code set to 400 and Update Context Source Registration With Return with Response reason set to 'Bad Request' and Update Context Source Registration With Return with Response Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/ BadRequestData' and Response Body Title is a string and is not	SUT <- Client
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations/{registration_id}' method set to 'PATCH' Update Context Source Registration with id set to '\$ {valid_registration_id}' and registration update fragment set to '\$ {fragment_with_id}' and content- type set to 'application/ld+json' } then { the SUT sends a valid Response for the operations: Update Context Source Registration With Return with Response Status Code set to 400 and Update Context Source Registration With Return with Response reason set to 'Bad Request' and Update Context Source Registration With Return with Response Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/ BadRequestData' and Response Body Title is a string and is not \${EMPTY}	SUT <- Client
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations/{registration_id}' method set to 'PATCH' Update Context Source Registration with id set to '\$ {valid_registration_id}' and registration update fragment set to '\$ {fragment_with_id}' and content- type set to 'application/ld+json' } then { the SUT sends a valid Response for the operations: Update Context Source Registration With Return with Response Status Code set to 400 and Update Context Source Registration With Return with Response reason set to 'Bad Request' and Update Context Source Registration With Return with Response Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/ BadRequestData' and Response Body Title is a string and is not \${EMPTY} and Response Body Detail is a string and is not	SUT <- Client
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceRegistrations/{registration_id}' method set to 'PATCH' Update Context Source Registration with id set to '\$ {valid_registration_id}' and registration update fragment set to '\$ {fragment_with_id}' and content-type set to 'application/Id+json' } then { the SUT sends a valid Response for the operations: Update Context Source Registration With Return with Response Status Code set to 400 and	SUT <- Client
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations/{registration_id}' method set to 'PATCH' Update Context Source Registration with id set to '\$ {valid_registration_id}' and registration update fragment set to '\$ {fragment_with_id}' and content- type set to 'application/ld+json' } then { the SUT sends a valid Response for the operations: Update Context Source Registration With Return with Response Status Code set to 400 and Update Context Source Registration With Return with Response reason set to 'Bad Request' and Update Context Source Registration With Return with Response Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/ BadRequestData' and Response Body Title is a string and is not \${EMPTY} and Response Body Detail is a string and is not	SUT <- Client
Expected Deliavious	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceRegistrations/{registration_id}' method set to 'PATCH' Update Context Source Registration with id set to '\$ {valid_registration_id}' and registration update fragment set to '\$ {fragment_with_id}' and content-type set to 'application/Id+json' } then { the SUT sends a valid Response for the operations: Update Context Source Registration With Return with Response Status Code set to 400 and	SUT <- Client

4.3.1.3 RegisterContextSource

TP Id	TP/NGSI-LD/CS/CSR/033_02
Test objective	Check that one cannot create a context source with invalid content
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.9.2
Config Id	CF_05
	PICS_CsoRegCre
Initial conditions	with { the SUT containing an initial state }
Permutations	
TP Permutation id	TP/NGSI-LD/CS/CSR/033_02_01
Name	033_02_01 Create a context source registration with invalid JSON file

Parent Release	v1.3.1	
Description	Create a context source registration with invalid JSON file	
Expected behaviour	Test events	Direction
Expected bellaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'POST' Create Context Source Registration Request with Content- Type set to 'application/ Id+json' and payload set to '\$ {subscription_payload}' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Create Context Source Registration with Response Status Code set to 400 and Create Context Source Registration with Response Body containing the type 'https:// uri.etsi.org/ngsi-Id/errors/ BadRequestData' and Create Context Source Registration with Response body containing 'title' element }	SUT -> Client

TP Id	TP/NGSI-LD/CS/CSR/033_01_02	
Test objective	Check that one can create a context source registration that never expires	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.9.2	
Config Id	CF_05	
PICS Selection	PICS_CsoRegCre	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/CSR/033_01_02	
Name	033_01_02 Create Context Source Registration That Never Expires	
Parent Release	v1.3.1	
Description	Check that one can create a context source registration that never expires	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'POST' Request Header['Content- Type'] set to 'application/ld+json' and payload defined in file: '\$ {registration_payload}' } then { the SUT sends a valid Response for the operations: Create Context Source Registration With Return with Response Status Code set to 201 and Create Context Source Registration With Return with Response Header: Location containing	SUT <- Client SUT -> Client
	<pre>\$\{\text{registration_id}\} and Retrieve Context Source Registration with Check Created Resource Set To and Query Parameter: 'ignored_keys' set to '\$ {\text{None}\}' list of keys }</pre>	

TP Id	TP/NGSI-LD/CS/CSR/033_06
•	Check that an HTTP error response of type BadRequestData is raised if the Content-Type header is "application/json" and the request payload body (as JSON) contains a "@context" term

Reference	ETSI GS CIM 009 V1.5.1 [1], clause 6.3.5	
Config Id	CF_05	
PICS Selection	PICS_HTTP_CONTEXT_RESOLV	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/CSR/033_06_01	
Name	033_06_01 Create one context source registration containing a JSON-LD @c JSON content type	ontext with a
Parent Release	v1.3.1	
Description	Check that an HTTP error response of type BadRequestData is raised if the Content-Type header is "application/json" and the request payload body (as JSON) contains a "@context" term	
Expected behaviour	Test events	Direction
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'POST' Request Header['Content- Type'] set to 'application/ld+json' and payload defined in file: '\${updated_payload}' }	
	then { the SUT sends a valid Response for the operations: Create Context Source Registration With Return with Response Status Code set to 400 and Create Context Source Registration With Return with Response Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/ BadRequestData' and Create Context Source Registration With Return with Response body containing 'title' element }	SUT -> Client

TP ld	TP/NGSI-LD/CS/CSR/033_04	
Test objective	Check that the @context is obtained from a Link Header if the Content-Type h	neader is
	"application/json"	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 6.3.5	
Config Id	CF_05	
PICS Selection	PICS_HTTP_CONTEXT_RESOLV	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/CSR/033_04_01	
Name	033_04_01 Create one context source registration using a provided Link head	ler with JSON
	content type with Context	
Parent Release	v1.3.1	
Description	Check that the @context is obtained from a Link Header if the Content-Type h	neader is
	"application/json"	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/csourceRegistrations'	
	method set to 'POST'	
	Request Header['Content-	
	Type'] set to 'application/ld+json' and	
	payload defined in file: '\${updated_payload}'	
	than (SUT -> Client
	then {	SUT -> Client
	the SUT sends a valid Response for the operations: Create Context Source Registration With Return with Response Status	
	Code set to	
	201 and	
	Retrieve Context Source Registration with Response Body containing	
	the key '['information']	

	[0]['entities'][0]	
	['type']', with the value 'Building'	
	}	
TP Permutation id	TP/NGSI-LD/CS/CSR/033_04_02	
Name	033_04_02 Create one context source registration using a provided Link head	der with JSON
	content type without Context	
Parent Release	v1.3.1	
Description	Check that the @context is obtained from a Link Header if the Content-Type h	neader is
-	"application/json"	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/csourceRegistrations'	
	method set to 'POST'	
	Request Header['Content-	
	Type'] set to 'application/ld+json' and	
	payload defined in file: '\${updated_payload}'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Create Context Source Registration With Return with Response Status	
	Code set to	
	201 and	
	Retrieve Context Source Registration with Response Body containing	
	the key '['information']	
	[0]['entities'][0]	
	['type']', with the value 'https://ngsi-ld-test-	
	suite/context#Building'	
	}	

TP Id	TP/NGSI-LD/CS/CSR/033 01 03	
Test objective	Check that one can create a context source registration without specifying an I	ID
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.9.2	
Config Id	CF_05	
PICS Selection	PICS_CsoRegCre	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/CSR/033_01_03	
Name	033_01_03 Create Context Source Registration Without Specifying an ID	
Parent Release	v1.3.1	
Description	Check that one can create a context source registration without specifying an	
Expected behaviour	Test events	Direction
	the SUT receives a Request from the client containing:	SUT -> Client
	201 and Retrieve Context Source Registration with Check Created Resource Set To and Query Parameter: 'ignored_keys' set to '\$ {None}' list of keys }	

TP ld	TP/NGSI-LD/CS/CSR/033_03	
Test objective	Check that one cannot create a context source registration that already exists	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.9.2	
Config Id	CF_05	
PICS Selection	PICS_CsoRegCre	
Initial conditions	with { the SUT being in the "initial state" and the SUT contains a Context Source	
	Registration with id equal to \${context_source_registration_id} and payload se	t to
	\${context_source_registration_payload_file_path} }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/CSR/033_03_01	
Name	033_03_01 Create a context source registration that already exists	
Parent Release	v1.3.1	
Description	Check that one cannot create a context source registration that already exists	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'POST' Request Header['Content- Type'] set to 'application/ld+json' and payload defined in file: '\${updated_payload}' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Create Context Source Registration With Return with Response Status Code set to 409 and Create Context Source Registration With Return with Response body containing 'title' element }	SUT -> Client

TP Id	TP/NGSI-LD/CS/CSR/033_01_01	
Test objective	Check that one can create a context source registration with specific ID and ex	piration date
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.9.2	•
Config Id	CF_05	
PICS Selection	PICS_CsoRegCre	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/CSR/033_01_01	
Name	033_01_01 Create Context Source Registration With Specific Date Expiration	Date
Parent Release	v1.3.1	
Description	Check that one can create a context source registration with specific ID and ex	
Expected behaviour	Test events	Direction SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceRegistrations' method set to 'POST' Request Header['Content- Type'] set to 'application/Id+json' and payload defined in file: '\$ {registration_payload}' } then { the SUT sends a valid Response for the operations:	SUT -> Client
	Create Context Source Registration With Return with Response Status Code set to 201 and Retrieve Context Source Registration with Check Created Resource Set To and Query Parameter: 'ignored_keys' set to '\$ {None}' list of keys }	

TP Id	TP/NGSI-LD/CS/CSR/033_05	
Test objective	Check that the default @context is used if the Content-Type header is "applicated by the Content Conte	ation/json" and
Deference	the Link header does not contain a JSON-LD @context	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 6.3.5	
Config Id PICS Selection	CF_05 PICS_HTTP_CONTEXT_RESOLV	
Initial conditions	with { the SUT containing an initial state }	
initial conditions	Permutations	
TP Permutation id	TP/NGSI-LD/CS/CSR/033_05_01	
Name	033_05_01 Create one context source registration using the default context w content type with Context	ith JSON
Parent Release	v1.3.1	
Description	Check that the default @context is used if the Content-Type header is "applicate the Link header does not contain a JSON-LD @context and retrieve the informal ngsild context."	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'POST' Request Header['Content- Type'] set to 'application/ld+json' and payload defined in file: '\${updated_payload}' } then { the SUT sends a valid Response for the operations: Create Context Source Registration With Return with Response Status	SUT <- Client SUT -> Client
TP Permutation id	Code set to 201 and Retrieve Context Source Registration with Response Body containing the key '['information'] [0]['entities'][0]['type']', with the value 'ngsi- ld:default-context/Building' } TP/NGSI-LD/CS/CSR/033 05 02	
Name		ith ISON
	033_05_02 Create one context source registration using the default context w content type without Context	ith JSON
Parent Release	v1.3.1	
Description	Check that the default @context is used if the Content-Type header is "applicate the Link header does not contain a JSON-LD @context and retrieve the informal ngsild context."	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'POST' Request Header['Content- Type'] set to 'application/ld+json' and payload defined in file: '\${updated_payload}' } then {	SUT <- Client SUT -> Client
	the SUT sends a valid Response for the operations: Create Context Source Registration With Return with Response Status Code set to 201 and Retrieve Context Source Registration with Response Body containing the key '['information'] [0]['entities'][0] ['type']', with the value 'Building' }	

TP Id	TP/NGSI-LD/CS/CSR/033_07
Test objective	Check that the @context is obtained from the request payload body itself if the Content-
	Type header is "application/ld+json"

Reference	ETSI GS CIM 009 V1.5.1 [1], clause 6.3.5	
Config Id	CF_05	
PICS Selection	PICS_HTTP_CONTEXT_RESOLV	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/CSR/033_07_01	
Name	033_07_01 Create one context source registration using a JSON-LD @context	t obtained
	from the request payload with Context	
Parent Release	v1.3.1	
Description	Check that the @context is obtained from the request payload body itself if the	
	Type header is "application/ld+json" and retrieve the information with ngsild co	ontext
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/	SUT <- Client
	ngsi-ld/v1/csourceRegistrations' method set to 'POST'	
	Request Header['Content-	
	Type'] set to 'application/ld+json' and	
	payload defined in file: '\${updated_payload}'	
	payload defined in file: \(\psi_{\text{updated_payload}}\)	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	COT > CHOTTE
	Create Context Source Registration With Return with Response Status	
	Code set to	
	201 and	
	Retrieve Context Source Registration with Response Body containing	
	the key '['information']	
	[0]['entities'][0]	
	['type']', with the value 'Vehicle'	
	}	
TP Permutation id	TP/NGSI-LD/CS/CSR/033_07_02	
Name	033_07_02 Create one context source registration using a JSON-LD @context from the request payload without Context	t obtained
Parent Release	V1.3.1	•
Description	Check that the @context is obtained from the request payload body itself if the	
-	Type header is "application/ld+json" and retrieve the information without ngsilo	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/	SUT <- Client
	ngsi-ld/v1/csourceRegistrations'	
	method set to 'POST'	
	Request Header['Content-	
	Type'] set to 'application/ld+json' and	
	payload defined in file: '\${updated_payload}'	
	than (SUT -> Client
	then { the SUT sends a valid Response for the operations:	SUT -> Client
	Create Context Source Registration With Return with Response Status	
	Code set to	
	201 and	
	Retrieve Context Source Registration with Response Body containing	
	the key '['information']	
	[0]['entities'][0]	
	['type']', with the value 'https://ngsi-ld-test-	
		1
	suite/context#Vehicle'	

TP Id	TP/NGSI-LD/CS/CSR/033_09
Test objective	Check that an HTTP error response of type BadRequestData is raised if the Content-Type header is "application/ld+json" and a JSON-LD Link header is present in the incoming HTTP request
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 6.3.5
Config Id	CF_05
PICS Selection	PICS HTTP CONTEXT RESOLV

Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/CSR/033_09_01	
Name	033_09_01 Create one context source registration with a Link header and a Journal type	SON-LD
Parent Release	v1.3.1	
Description	Check that an HTTP error response of type BadRequestData is raised if the C header is "application/ld+json" and a JSON-LD Link header is present in the ir request	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'POST' Request Header['Content- Type'] set to 'application/ld+json' and payload defined in file: '\${updated_payload}' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Create Context Source Registration With Return with Response Status Code set to 400 and Create Context Source Registration With Return with Response Body containing the type 'https:// uri.etsi.org/ngsi-Id/errors/ BadRequestData' and Create Context Source Registration With Return with Response body containing 'title' element }	SUT -> Client

TP ld	TD/NCCLLD/CC/CCD/022 40	
	TP/NGSI-LD/CS/CSR/033_10	
Test objective	Check that one cannot create a context source with invalid content	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 6.3.5	
Config Id	CF_05	
PICS Selection	PICS_HTTP_CONTEXT_RESOLV	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/CSR/033_10_01	
Name	033_10_01 Create a context source registration with a different data structure	than
	CSourceRegistration data type	
Parent Release	v1.3.1	
Description	Check that one cannot create a context source with invalid content	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/csourceRegistrations'	
	method set to 'POST'	
	Request Header['Content-	
	Type'] set to 'application/ld+json' and	
	payload defined in file: '\${updated_payload}'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Create Context Source Registration With Return with Response Status	
	Code set to	
	400 and	
	Create Context Source Registration With Return with Response	
	Header: Location containing	
	\$\${registration_id}	
	}	
TP Permutation id	TP/NGSI-LD/CS/CSR/033_10_02	
Name	033_10_02 Create a context source registration with a date in the past	
Parent Release	v1.3.1	

Description	Check that one cannot create a context source with invalid content	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/csourceRegistrations'	
	method set to 'POST'	
	Request Header['Content-	
	Type'] set to 'application/ld+json' and	
	payload defined in file: '\${updated_payload}'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Create Context Source Registration With Return with Response Status	
	Code set to	
	400 and	
	Create Context Source Registration With Return with Response	
	Header: Location containing	
	\$\${registration_id}	
	}	

TP ld	TP/NGSI-LD/CS/CSR/033_08	
Test objective	Check that an HTTP error response of type BadRequestData is raised if the C	
	header is "application/ld+json" and the request payload body does not contain	a @context
	term	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 6.3.5	
Config Id	CF_05	
PICS Selection	PICS_HTTP_CONTEXT_RESOLV	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/CSR/033_08_01	
Name	033_08_01 Create one context source registration not containing a JSON-LD	@context with
	a JSON-LD content type	
Parent Release	v1.3.1	
Description	Check that an HTTP error response of type BadRequestData is raised if the C	
	header is "application/ld+json" and the request payload body does not contain	a @context
	term	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to /	
	ngsi-ld/v1/csourceRegistrations'	
	method set to 'POST'	
	Request Header['Content-	
	Type'] set to 'application/ld+json' and	
	payload defined in file: '\${updated_payload}'	
	then (SUT -> Client
	then { the SUT sends a valid Response for the operations:	SUT -> Client
	Create Context Source Registration With Return with Response Status Code set to	
	400 and	
	Create Context Source Registration With Return with Response Body	
	containing the type 'https://	
	uri.etsi.org/ngsi-ld/errors/	
	BadRequestData' and	
	Create Context Source Registration With Return with Response body	
	Icontaining 'title' element	
	containing 'title' element }	

4.3.2 RegistrationSubscription

4.3.2.1 RetrieveContextSourceRegistrationSubscription

T		
TP ld	TP/NGSI-LD/CS/REGSUB/040_01	
Test objective	Check that one can retrieve a context source registration subscription	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.11.4	
Config Id	CF_05	
PICS Selection	PICS_CsoRegSubRet	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing a Context So	ource
	Registration Subscription (CSRS1) }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/REGSUB/040_01_01	
Name	040_01_01 Retrieve Context Source Registration Subscription	
Parent Release	v1.3.1	
Description	Check that one can retrieve a context source registration subscription	
Expected behaviour	Test events	Direction
Expected benaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceSubscriptions/{subscription_id}' method set to 'GET' Retrieve Context Source Registration Subscription with data: and Query Parameter: subscription id set to '\$ {subscription_id}' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ isonld-contexts/ngsi-ld-test-suite-compound.jsonld' } then { the SUT sends a valid Response for the operations: Retrieve Context Source Registration Subscription with Response Status Code set to 200 and Retrieve Context Source Registration Subscription with Response reason set to 'OK' and Retrieve Context Source Registration Subscription with The response headers Link is set to 'expected_link_header=\$ {expected link header}' and Retrieve Context Source Registration Subscription with Check Created Resource Set To and Query Parameter: 'created_resource' set to 'expectation_payload' and Ouery Parameter: 'response_body' set to 'response.json()' and Query Parameter: 'ignored_keys' set to 'ignored_attributes' list of keys and Retrieve Context Source Registration Subscription with The Response Body '\$ {response.json()}' might contain the key 'lastNotification' in the NotificationParams and Retrieve Context Source Registration Subscription with The Response Body '\$ {response.json()}' might contain the key 'lastFailure' in the NotificationParams and Retrieve Context Source Registration Subscription with The Response Body '\$ {response.json()}' might contain the key 'lastSuccess' in the NotificationParams and Retrieve Context Source Registration Subscription with The Response Body '\$ {response.json()}' might contain the key 'lastSuccess' in the NotificationParams and Retrieve Context Source Registration Subscription with The Response Body '\$ {response.json()}' might contain the key 'lastSucce	SUT -> Client
	Body `\$ {response.json()}' might contain the key 'lastSuccess' in the NotificationParams and Retrieve Context Source Registration Subscription with The Response Body `\$	

TP ld	TP/NGSI-LD/CS/REGSUB/040_03	
Test objective	Check that one cannot retrieve an unknown context source registration subsci	ription, an
	error of type ResourceNotFound shall be raised	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.11.4	
Config Id	CF_05	
PICS Selection	PICS_CsoRegSubRet	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/REGSUB/040_03_01	
Name	040_03_01 Retrieve Unknown Context Source Registration Subscription	
Parent Release	v1.3.1	
Description	Check that one cannot retrieve an unknown context source registration subsci	ription, an
-	error of type ResourceNotFound shall be raised	•
Expected behaviour	Test events	Direction
-	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/csourceSubscriptions/{subscription_id}'	
	method set to 'GET'	
	Retrieve Context Source Registration Subscription with data: and	
	Query Parameter: subscription id set to 'urn:ngsi-	
	ld:Subscription:unknowSubscription'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Retrieve Context Source Registration Subscription with Response	
	Status Code set to	
	404 and	
	Retrieve Context Source Registration Subscription with Response	
	reason set to 'Not Found' and	
	Retrieve Context Source Registration Subscription with Response	
	Header: Content-	
	Type set to application/ ison and	
	Retrieve Context Source Registration Subscription with Response Body	
	containing the type 'problem_type=\$	
	{ERROR_TYPE_RESOURCE_NOT_FOUND}' and Response Body Title is a	
	string and is not	
	\${EMPTY}	
	and Response Body Detail is a string and is not	
	\${EMPTY}	
	and Response Boty Title is not equal to Response Body Detail	
	}	
	U	1

TP Id	TP/NGSI-LD/CS/REGSUB/040_02	
Test objective	Check that one cannot retrieve a context source registration subscription with an error of type BadRequestData shall be raised	an invalid URI,
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.11.4	
Config Id	CF_05	
PICS Selection	PICS_CsoRegSubRet	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/REGSUB/040_02_01	
Name	040_02_01 Retrieve Context Source Registration Subscription With An Invalid	l ld
Parent Release	v1.3.1	
Description	Check that one cannot retrieve a context source registration subscription with an invalid URI, an error of type BadRequestData shall be raised	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceSubscriptions/{subscription_id}' method set to 'GET' Retrieve Context Source Registration Subscription with data: and Query Parameter: subscription id set to 'invalidUri' }	SUT <- Client

then {	SUT -> Client
the SUT sends a valid Response for the operations:	
Retrieve Context Source Registration Subscription with Response	
Status Code set to	
400 and	
Retrieve Context Source Registration Subscription with Response	
reason set to 'Bad Request' and	
Retrieve Context Source Registration Subscription with Response	
Header: Content-	
Type set to application/	
ison and	
Retrieve Context Source Registration Subscription with Response Body	
containing the type 'problem_type=\$	
{ERROR_TYPE_BAD_REQUEST_DATA}' and Response Body Title is a	
string and is not	
\${EMPTY}	
and Response Body Detail is a string and is not	
\${EMPTY}	
and Response Boty Title is not equal to Response Body Detail	
}	

4.3.2.2 ContextSourceRegistrationSubscriptionNotificationBehaviour

	LD/CS/REGSUB/047_16	
Test objective Check if o		
	ne updates a context source registration subscription, a CsourceNot	ification will
be sent wi	th all currently matching context source registrations	
Reference ETSI GS (CIM 009 V1.5.1 [1], clause 5.11.7	
Config Id CF_05		
PICS Selection PICS_NO	TIFICATIONS_CSR	
Initial conditions with { the	SUT being in the "initial state" and the SUT containing a list of Conte	xt Source
Registration	ons (CSRs) providing latest information about some entities and the	SUT
containing	a Context Source Registration Subscription (CSRS1) }	
	Permutations	
TP Permutation id TP/NGSI-I	LD/CS/REGSUB/047_16_01	
Name 047_16_0	1 MatchFirstContextSourceRegistration	
Parent Release v1.3.1	<u>"</u>	
Description Check if o	ne updates a context source registration subscription, a CsourceNot	ification will
	th all currently matching context source registrations	
Expected behaviour	Test events	Direction
when {		SUT <- Client
the SUT	Freceives a Request from the client containing:	
	set to '/	
ngsi-ld/v1/	csourceSubscriptions/{subscription_id}'	
	od set to 'PATCH'	
Upda	te Context Source Registration Subscription with subscription id	
set to '\$		
{subscripti	on_id}' and subscription update fragment set to '\$	
{subscripti	on_update_fragment}'	
}		
then {		SUT -> Client
the clier		
	' receives a valid Notification containing:	
	cation received Waiting for Notification and validate it and	
	Parameter: expected_subscription_id set to 'subscription_id' and	
	Parameter: expected_context_source_registration_ids set to '\$	
	ext_source_registration_id}' and	
	Parameter: expected_trigger_reason set to 'newlyMatching' and	
	Parameter: expected_notification_data_entities set to '\$	
{EMPTY}'		
Query F	Parameter: timeout set to '5'	
}	D/00/DE00/JD/047-40-00	
	LD/CS/REGSUB/047_16_02	
	2 MatchSecondContextSourceRegistration	
Parent Release v1.3.1		
	ne updates a context source registration subscription, a CsourceNot	ification will
lho cont wi	th all currently matching context source registrations	

Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/csourceSubscriptions/{subscription_id}'	
	method set to 'PATCH'	
	Update Context Source Registration Subscription with subscription id	
	set to '\$	
	{subscription_id}' and subscription update fragment set to '\$	
	{subscription_update_fragment}'	
	then {	SUT -> Client
	the client at '\$	SOT -> Client
	{endpoint}' receives a valid Notification containing:	
	Notification received Waiting for Notification and validate it and	
	Query Parameter: expected_subscription_id set to 'subscription_id' and	
	Query Parameter: expected_subscription_id set to subscription_id and Query Parameter: expected_context_source_registration_ids set to '\$	
	{second_context_source_registration_id}' and	
	Query Parameter: expected_trigger_reason set to 'newlyMatching' and	
	Query Parameter: expected_notification_data_entities set to '\$	
	{EMPTY}' and	
	Query Parameter: timeout set to '5'	
	}	
TP Permutation id	TP/NGSI-LD/CS/REGSUB/047_16_03	1
Name	047_16_03 MatchBothContextSourceRegistrations	
Parent Release	v1.3.1	
Description	Check if one updates a context source registration subscription, a CsourceN	otification will
-	be sent with all currently matching context source registrations	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/csourceSubscriptions/{subscription_id}'	
	method set to 'PATCH'	
	Update Context Source Registration Subscription with subscription id	
	set to '\$	
	{subscription_id}' and subscription update fragment set to '\$	
	{subscription_update_fragment}'	
	}	
	then {	SUT -> Client
	the client at '\$	
	{endpoint}' receives a valid Notification containing:	
	Notification received Waiting for Notification and validate it and	
	Query Parameter: expected_subscription_id set to 'subscription_id' and	
	Query Parameter: expected_context_source_registration_ids set to '\$	
	{first_context_source_registration_id}' and	
	Query Parameter: expected_trigger_reason set to 'newlyMatching' and	
	Query Parameter: expected_notification_data_entities set to '\$	
	{EMPTY}' and	
	Query Parameter: timeout set to '5'	

TP ld	TP/NGSI-LD/CS/REGSUB/047_14
Test objective	Check if a context source registrations subscription defines a geoQ member, a
	CsourceNotification will be triggered from matching context source registrations with a
	matching location member
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.11.7
Config Id	CF_05
PICS Selection	PICS_NOTIFICATIONS_CSR
Initial conditions	with { the SUT being in the "initial state" and the SUT containing a Context Source
	Registration Subscription (CSRS1) }
	Permutations
TP Permutation id	TP/NGSI-LD/CS/REGSUB/047_14_01
Name	047_14_01 Receive cSourceNotification For Matching Context Source Registrations On
	Location
Parent Release	v1.3.1

Description	Check if a context source registrations subscription defines a geoQ member, CsourceNotification will be triggered from matching context source registration matching location member	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'POST' Create Context Source Registration Request with Content- Type set to 'application/ ld+json' and payload set to '\$ {context_source_registration_payload}' }	SUT <- Client
	then { the client at '\$ {endpoint}' receives a valid Notification containing: Notification received Waiting for Notification and validate it and Query Parameter: expected_subscription_id set to 'subscription_id' and Query Parameter: expected_context_source_registration_ids set to 'expected_context_source_registration_ids' and Query Parameter: expected_trigger_reason set to 'newlyMatching' and Query Parameter: expected_notification_data_entities set to '\$ {EMPTY}' and Query Parameter: timeout set to '5' }	SUT -> Client

TD L	TD/NOOL D/OO/DECOLID/O47, 40	
TP Id	TP/NGSI-LD/CS/REGSUB/047_10	
Test objective	Check if a context source registration subscription defines temporalQ membe	
	timeproperty observedAt, the temporal query is matched against the observat	ionInterval of
D (matching context source registrations	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.11.7	
Config Id	CF_05	
PICS Selection	PICS_NOTIFICATIONS_CSR	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing a Context So	urce
	Registration Subscription (CSRS1) }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/REGSUB/047_10_01	
Name	047_10_01 Receive cSourceNotification For Matching Context Source Regist	rations On
	Observation Interval	
Parent Release	v1.3.1	
Description	Check if a context source registration subscription defines temporalQ membe	
	timeproperty observedAt, the temporal query is matched against the observat	ionInterval of
	matching context source registrations	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/csourceRegistrations'	
	method set to 'POST'	
	Create Context Source Registration Request with Content-	
	Type set to 'application/	
	ld+json' and payload set to '\$	
	{context_source_registration_payload}'	
	}	
	then {	SUT -> Client
	the client at '\$	
	{endpoint}' receives a valid Notification containing:	
	Notification received Waiting for Notification and validate it and	
	Query Parameter: expected_subscription_id set to 'subscription_id' and	
	Query Parameter: expected_context_source_registration_ids set to	
	'expected_context_source_registration_ids' and	
	Query Parameter: expected_trigger_reason set to 'newlyMatching' and	
	Query Parameter: expected_notification_data_entities set to '\$	
	{EMPTY}' and	
	Query Parameter: timeout set to '5'	
	}	

TP ld	TP/NGSI-LD/CS/REGSUB/047_09	
Test objective	Check if a context source registration subscription defines an "entities" memb	
	CsourceNotification will be triggered from context source registrations with inf	formation
	member matching the described "entities"	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.11.7	
Config Id	CF_05	
PICS Selection	PICS_NOTIFICATIONS_CSR	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/REGSUB/047_09_01	
Name	047_09_01 Receive cSourceNotification For No Longer Matching Context So	urce
	Registrations Providing Latest Information	
Parent Release	v1.3.1	
Description	Check if a context source registration subscription defines an "entities" memb	
	CsourceNotification will be triggered from context source registrations with inf	formation
	member matching the described "entities"	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/csourceRegistrations/{context_source_registration_id}'	
	method set to 'PATCH'	
	Update Context Source Registration with CSR Id set to '\$	
	{context_source_registration_id}' and subscription update fragment set to '\$	
	{update_fragment}'	
	}	0
	then {	SUT -> Client
	the client at '\$	
	{endpoint}' receives a valid Notification containing:	
	Notification received Waiting for Notification and validate it and	
	Query Parameter: expected_subscription_id set to 'subscription_id' and	
	Query Parameter: expected_context_source_registration_ids set to 'expected_context_source_registration_ids' and	
	Query Parameter: expected_trigger_reason set to 'noLongerMatching'	
	and	
	Query Parameter: expected_notification_data_entities set to '\$	
	{EMPTY}' and	
	Query Parameter: timeout set to '5'	
	}	

TP ld	TP/NGSI-LD/CS/REGSUB/047 12
	_
Test objective	Check if a context source registrations subscription defines entities member and
	watchedAttributes member, a CsourceNotification will be triggered from context source
	registrations with information member matching the described "entities" and "attributes"
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.11.7
Config Id	CF_05
PICS Selection	PICS_NOTIFICATIONS_CSR
Initial conditions	with { the SUT being in the "initial state" and the SUT containing a Context Source
	Registration Subscription (CSRS1) }
	Permutations
TP Permutation id	TP/NGSI-LD/CS/REGSUB/047_12_01
Name	047_12_01 Receive cSourceNotification For Matching Context Source Registrations On
	Watched Attributes
Parent Release	v1.3.1
Description	Check if a context source registrations subscription defines entities member and
	watchedAttributes member, a CsourceNotification will be triggered from context source
	registrations with information member matching the described "entities" and "attributes"

Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/csourceRegistrations'	
	method set to 'POST'	
	Create Context Source Registration Request with Content-	
	Type set to 'application/	
	ld+json' and payload set to '\$	
	{context_source_registration_payload}'	
	}	
	then {	SUT -> Client
	the client at '\$	
	{endpoint}' receives a valid Notification containing:	
	Notification received Waiting for Notification and validate it and	
	Query Parameter: expected_subscription_id set to 'subscription_id' and	
	Query Parameter: expected_context_source_registration_ids set to	
	expected_context_source_registration_ids and	
	Query Parameter: expected_trigger_reason set to 'newlyMatching' and	
	Query Parameter: expected_notification_data_entities set to '\$	
	{EMPTY}' and	
	Query Parameter: timeout set to '5'	
	}	

TP ld	TP/NGSI-LD/CS/REGSUB/047_15	
Test objective	Check if a context source registrations subscription does not define a geopro	perty in the
	geoQ member, a CsourceNotification will be triggered from matching context	source
	registrations with a matching location member	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.11.7	
Config Id	CF_05	
PICS Selection	PICS_NOTIFICATIONS_CSR	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing a Context S	ource
	Registration Subscription (CSRS1) }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/REGSUB/047_15_01	
Name	047_15_01 Receive cSourceNotification For Matching Context Source Regis	strations On
	Location As Default	
Parent Release	v1.3.1	
Description	Check if a context source registrations subscription does not define a geopro	
	geoQ member, a CsourceNotification will be triggered from matching context	source
	registrations with a matching location member	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/csourceRegistrations'	
	method set to 'POST'	
	Create Context Source Registration Request with Content-	
	Type set to 'application/	
	Id+json' and payload set to '\$	
	{context_source_registration_payload}'	
	then (SUT -> Client
	then { the client at '\$	SUT -> Client
	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	
	{endpoint}' receives a valid Notification containing: Notification received Waiting for Notification and validate it and	
	Query Parameter: expected_subscription_id set to 'subscription_id' and	
	Query Parameter: expected_subscription_id set to subscription_id and Query Parameter: expected_context_source_registration_ids set to	
	expected_context_source_registration_ids' and	
	Query Parameter: expected_trigger_reason set to 'newlyMatching' and	
	Query Parameter: expected_trigger_teason set to flewlywatching and Query Parameter: expected_notification_data_entities set to '\$	
	{EMPTY}' and	
	Query Parameter: timeout set to '5'	
	}	
	IJ	

TP Id TP/NGSI-LD/CS/REGSUB/047_13	
Test objective Check if a context source registrations subscription does not define wa	tchedAttributes
member, a CsourceNotification will be triggered from context source re	gistrations with
information member matching all attributes of the described entities	
Reference ETSI GS CIM 009 V1.5.1 [1], clause 5.11.7	
Config Id CF_05	
PICS Selection PICS_NOTIFICATIONS_CSR	
Initial conditions with { the SUT being in the "initial state" and the SUT containing a Con-	text Source
Registration Subscription (CSRS1) }	
Permutations	
TP Permutation id TP/NGSI-LD/CS/REGSUB/047_13_01	
Name 047_13_01 Receive cSourceNotification For Matching Context Source	Registrations On Any
watchedAttribute	
Parent Release v1.3.1	
Description Check if a context source registrations subscription does not define wa	tchedAttributes
member, a CsourceNotification will be triggered from context source re	gistrations with
information member matching all attributes of the described entities	
Expected behaviour Test events	Direction
when {	SUT <- Client
the SUT receives a Request from the client containing:	
URL set to '/	
ngsi-ld/v1/csourceRegistrations'	
method set to 'POST'	
method set to 'POST' Create Context Source Registration Request with Content-	
method set to 'POST' Create Context Source Registration Request with Content- Type set to 'application/	
method set to 'POST' Create Context Source Registration Request with Content- Type set to 'application/ Id+json' and payload set to '\$	
method set to 'POST' Create Context Source Registration Request with Content- Type set to 'application/	
method set to 'POST' Create Context Source Registration Request with Content- Type set to 'application/ Id+json' and payload set to '\$ {context_source_registration_payload}' }	SLIT -> Client
method set to 'POST' Create Context Source Registration Request with Content- Type set to 'application/ Id+json' and payload set to '\$ {context_source_registration_payload}' } then {	SUT -> Client
method set to 'POST' Create Context Source Registration Request with Content- Type set to 'application/ Id+json' and payload set to '\$ {context_source_registration_payload}' } then { the client at '\$	SUT -> Client
method set to 'POST' Create Context Source Registration Request with Content- Type set to 'application/ Id+json' and payload set to '\$ {context_source_registration_payload}' } then { the client at '\$ {endpoint}' receives a valid Notification containing:	SUT -> Client
method set to 'POST' Create Context Source Registration Request with Content- Type set to 'application/ Id+json' and payload set to '\$ {context_source_registration_payload}' } then { the client at '\$ {endpoint}' receives a valid Notification containing: Notification received Waiting for Notification and validate it and	
method set to 'POST' Create Context Source Registration Request with Content- Type set to 'application/ Id+json' and payload set to '\$ {context_source_registration_payload}' } then { the client at '\$ {endpoint}' receives a valid Notification containing: Notification received Waiting for Notification and validate it and Query Parameter: expected_subscription_id set to 'subscription_id' a	
method set to 'POST' Create Context Source Registration Request with Content- Type set to 'application/ Id+json' and payload set to '\$ {context_source_registration_payload}' } then { the client at '\$ {endpoint}' receives a valid Notification containing: Notification received Waiting for Notification and validate it and Query Parameter: expected_subscription_id set to 'subscription_id' a Query Parameter: expected_context_source_registration_ids set to	
method set to 'POST' Create Context Source Registration Request with Content- Type set to 'application/ Id+json' and payload set to '\$ {context_source_registration_payload}' } then { the client at '\$ {endpoint}' receives a valid Notification containing: Notification received Waiting for Notification and validate it and Query Parameter: expected_subscription_id set to 'subscription_id' a Query Parameter: expected_context_source_registration_ids set to 'expected_context_source_registration_ids' and	and
method set to 'POST' Create Context Source Registration Request with Content- Type set to 'application/ Id+json' and payload set to '\$ {context_source_registration_payload}' } then { the client at '\$ {endpoint}' receives a valid Notification containing: Notification received Waiting for Notification and validate it and Query Parameter: expected_subscription_id set to 'subscription_id' a Query Parameter: expected_context_source_registration_ids set to 'expected_context_source_registration_ids' and Query Parameter: expected_trigger_reason set to 'newlyMatching' a	and
method set to 'POST' Create Context Source Registration Request with Content- Type set to 'application/ Id+json' and payload set to '\$ {context_source_registration_payload}' } then { the client at '\$ {endpoint}' receives a valid Notification containing: Notification received Waiting for Notification and validate it and Query Parameter: expected_subscription_id set to 'subscription_id' a Query Parameter: expected_context_source_registration_ids set to 'expected_context_source_registration_ids' and Query Parameter: expected_trigger_reason set to 'newlyMatching' a Query Parameter: expected_notification_data_entities set to '\$	and
method set to 'POST' Create Context Source Registration Request with Content- Type set to 'application/ Id+json' and payload set to '\$ {context_source_registration_payload}' } then { the client at '\$ {endpoint}' receives a valid Notification containing: Notification received Waiting for Notification and validate it and Query Parameter: expected_subscription_id set to 'subscription_id' a Query Parameter: expected_context_source_registration_ids set to 'expected_context_source_registration_ids' and Query Parameter: expected_trigger_reason set to 'newlyMatching' a	and

TDII	TTD/N GOL I D (GO (DE GOL ID (G IT)))	
TP Id	TP/NGSI-LD/CS/REGSUB/047_11	
Test objective	Check if a context source registration subscription defines temporalQ member with	
	timeproperty createdAt or modifiedAt, the temporal query is matched against the	he
	managementInterval of matching context source registrations	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.11.7	
Config Id	CF_05	
PICS Selection	PICS_NOTIFICATIONS_CSR	
Initial conditions	with { the SUT containing an initial state }	
Permutations Permutations		
TP Permutation id	TP/NGSI-LD/CS/REGSUB/047_11_01	
Name	047_11_01 CreatedAt	
Parent Release	v1.3.1	
Description	Scription Check if a context source registration subscription defines temporalQ member with	
	timeproperty createdAt or modifiedAt, the temporal query is matched against the	he
	managementInterval of matching context source registrations	
Expected behaviour	Test events	Direction
_	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/csourceSubscriptions'	
	method set to 'POST'	
	Create Context Source Registration Subscription with filename set to '\$	
	{subscription_payload}', accept set to '\$	
	{EMPTY}', and content-type set to 'application/	

	line en a	1
	ld+json'	
	then {	SUT -> Client
	the client at '\$	
	{endpoint}' receives a valid Notification containing:	
	Notification received Waiting for Notification and validate it and	
	Query Parameter: expected_subscription_id set to 'subscription_id' and Query Parameter: expected_context_source_registration_ids set to	
	'expected_context_source_registration_ids' and	
	Query Parameter: expected_trigger_reason set to 'newlyMatching' and	
	Query Parameter: expected_notification_data_entities set to '\$	
	{EMPTY}' and	
	Query Parameter: timeout set to '5'	
	}	
TP Permutation id	TP/NGSI-LD/CS/REGSUB/047_11_02	
Name Parent Release	047_11_02 ModifiedAt	
Description	V1.3.1	r swith
Description	Check if a context source registration subscription defines temporalQ member timeproperty createdAt or modifiedAt, the temporal query is matched against the source registration subscription defines temporalQ member timeproperty createdAt or modifiedAt, the temporal query is matched against the source registration subscription defines temporalQ member timeproperty createdAt or modifiedAt, the temporal query is matched against the source registration subscription defines temporalQ member timeproperty createdAt or modifiedAt, the temporal query is matched against the source registration subscription defines temporalQ member timeproperty createdAt or modifiedAt, the temporal query is matched against the source registration and the source registration a	
	managementInterval of matching context source registrations	ine
Expected behaviour	Test events	Direction
•		
	Iwnen {	SUT <- Client
	when { the SUT receives a Request from the client containing:	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceSubscriptions'	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceSubscriptions' method set to 'POST'	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceSubscriptions' method set to 'POST' Create Context Source Registration Subscription with filename set to '\$	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceSubscriptions' method set to 'POST' Create Context Source Registration Subscription with filename set to '\$ {subscription_payload}', accept set to '\$	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceSubscriptions' method set to 'POST' Create Context Source Registration Subscription with filename set to '\$ {subscription_payload}', accept set to '\$ {EMPTY}', and content-type set to 'application/	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceSubscriptions' method set to 'POST' Create Context Source Registration Subscription with filename set to '\$ {subscription_payload}', accept set to '\$	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceSubscriptions' method set to 'POST' Create Context Source Registration Subscription with filename set to '\$ {subscription_payload}', accept set to '\$ {EMPTY}', and content-type set to 'application/ Id+json' }	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceSubscriptions' method set to 'POST' Create Context Source Registration Subscription with filename set to '\$ {subscription_payload}', accept set to '\$ {EMPTY}', and content-type set to 'application/ Id+json' } then {	SUT -> Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceSubscriptions' method set to 'POST' Create Context Source Registration Subscription with filename set to '\$ {subscription_payload}', accept set to '\$ {EMPTY}', and content-type set to 'application/ Id+json' } then { the client at '\$	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceSubscriptions' method set to 'POST' Create Context Source Registration Subscription with filename set to '\$ {subscription_payload}', accept set to '\$ {EMPTY}', and content-type set to 'application/ Id+json' } then {	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceSubscriptions' method set to 'POST' Create Context Source Registration Subscription with filename set to '\$ {subscription_payload}', accept set to '\$ {EMPTY}', and content-type set to 'application/ Id+json' } then { the client at '\$ {endpoint}' receives a valid Notification containing: Notification received Waiting for Notification and validate it and Query Parameter: expected_subscription_id set to 'subscription_id' and	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceSubscriptions' method set to 'POST' Create Context Source Registration Subscription with filename set to '\$ {subscription_payload}', accept set to '\$ {EMPTY}', and content-type set to 'application/ Id+json' } then { the client at '\$ {endpoint}' receives a valid Notification containing: Notification received Waiting for Notification and validate it and Query Parameter: expected_subscription_id set to 'subscription_id' and Query Parameter: expected_context_source_registration_ids set to	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceSubscriptions' method set to 'POST' Create Context Source Registration Subscription with filename set to '\$ {subscription_payload}', accept set to '\$ {EMPTY}', and content-type set to 'application/ Id+json' } then { the client at '\$ {endpoint}' receives a valid Notification containing: Notification received Waiting for Notification and validate it and Query Parameter: expected_subscription_id set to 'subscription_id' and Query Parameter: expected_context_source_registration_ids set to 'expected_context_source_registration_ids' and	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceSubscriptions' method set to 'POST' Create Context Source Registration Subscription with filename set to '\$ {subscription_payload}', accept set to '\$ {EMPTY}', and content-type set to 'application/ Id+json' } then { the client at '\$ {endpoint}' receives a valid Notification containing: Notification received Waiting for Notification and validate it and Query Parameter: expected_subscription_id set to 'subscription_id' and Query Parameter: expected_context_source_registration_ids set to 'expected_context_source_registration_ids' and Query Parameter: expected_trigger_reason set to 'newlyMatching' and	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceSubscriptions' method set to 'POST' Create Context Source Registration Subscription with filename set to '\$ {subscription_payload}', accept set to '\$ {EMPTY}', and content-type set to 'application/ Id+json' } then { the client at '\$ {endpoint}' receives a valid Notification containing: Notification received Waiting for Notification and validate it and Query Parameter: expected_subscription_id set to 'subscription_id' and Query Parameter: expected_context_source_registration_ids set to 'expected_context_source_registration_ids' and Query Parameter: expected_trigger_reason set to 'newlyMatching' and Query Parameter: expected_notification_data_entities set to '\$	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceSubscriptions' method set to 'POST' Create Context Source Registration Subscription with filename set to '\$ {subscription_payload}', accept set to '\$ {EMPTY}', and content-type set to 'application/ Id+json' } then { the client at '\$ {endpoint}' receives a valid Notification containing: Notification received Waiting for Notification and validate it and Query Parameter: expected_subscription_id set to 'subscription_id' and Query Parameter: expected_context_source_registration_ids set to 'expected_context_source_registration_ids' and Query Parameter: expected_trigger_reason set to 'newlyMatching' and Query Parameter: expected_notification_data_entities set to '\$ {EMPTY}' and	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceSubscriptions' method set to 'POST' Create Context Source Registration Subscription with filename set to '\$ {subscription_payload}', accept set to '\$ {EMPTY}', and content-type set to 'application/ Id+json' } then { the client at '\$ {endpoint}' receives a valid Notification containing: Notification received Waiting for Notification and validate it and Query Parameter: expected_subscription_id set to 'subscription_id' and Query Parameter: expected_context_source_registration_ids set to 'expected_context_source_registration_ids' and Query Parameter: expected_trigger_reason set to 'newlyMatching' and Query Parameter: expected_notification_data_entities set to '\$	

TP ld	TP/NGSI-LD/CS/REGSUB/047_08
Test objective	Check if a context source registration subscription does not define a temporalQ member, a
	CsourceNotification will be triggered from matching context source registrations for context
	sources providing latest information
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.11.7
Config Id	CF_05
PICS Selection	PICS_NOTIFICATIONS_CSR
Initial conditions	with { the SUT being in the "initial state" and the SUT containing a Context Source
	Registration (CSR1) providing latest information about some entities and the SUT containing
	a Context Source Registration Subscription (CSRS1) }
	Permutations
TP Permutation id	TP/NGSI-LD/CS/REGSUB/047_08_01
Name	047_08_01 Receive cSourceNotification For Matching Context Source Registrations
	Providing Latest Information
Parent Release	v1.3.1
Description	Check if a context source registration subscription does not define a temporalQ member, a
	CsourceNotification will be triggered from matching context source registrations for context
	sources providing latest information

Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/csourceRegistrations/{context_source_registration_id}' method set to 'PATCH'	
	Update Context Source Registration with CSR Id set to '\$	
	{context_source_registration_id} and subscription update fragment set to '\$	
	{update_fragment}'	
	}	
	then {	SUT -> Client
	the client at '\$	
	{endpoint}' receives a valid Notification containing:	
	Notification received Waiting for Notification and validate it and	
	Query Parameter: expected_subscription_id set to 'subscription_id' and	
	Query Parameter: expected_context_source_registration_ids set to	
	'expected_context_source_registration_ids' and	
	Query Parameter: expected_trigger_reason set to 'updated' and Query Parameter: expected notification data entities set to '\$	
	{EMPTY}' and	
	Query Parameter: timeout set to '5'	
	Query Farameter. timeout set to 5	
	If	

		1
TP ld	TP/NGSI-LD/CS/REGSUB/047_04	
Test objective	The structure of the csource notification message shall be as mandated by cla	use 5.3.2
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.11.7	
Config Id	CF_05	
PICS Selection	PICS_NOTIFICATIONS_CSR	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing a Context Sou	ırce
	Registration Subscription (CSRS1) }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/REGSUB/047_04_01	
Name	047_04_01 Receive cSourceNotification With Compliant Structure	
Parent Release	v1.3.1	
Description	The structure of the csource notification message shall be as mandated by cla	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/csourceRegistrations'	
	method set to 'POST'	
	Create Context Source Registration Request with Content-	
	Type set to 'application/	
	ld+json' and payload set to '\$	
	{context_source_registration_payload}'	
	}	
	then {	SUT -> Client
	the client at '\$	
	{endpoint}' receives a valid Notification containing:	
	Notification received Waiting for Notification and validate it and	
	Query Parameter: expected_subscription_id set to 'subscription_id' and	
	Query Parameter: expected_context_source_registration_ids set to	
	'expected_context_source_registration_ids' and	
	Query Parameter: expected_trigger_reason set to 'newlyMatching' and	
	Query Parameter: expected_notification_data_entities set to '\$	
	(EMPTY)' and	
	Query Parameter: timeout set to '5'	
	}	

TP Id	TP/NGSI-LD/CS/REGSUB/047_06
Test objective	Check that if a cSourceNotification is not sent successfully, the "notification.timesSent"
	member shall be incremented by one and the notification.lastFailure" and
	"notification lastNotification" members shall be updated with the current timestamp and the
	status of the context source registration subscription shall be updated to "failed"
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.11.7

Config Id	CF_05	
PICS Selection	PICS_NOTIFICATIONS_CSR	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing a Context So Registration Subscription (CSRS1) }	urce
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/REGSUB/047_06_01	
Name	047_06_01 If A cSourceNotification Is Not Successfully Sent The Notification Be Updated	Member Shall
Parent Release	v1.3.1	
Description	Check that if a cSourceNotification is not sent successfully, the "notification.tir member shall be incremented by one and the notification.lastFailure" and "notification.lastNotification" members shall be updated with the current times status of the context source registration subscription shall be updated to "faile	tamp and the d"
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceSubscriptions/{subscription_id}' method set to 'GET' Retrieve Context Source Registration Subscription with data: and Query Parameter: subscription id set to '\$ {subscription_id}' }	SUT <- Client
	then { the SUT sends a valid Response for the operation: Retrieve Context Source Registration Subscription with Response containing: * Notification expectation file path set to 'notifications/ expectations/1-timesSent-failed.json' * Expected Additional Members set to '\$ {expected_notification_additional_members}' * Response Body set to '\$ {response.json()}'	SUT -> Client

TP Id	TP/NGSI-LD/CS/REGSUB/047_02	
Test objective	Check that if the created context source registration subscription does not define	
	timeInterval member, a cSourceNotification, with the appropriate trigger reason	
	"triggerReason" member, will be sent initially on subscription and whenever the	ere is a
	change of a matching Context Source Registration	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.11.7	
Config Id	CF_05	
PICS Selection	PICS_NOTIFICATIONS_CSR	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing a Context Sou	
	Registration (CSR1) providing latest information about some entities and the S	SUT containing
	a Context Source Registration Subscription (CSRS1) }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/REGSUB/047_02_01	
Name	047_02_01 Receive cSourceNotification Initially On Subscription And Whenev	er There Is A
	Change Of A Matching Context Source Registration	
Parent Release	v1.3.1	
Description	Check that if the created context source registration subscription does not define	
	timeInterval member, a cSourceNotification, with the appropriate trigger reason	
	"triggerReason" member, will be sent initially on subscription and whenever the	ere is a
	change of a matching Context Source Registration	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/csourceRegistrations/{context_source_registration_id}'	
	method set to 'PATCH'	
	Update Context Source Registration with CSR Id set to '\$	
	{context_source_registration_id} and subscription update fragment set to '\$	
	{update_fragment}'	
	}	

then {	SUT -> Client
the client at '\$	
{endpoint}' receives a valid Notification containing:	
Notification received Waiting for Notification and validate it and	
Query Parameter: expected_subscription_id set to 'subscription_id' and	
Query Parameter: expected_context_source_registration_ids set to	
'expected_context_source_registration_ids' and	
Query Parameter: expected_trigger_reason set to 'updated' and	
Query Parameter: expected_notification_data_entities set to '\$	
{EMPTY}' and	
Query Parameter: timeout set to '5'	
}	

	the
with { the SUT being in the "initial state" and the SUT containing a Context Son Registration Subscription (CSRS1) }	ource
Permutations	
TP/NGSI-LD/CS/REGSUB/047_07_01	
047_07_01 PausedSubscription	
v1.3.1	
Check that a cSourceNotification shall only be sent if and only if the status of	the
Test events	Direction
when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceSubscriptions/{subscription_id}' method set to 'PATCH' Update Context Source Registration Subscription with subscription id set to '\$ {subscription_id}' and subscription update fragment set to '\$ {subscription_update_fragment}' } then { the client at '\$ {endpoint}' receives a valid Notification containing: Notification received Waiting for no Notification data and Query Parameter: timeout set to '5'	SUT <- Client SUT -> Client
TP/NGSI-LD/CS/REGSUB/047 07 02	
	the
	Direction
when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceSubscriptions/{subscription_id}' method set to 'PATCH' Update Context Source Registration Subscription with subscription id set to '\$ {subscription_id}' and subscription update fragment set to '\$ {subscription_update_fragment}'	SUT <- Client
	TP/NGSI-LD/CS/REGSUB/047_07_01 047_07_01 PausedSubscription v1.3.1 Check that a cSourceNotification shall only be sent if and only if the status of corresponding subscription is active, neither paused or expired Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceSubscriptions/{subscription_id}' method set to 'PATCH' Update Context Source Registration Subscription with subscription id set to '\$ {subscription_id}' and subscription update fragment set to '\$ {subscription_update_fragment}' } then { the client at '\$ {endpoint}' receives a valid Notification containing: Notification received Waiting for no Notification data and Query Parameter: timeout set to '5' } TP/NGSI-LD/CS/REGSUB/047_07_02 047_07_02 ExpiredSubscription v1.3.1 Check that a cSourceNotification shall only be sent if and only if the status of corresponding subscription is active, neither paused or expired Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceSubscriptions/{subscription_id}' method set to 'PATCH' Update Context Source Registration Subscription with subscription id set to '\$ {subscription_id}' and subscription update fragment set to '\$

TD L-I	TTD/N 0.00 L D /0.0 /DT 0.0 L D /0.47 . 0.5	
TP Id	TP/NGSI-LD/CS/REGSUB/047_05	
Test objective	Check that if a cSourceNotification is sent successfully to the "endpoint" mem	ber, the
	"notification.timesSent" member shall be incremented by one and the	1 4 1 24
	"notification.lastSuccess" and "notification.lastNotification" members shall be	
	the current timestamp and the status of the context source registration subscr	iption shall be
Defenses	updated to "ok"	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.11.7	
Config Id	CF_05	
PICS Selection	PICS_NOTIFICATIONS_CSR	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing a Context So	urce
	Registration Subscription (CSRS1) }	
TD D	Permutations	
TP Permutation id	TP/NGSI-LD/CS/REGSUB/047_05_01	. O. II D
Name	047_05_01 If A cSourceNotification Is Successfully Sent The Notification Men	nber Shall Be
D	Updated	
Parent Release	V1.3.1	
Description	Check that if a cSourceNotification is sent successfully to the "endpoint" mem	ber, the
	"notification.timesSent" member shall be incremented by one and the	
	"notification.lastSuccess" and "notification.lastNotification" members shall be u	
	the current timestamp and the status of the context source registration subscrupdated to "ok"	iption snall be
Expected behaviour	Test events	Direction
Expected beliavious	1631 GVGIII3	Direction
	when (SLIT <- Client
	when { the SLIT receives a Request from the client containing:	SUT <- Client
,	the SUT receives a Request from the client containing:	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceSubscriptions/{subscription_id}'	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceSubscriptions/{subscription_id}' method set to 'GET'	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceSubscriptions/{subscription_id}' method set to 'GET' Retrieve Context Source Registration Subscription with data: and	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceSubscriptions/{subscription_id}' method set to 'GET' Retrieve Context Source Registration Subscription with data: and Query Parameter: subscription id set to '\$	SUT <- Client
•	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceSubscriptions/{subscription_id}' method set to 'GET' Retrieve Context Source Registration Subscription with data: and	SUT <- Client
•	the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceSubscriptions/{subscription_id}' method set to 'GET' Retrieve Context Source Registration Subscription with data: and Query Parameter: subscription id set to '\$ {subscription_id}' }	
•	the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceSubscriptions/{subscription_id}' method set to 'GET' Retrieve Context Source Registration Subscription with data: and Query Parameter: subscription id set to '\$ {subscription_id}' } then {	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceSubscriptions/{subscription_id}' method set to 'GET' Retrieve Context Source Registration Subscription with data: and Query Parameter: subscription id set to '\$ {subscription_id}' } then { the SUT sends a valid Response for the operation:	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceSubscriptions/{subscription_id}' method set to 'GET' Retrieve Context Source Registration Subscription with data: and Query Parameter: subscription id set to '\$ {subscription_id}' } then { the SUT sends a valid Response for the operation: Retrieve Context Source Registration Subscription with Response	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceSubscriptions/{subscription_id}' method set to 'GET' Retrieve Context Source Registration Subscription with data: and Query Parameter: subscription id set to '\$ {subscription_id}' } then { the SUT sends a valid Response for the operation:	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceSubscriptions/{subscription_id}' method set to 'GET' Retrieve Context Source Registration Subscription with data: and Query Parameter: subscription id set to '\$ {subscription_id}' } then { the SUT sends a valid Response for the operation: Retrieve Context Source Registration Subscription with Response containing:	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceSubscriptions/{subscription_id}' method set to 'GET' Retrieve Context Source Registration Subscription with data: and Query Parameter: subscription id set to '\$ {subscription_id}' } then { the SUT sends a valid Response for the operation: Retrieve Context Source Registration Subscription with Response containing: * Notification expectation file path set to 'notifications/	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceSubscriptions/{subscription_id}' method set to 'GET' Retrieve Context Source Registration Subscription with data: and Query Parameter: subscription id set to '\$ {subscription_id}' } then { the SUT sends a valid Response for the operation: Retrieve Context Source Registration Subscription with Response containing: * Notification expectation file path set to 'notifications/ expectations/1-timesSent-ok.json'	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceSubscriptions/{subscription_id}' method set to 'GET' Retrieve Context Source Registration Subscription with data: and Query Parameter: subscription id set to '\$ {subscription_id}' } then { the SUT sends a valid Response for the operation: Retrieve Context Source Registration Subscription with Response containing: * Notification expectation file path set to 'notifications/ expectations/1-timesSent-ok.json' * Expected Additional Members set to '\$	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceSubscriptions/{subscription_id}' method set to 'GET' Retrieve Context Source Registration Subscription with data: and Query Parameter: subscription id set to '\$ {subscription_id}' } then { the SUT sends a valid Response for the operation: Retrieve Context Source Registration Subscription with Response containing: * Notification expectation file path set to 'notifications/ expectations/1-timesSent-ok.json' * Expected Additional Members set to '\$ {expected_notification_additional_members}'	

TP ld	TP/NGSI-LD/CS/REGSUB/047_01		
Test objective	Check that if the created context source registration subscription defines a timeInterval member, a cSourceNotification will be sent periodically, initially on subscription and when the time interval is reached		
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.11.7		
Config Id	CF_05		
PICS Selection	PICS_NOTIFICATIONS_CSR		
Initial conditions	with { the SUT containing an initial state }		
	Permutations		
TP Permutation id	TP/NGSI-LD/CS/REGSUB/047_01_01		
Name	047_01_01 Receive cSourceNotification Periodically And Initially On Subscription		
Parent Release	v1.3.1		
Description	Check that if the created context source registration subscription defines a timeInterval member, a cSourceNotification will be sent periodically, initially on subscription and when the time interval is reached		

Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/csourceSubscriptions'	
	method set to 'POST'	
	Create Context Source Registration Subscription with filename set to '\$	
	{subscription_payload}', accept set to '\$	
	{EMPTY}', and content-type set to 'application/	
	ld+json'	
	}	
	then {	SUT -> Client
	the client at '\$	
	{endpoint}' receives a valid Notification containing:	
	Notification received After waiting '15' seconds	
	}	

TP Id	TP/NGSI-LD/CS/REGSUB/047_03		
Test objective	Check that instead of providing the original context source registration, implementations		
	should return context source registration information relevant for the subscription, in		
	particular only matching RegistrationInfo elements		
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.11.7		
Config Id	CF_05		
PICS Selection	PICS_NOTIFICATIONS_CSR		
Initial conditions	with { the SUT being in the "initial state" and the SUT containing a Context So	ource	
	Registration Subscription (CSRS1) }		
	Permutations Permutations		
TP Permutation id	TP/NGSI-LD/CS/REGSUB/047_03_01		
Name	047_03_01 Receive cSourceNotification With Relevant Information		
Parent Release	v1.3.1		
Description	Check that instead of providing the original context source registration, impler		
	should return context source registration information relevant for the subscrip-	tion, in	
	particular only matching RegistrationInfo elements		
Expected behaviour	Test events	Direction	
	when {	SUT <- Client	
	the SUT receives a Request from the client containing:		
	URL set to '/		
	ngsi-ld/v1/csourceRegistrations'		
	method set to 'POST'		
	Create Context Source Registration Request with Content-		
	Type set to 'application/		
	Id+json' and payload set to '\$		
	{context_source_registration_payload}'		
	then {	SUT -> Client	
	the client at '\$	301 -> Ciletit	
	{endpoint}' receives a valid Notification containing:		
	Notification received Waiting for Notification and validate it and		
	Query Parameter: expected_subscription_id set to 'subscription_id' and		
	Query Parameter: expected_subscription_id set to subscription_id and		
	'expected_context_source_registration_ids' and		
	Query Parameter: expected_trigger_reason set to 'newlyMatching' and		
	Query Parameter: expected_notification_data_entities set to		
	respected notification data entities and		
	'expected_notification_data_entities' and Query Parameter: timeout set to '5'		

4.3.2.3 DeleteContextSourceRegistrationSubscription

TP Id	TP/NGSI-LD/CS/REGSUB/042_03
Test objective	Check that one cannot delete an unknown context source registration subscription
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.11.6
Config Id	CF_05
PICS Selection	PICS_CsoRegSubDel
Initial conditions	with { the SUT containing an initial state }

Permutations		
TP Permutation id	TP/NGSI-LD/CS/REGSUB/042_03_01	
Name	042_03_01 Delete Unknown Context Source Registration Subscription With Invalid Uri	
Parent Release	v1.3.1	
Description	Check that one cannot delete an unknown context source registration subscrip	otion
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/	SUT <- Client
	ngsi-Id/v1/csourceSubscriptions/{subscription_id}' method set to 'DELETE' Delete Context Source Registration Subscription with id set to 'urn:ngsi-	
	Id:Subscription:unknowSubscription' }	
	then { the SUT sends a valid Response for the operations: Delete Context Source Registration Subscription with Response Status Code set to 404 and	SUT -> Client
	Delete Context Source Registration Subscription with Response Body containing the type 'https://uri.etsi.org/ngsi-ld/errors/ ResourceNotFound' and	
	Delete Context Source Registration Subscription with Response body containing 'title' element }	

TP Id	TP/NGSI-LD/CS/REGSUB/042_01	
Test objective	Check that one can delete a context source registration subscription	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.11.6	
Config Id	CF_05	
PICS Selection	PICS_CsoRegSubDel	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing a Context So	urce
	Registration Subscription (CSRS1) } Permutations	
TP Permutation id	TP/NGSI-LD/CS/REGSUB/042_01_01	
Name	042_01_01 Delete Context Source Registration Subscription	
Parent Release	v1.3.1	
Description	Check that one can delete a context source registration subscription	
Expected behaviour	Test events	Direction
·	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/csourceSubscriptions/{subscription_id}'	
	method set to 'DELETE'	
	Delete Context Source Registration Subscription with id set to '\$	
	{subscription_id}'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Delete Context Source Registration Subscription with Response Status	
	Code set to	
	204 and	
	Retrieve Context Source Registration Subscription with Response	
	Status Code set to	
	\${response1.status_code}	
	 }	

	TP/NGSI-LD/CS/REGSUB/042_02
Test objective	Check that one cannot delete a context source registration subscription with an invalid URI
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.11.6
Config Id	CF_05
PICS Selection	PICS_CsoRegSubDel
Initial conditions	with { the SUT containing an initial state }

Permutations		
TP Permutation id	TP/NGSI-LD/CS/REGSUB/042_02_01	
Name	042_02_01 Delete Context Source Registration Subscription With Invalid Uri	
Parent Release	v1.3.1	
Description	Check that one cannot delete a context source registration subscription with a	n invalid URI
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/csourceSubscriptions/{subscription_id}'	
	method set to 'DELETE'	
	Delete Context Source Registration Subscription with id set to	
	'invalidUri'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Delete Context Source Registration Subscription with Response Status	
	Code set to	
	400 and	
	Delete Context Source Registration Subscription with Response Body	
	containing the type 'https://	
	uri.etsi.org/ngsi-ld/errors/	
	BadRequestData' and	
	Delete Context Source Registration Subscription with Response body	
	containing 'title' element	
	}	

4.3.2.4 CreateContextSourceRegistrationSubscription

TP Id	TP/NGSI-LD/CS/REGSUB/038_02	
Test objective	Check that one can create a context source registration subscription without providing an id	
	and it will be automatically generated	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.11.2	
Config Id	CF_05	
PICS Selection	PICS_CreSubCsoReg	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/REGSUB/038_02_01	
Name	038_02_01 Create Context Source Registration Subscription Without An Id	
Parent Release	v1.3.1	
Description	Check that one can create a context source registration subscription without p	roviding an id
	and it will be automatically generated	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/csourceSubscriptions'	
	method set to 'POST'	
	Create Context Source Registration Subscription with filename set to '\$	
	{subscription_payload}', accept set to '\$	
	[EMPTY]', and content-type set to 'application/	
	\\	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	OO1 > OIICH
	Create Context Source Registration Subscription with Response Status	
	Code set to	
	201 and	
	Retrieve Context Source Registration Subscription with Check Created	
	Resource Set To and	
	Query Parameter: 'ignored_keys' set to '\$	
	{None}' list of keys	
	}	

TP Id	TP/NGSI-LD/CS/REGSUB/038_04	
Test objective Check that one can create a context source registration subscription with isAc		tive member
	set to false and its initial status will be set to "paused"	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.11.2	
Config Id	CF_05	
PICS Selection	PICS_CreSubCsoReg	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/REGSUB/038_04_01	
Name	038_04_01 Create Inactive Context Source Registration Subscription	
Parent Release	v1.3.1	
Description	Check that one can create a context source registration subscription with isAc	tive member
-	set to false and its initial status will be set to "paused"	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/csourceSubscriptions'	
	method set to 'POST'	
	Create Context Source Registration Subscription with filename set to '\$	
	{subscription_payload}', accept set to '\$	
	{EMPTY}', and content-type set to 'application/	
	ld+json'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Create Context Source Registration Subscription with Response Status	
	Code set to	
	201 and	
	Create Context Source Registration Subscription with Response	
	Header: Location containing	
	\$\${subscription_id} and	
	Retrieve Context Source Registration Subscription with Check that the	
	payload body contains a boolean attribute with name	
	'expected_attribute_name=isActive' and value	
	'expected_attribute_value=False and	
	Retrieve Context Source Registration Subscription with Check	
	Response Body containing an Attribute set to and	
	Query Parameter: expected_attribute_name set to 'status' and	
	Query Parameter: response_body set to 'response1.json()' and	
	Query Parameter: expected_attribute_value set to 'paused'	
	}	

TD	TT 11001 LT 100 (DT 00 LT) 1000		
TP ld	TP/NGSI-LD/CS/REGSUB/038_06		
Test objective	Check that one can create a context source registration subscription without a	n expiresAt	
	member and it will be considered as perpetual		
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.11.2		
Config Id	CF_05		
PICS Selection	PICS_CreSubCsoReg		
Initial conditions	with { the SUT containing an initial state }		
	Permutations		
TP Permutation id	TP/NGSI-LD/CS/REGSUB/038_06_01		
Name	038_06_01 Create Context Source Registration Subscription Without expires	At Member	
Parent Release	v1.3.1		
Description	Check that one can create a context source registration subscription without an expiresAt		
	member and it will be considered as perpetual		
Expected behaviour	Test events	Direction	
	when {	SUT <- Client	
	the SUT receives a Request from the client containing:		
	URL set to '/		
	ngsi-ld/v1/csourceSubscriptions'		
	method set to 'POST'		
	Create Context Source Registration Subscription with filename set to '\$		
	{subscription_payload}', accept set to '\$		
	{EMPTY}', and content-type set to 'application/		

ld+json'		
}		
then {		SUT -> Client
the SI	UT sends a valid Response for the operations:	
Cre	eate Context Source Registration Subscription with Response Status	
Code se	et to	
201 a	nd	
Cre	eate Context Source Registration Subscription with Response	
Header:	Location containing	
\$\${subs	cription_id} and	
Ret	trieve Context Source Registration Subscription with Check	
Respons	se Body containing an Attribute set to and	
. (Query Parameter: expected_attribute_name set to 'status' and	
	Query Parameter: response_body set to 'response1.json()' and	
	Query Parameter: expected_attribute_value set to 'active'	
}		

TP Id	TP/NGSI-LD/CS/REGSUB/038_01	
Test objective	Check that one can create a minimal context source registration subscription	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.11.2	
Config Id	CF_05	
PICS Selection	PICS_CreSubCsoReg	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/REGSUB/038_01_01	
Name	038_01_01 Create Context Source Registration Subscription	
Parent Release	v1.3.1	
Description	Check that one can create a minimal context source registration subscription	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceSubscriptions' method set to 'POST' Create Context Source Registration Subscription with filename set to '\$ {subscription_payload}', accept set to '\$ {EMPTY}', and content-type set to 'application/	SUT <- Client

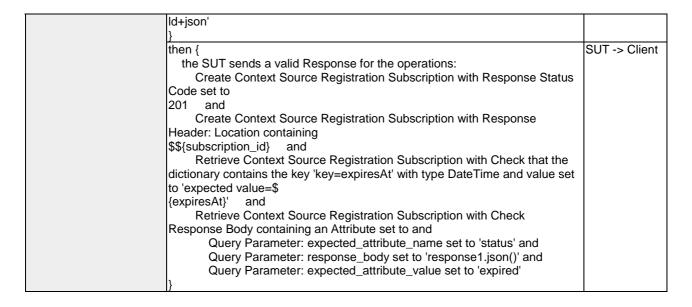
then {	SUT -> Client
the SUT sends a valid Response for the operations:	
Create Context Source Registration Subscription with Response Stat	tus
Code set to	
201 and	
Create Context Source Registration Subscription with Response	
Header: Location containing	
\$\${subscription_id} and	
Retrieve Context Source Registration Subscription with Check Create	ed
Resource Set To and	
Query Parameter: 'ignored_keys' set to '\$	
{None}' list of keys and	
Retrieve Context Source Registration Subscription with The Respons	se l
Body`\$	
{response1.json()}' might contain the key 'lastNotification' in the	
NotificationParams and	
Retrieve Context Source Registration Subscription with The Respons	se
Body`\$	
{response1.json()}' might contain the key 'lastNotification' in the	
NotificationParams and	
Retrieve Context Source Registration Subscription with The Respons	se
Body`\$	
{response1.json()}' might contain the key 'lastNotification' in the	
NotificationParams and	
Retrieve Context Source Registration Subscription with The Respons	эе
Body`\$	
{response1.json()}' might contain the key 'timesSent' in the	
NotificationParams	

TP ld	TP/NGSI-LD/CS/REGSUB/038_03	
Test objective	Check that one can create a context source registration subscription without p	roviding
	isActive member and will be active by default	-
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.11.2	
Config Id	CF_05	
PICS Selection	PICS_CreSubCsoReg	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/REGSUB/038_03_01	
Name	038_03_01 Create Context Source Registration Subscription Without isActive	Member
Parent Release	v1.3.1	
Description	Check that one can create a context source registration subscription without p	roviding
	isActive member and will be active by default	
Expected behaviour	Test events	Direction
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceSubscriptions' method set to 'POST' Create Context Source Registration Subscription with filename set to '\$ {subscription_payload}', accept set to '\$ {EMPTY}', and content-type set to 'application/ Id+json' }	
	then { the SUT sends a valid Response for the operations: Create Context Source Registration Subscription with Response Status Code set to 201 and Create Context Source Registration Subscription with Response Header: Location containing \$\${subscription_id} and Retrieve Context Source Registration Subscription with Check Response Body containing an Attribute set to and Query Parameter: expected_attribute_name set to 'status' and Query Parameter: response_body set to 'response1.json()' and	SUT -> Client

Query Parameter: expected_attribute_value set to 'active'	
}	

TP Id	TP/NGSI-LD/CS/REGSUB/038_07	
Test objective	Check that one cannot create a context source registration subscription where	another
	context source registration subscription whose id is equivalent exists	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.11.2	
Config Id	CF_05	
PICS Selection	PICS_CreSubCsoReg	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing a Context Sou	ırce
	Registration Subscription (CSRS1) }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/REGSUB/038_07_01	
Name	038_07_01 Create Existing Context Source Registration Subscription	
Parent Release	v1.3.1	
Description	Check that one cannot create a context source registration subscription where	another
	context source registration subscription whose id is equivalent exists	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/csourceSubscriptions' method set to 'POST'	
	Create Context Source Registration Subscription with filename set to '\$	
	{subscription_payload}', accept set to '\$	
	{EMPTY}', and content-type set to 'application/	
	Id+json'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Create Context Source Registration Subscription with Response Status	
	Code set to	
	409 and	
	Create Context Source Registration Subscription with Response Body	
	containing the type 'problem_type=\$	
	{ERROR_TYPE_ALREADY_EXISTS}' and Response Body Title is a string	
	and is not \${EMPTY}	
	and Response Body Detail is a string and is not	
	\${EMPTY}	
	and Response Boty Title is not equal to Response Body Detail	
	U	ļ.

TP Id	TP/NGSI-LD/CS/REGSUB/038_05		
Test objective	Check that one can create a context source registration subscription with an ex		
	member and when it is due the status of the subscription changes to "expired"		
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.11.2		
Config Id	CF_05		
PICS Selection	PICS_CreSubCsoReg		
Initial conditions	with { the SUT containing an initial state }		
	Permutations		
TP Permutation id	TP/NGSI-LD/CS/REGSUB/038_05_01		
Name	038_05_01 Create Context Source Registration Subscription With expiresAt Member		
Parent Release	v1.3.1		
Description	Check that one can create a context source registration subscription with an expiresAt		
	member and when it is due the status of the subscription changes to "expired"		
Expected behaviour	Test events	Direction	
	when {	SUT <- Client	
	the SUT receives a Request from the client containing:		
	URL set to '/		
	ngsi-ld/v1/csourceSubscriptions'		
	method set to 'POST'		
	Create Context Source Registration Subscription with filename set to '\$		
	{subscription_payload}', accept set to '\$		
	{EMPTY}', and content-type set to 'application/		



TP Id	TP/NGSI-LD/CS/REGSUB/038_09	
Test objective	Check that one cannot create a context source registration subscription with a	n expiration
	timestamp representing a moment before the current date and time	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.11.2	
Config Id	CF_05	
PICS Selection	PICS_CreSubCsoReg	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/REGSUB/038_09_01	
Name	038_09_01 Create Expired Context Source Registration Subscription	
Parent Release	v1.3.1	
Description	Check that one cannot create a context source registration subscription with a	n expiration
	timestamp representing a moment before the current date and time	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/csourceSubscriptions'	
	method set to 'POST'	
	Create Context Source Registration Subscription with filename set to '\$	
	{subscription_payload}', accept set to '\$	
	{EMPTY}', and content-type set to 'application/	
	[ld+json'	
	<u>}</u>	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Create Context Source Registration Subscription with Response Status	
	Code set to	
	400 and	
	Create Context Source Registration Subscription with Response Body	
	containing the type 'problem_type=\$	
	{ERROR_TYPE_BAD_REQUEST_DATA}' and Response Body Title is a	
	string and is not	
	\${EMPTY}	
	and Response Body Detail is a string and is not \$\{EMPTY\}	
	· · · ·	
	and Response Boty Title is not equal to Response Body Detail	
	I <i>f</i>	

TP ld	TP/NGSI-LD/CS/REGSUB/038_08
Test objective	Check that one cannot create a context source registration subscription If the data types, cardinalities and restrictions expressed by clause 5.2.12 are not met
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.11.2
Config Id	CF_05

when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceSubscriptions' method set to 'POST' Create Context Source Registration Subscription with filename set to '\$ {subscription_payload}', accept set to '\$ {EMPTY}', and content-type set to 'application/ id+json' } then { the SUT sends a valid Response for the operations: Create Context Source Registration Subscription with Response Status Code set to 400 and Create Context Source Registration Subscription with Response Body containing the type 'problem_type=\$ {ERROR_TYPE_BAD_REQUEST_DATA}' and Response Body Title is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Boty Title is not equal to Response Body Detail } TP Permutation id TP/NGSI-LD/CS/REGSUB/038_08_02 Name 038_08_02 InvalidType Parent Release v1.3.1 Description Check that one cannot create a context source registration subscription If the data types, cardinalities and restrictions expressed by clause 5.2.12 are not met Expected behaviour Test events SUT Clie SUT -> Clie SUT -> Clie Test events	PICS Selection	PICS_CreSubCsoReg	
TP Permutation Id TP/NGSI-LD/CS/REGSUB/038_08_01	Initial conditions		
Name 038 08 01 WithoutNotification			
Parent Release	•		
Check that one cannot create a context source registration subscription if the data types, cardinalities and restrictions expressed by clause 5.2.12 are not met when { the SUT receives a Request from the client containing: URL set to 7 ngsi-id-v1/csourceSubscriptions' method set to POST' Create Context Source Registration Subscription with filename set to '\$ (Subscription_payload)', accept set to '\$ (EMPTY), and content-type set to 'application/ ld+json') then { the SUT sends a valid Response for the operations: Create Context Source Registration Subscription with Response Body containing the type 'problem_type=\$ (ERROR_TYPE_BAD_REQUEST_DATA)' and Response Body Title is a string and is not \$ (EMPTY) and Response Body Detail is a string and is not \$ (EMPTY) and Response Body Detail is a string and set one cannot create a context source registration subscription if the data types. September 1,13.1 Check that one cannot create a context source registration subscription if the data types. September 1,13.1 Check that one cannot create a context source registration subscription if the data types. September 1,13.1 Check that one cannot create a context source registration subscription if the data types. September 1,13.1 Check that one cannot create a context source registration subscription if the data types. September 1,13.1 Check that one cannot create a context source registration subscription if the data types. Create Context Source Registration Subscription with filename set to '\$ (Subscription_aptical)', accept set to 'application/ ld+json' Test events Direction Directio			
Cardinalities and restrictions expressed by clause 5.2.12 are not met			
when {	Description	cardinalities and restrictions expressed by clause 5.2.12 are not met	data types,
the SUT receives a Request from the client containing: URL set to 1/ ngsi-idv1/csourceSubscriptions' method set to 1POST Create Context Source Registration Subscription with filename set to '\$ (EMPTY), and content-type set to 'application/ Id+json' } then { the SUT sends a valid Response for the operations: Create Context Source Registration Subscription with Response Status Code set to 400 and Create Context Source Registration Subscription with Response Body containing the type 'problem_type=\$ (ERROR_TYPE_BAD_REQUEST_DATA)' and Response Body Title is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Body Title is not equal to Response Body Detail } TP Permutation id TP/NGSI-LD/CS/REGSUB/038_08_02 Name 038_08_02 InvalidType Parent Release V1.3.1 Check that one cannot create a context source registration subscription If the data types, cardinalities and restrictions expressed by clause 5.2.12 are not met Expected behaviour Expected behaviour Expected behaviour Expected Source Registration Subscription with filename set to '\$ (Subscription_payload)', accept set to '\$ (Subscription_payload)', accept set to '\$ (Subscription_payload)', accept set to '\$ (EMPTY), and content-type set to 'application/ id+json' } then { the SUT sends a valid Response for the operations:	Expected behaviour	Test events	Direction
the SUT sends a valid Response for the operations: Create Context Source Registration Subscription with Response Status Code set to 400 and Create Context Source Registration Subscription with Response Body containing the type 'problem_type=\$ [ERROR_TYPE_BAD_REQUEST_DATA)' and Response Body Title is a string and is not \$[EMPTY] and Response Body Detail is a string and is not \$[EMPTY] and Response Body Detail is a string and is not \$[EMPTY] and Response Body Detail is a string and is possible to the string and is not \$[EMPTY] and Response Body Detail is a string and is not \$[EMPTY] and Response Body Title is not equal to Response Body Detail] TP Permutation id TP/NGSI-LD/CS/REGSUB/038_08_02 Parent Release V1.3.1 Description Check that one cannot create a context source registration subscription If the data types, cardinalities and restrictions expressed by clause 5.2.12 are not met Expected behaviour Test events when { the SUT receives a Request from the client containing:		the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceSubscriptions' method set to 'POST' Create Context Source Registration Subscription with filename set to '\$ {subscription_payload}', accept set to '\$ {EMPTY}', and content-type set to 'application/ Id+json' }	SUT <- Client
Name		the SUT sends a valid Response for the operations: Create Context Source Registration Subscription with Response Status Code set to 400 and Create Context Source Registration Subscription with Response Body containing the type 'problem_type=\$ {ERROR_TYPE_BAD_REQUEST_DATA}' and Response Body Title is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY}	SUI -> Client
Name	TP Permutation id	TP/NGSI-LD/CS/REGSUB/038_08_02	
Parent Release V1.3.1 Check that one cannot create a context source registration subscription If the data types, cardinalities and restrictions expressed by clause 5.2.12 are not met Test events Direction	Name		
Expected behaviour Test events Direction when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceSubscriptions' method set to 'POST' Create Context Source Registration Subscription with filename set to '\$ {EMPTY}', and content-type set to 'application/ ld+json' } then { the SUT sends a valid Response for the operations: Create Context Source Registration Subscription with Response Status Code set to 400 and Create Context Source Registration Subscription with Response Body containing the type 'problem_type=\$ {ERROR_TYPE_BAD_REQUEST_DATA}' and Response Body Title is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Boty Title is not equal to Response Body Detail }	Parent Release		
Expected behaviour Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceSubscriptions' method set to 'POST' Create Context Source Registration Subscription with filename set to '\$ {subscription_payload}', accept set to '\$ {EMPTY}', and content-type set to 'application/ Id+json' } then { the SUT sends a valid Response for the operations: Create Context Source Registration Subscription with Response Status Code set to 400 and Create Context Source Registration Subscription with Response Body containing the type 'problem_type=\$ {ERROR_TYPE_BAD_REQUEST_DATA}' and Response Body Title is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Title is not equal to Response Body Detail }	Description		data types,
when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceSubscriptions' method set to 'POST' Create Context Source Registration Subscription with filename set to '\$ {subscription_payload}', accept set to '\$ {EMPTY}', and content-type set to 'application/ ld+json' } then { the SUT sends a valid Response for the operations: Create Context Source Registration Subscription with Response Status Code set to 400 and Create Context Source Registration Subscription with Response Body containing the type 'problem_type=\$ {ERROR_TYPE_BAD_REQUEST_DATA}' and Response Body Title is a string and is not \$(EMPTY) and Response Body Detail is a string and is not \$(EMPTY) and Response Boty Title is not equal to Response Body Detail }	Expected behaviour		Direction
the SUT sends a valid Response for the operations:		the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceSubscriptions' method set to 'POST' Create Context Source Registration Subscription with filename set to '\$ {subscription_payload}', accept set to '\$ {EMPTY}', and content-type set to 'application/	SUT <- Client
TP Permutation id TP/NGSI-LD/CS/REGSUB/038_08_03		the SUT sends a valid Response for the operations: Create Context Source Registration Subscription with Response Status Code set to 400 and Create Context Source Registration Subscription with Response Body containing the type 'problem_type=\$ {ERROR_TYPE_BAD_REQUEST_DATA}' and Response Body Title is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY}	SUI -> Client
1 1 1 1 1 1 1 1 1 1	TP Permutation id]) TP/NGSI-LD/CS/REGSUB/038_08_03	
Name 038_08_03 InvalidQuery			
Parent Release v1.3.1			

Description	Check that one cannot create a context source registration subscription If the cardinalities and restrictions expressed by clause 5.2.12 are not met	data types,
Expected behaviour	Test events	Direction
Expected benaviour		
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/csourceSubscriptions'	
	method set to 'POST'	
	Create Context Source Registration Subscription with filename set to '\$	
	{subscription_payload}', accept set to '\$	
	{EMPTY}', and content-type set to 'application/	
	ld+json'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Create Context Source Registration Subscription with Response Status	
	Code set to	
	400 and	
	Create Context Source Registration Subscription with Response Body	
	containing the type 'problem_type=\$	
	{ERROR_TYPE_BAD_REQUEST_DATA}' and Response Body Title is a	
	string and is not	
	\${EMPTY}	
	and Response Body Detail is a string and is not	
	\${EMPTY}	
	and Response Boty Title is not equal to Response Body Detail	
TD Downstation id	} TD/NOCL D/OC/DECOLID/000 00 04	
TP Permutation id Name	TP/NGSI-LD/CS/REGSUB/038_08_04 038_08_04 EmptyWatchedAttributes	
Parent Release		
	V1.3.1	data tunaa
Description	Check that one cannot create a context source registration subscription If the cardinalities and restrictions expressed by clause 5.2.12 are not met	uaia iypes,
Expected behaviour	Test events	Direction
Expedica Bellavious	when {	SUT <- Client
	the SUT receives a Request from the client containing:	OUT < Ollerit
	URL set to '/	
	ngsi-ld/v1/csourceSubscriptions'	
	method set to 'POST'	
	Create Context Source Registration Subscription with filename set to '\$	
	{subscription_payload}', accept set to '\$	
	{EMPTY}', and content-type set to 'application/	
	[ld+json'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Create Context Source Registration Subscription with Response Status	
	Code set to	
	400 and	
	Create Context Source Registration Subscription with Response Body	
	containing the type 'problem_type=\$	
	{ERROR_TYPE_BAD_REQUEST_DATA}' and Response Body Title is a	
	string and is not	
	\${EMPTY}	
	and Response Body Detail is a string and is not	
	\${EMPTY}	
	and Response Boty Title is not equal to Response Body Detail	
	}	
		•

4.3.2.5 UpdateContextSourceRegistrationSubscription

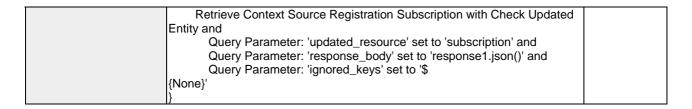
TP Id	TP/NGSI-LD/CS/REGSUB/039_02
Test objective	Check that one cannot update a context source registration subscription with an invalid URI
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.11.3
Config Id	CF_05
PICS Selection	PICS_CsoRegSubUpd
Initial conditions	with { the SUT containing an initial state }

Permutations		
TP Permutation id	TP/NGSI-LD/CS/REGSUB/039_02_01	
Name	039_02_01 Update Context Source Registration Subscription With Invalid Uri	
Parent Release	v1.3.1	
Description	Check that one cannot update a context source registration subscription with a	an invalid URI
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceSubscriptions/{subscription_id}' method set to 'PATCH' Update Context Source Registration Subscription with subscription id set to 'invalidUri' and subscription update fragment set to '\$ {subscription_update_fragment}' } then {	SUT <- Client
	the SUT sends a valid Response for the operations: Update Context Source Registration Subscription with Response Status Code set to 400 and Update Context Source Registration Subscription with Response Body containing the type 'problem_type=\$ {ERROR_TYPE_BAD_REQUEST_DATA}' and Response Body Title is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Title is not equal to Response Body Detail }	301 -> Glient

TP Id	TP/NGSI-LD/CS/REGSUB/039_04	
Test objective	Check that one cannot update a context source registration subscription with a	a fragment that
	does not meet the data types and restrictions expressed by clause 5.2.12	· ·
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.11.3	
Config Id	CF_05	
PICS Selection	PICS_CsoRegSubUpd	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing a set of Conte Registration Subscriptions (CSRSs) }	ext Source
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/REGSUB/039_04_01	
Name	039_04_01 InvalidType	
Parent Release	v1.3.1	
Description	Check that one cannot update a context source registration subscription with a does not meet the data types and restrictions expressed by clause 5.2.12	a fragment that
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceSubscriptions/{subscription_id}' method set to 'PATCH' Update Context Source Registration Subscription with subscription id set to '\$ {subscription_id}' and subscription update fragment set to '\$ {subscription_update_fragment}' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Update Context Source Registration Subscription with Response Status Code set to 400 and Update Context Source Registration Subscription with Response Body containing the type 'problem_type=\$ {ERROR_TYPE_BAD_REQUEST_DATA}' and Response Body Title is a string and is not \${EMPTY} and Response Body Detail is a string and is not	SUT -> Client

	\${EMPTY} and Response Boty Title is not equal to Response Body Detail	
	}	
TP Permutation id	TP/NGSI-LD/CS/REGSUB/039_04_02	
Name	039_04_02 InvalidNotification	
Parent Release	v1.3.1	
Description	Check that one cannot update a context source registration subscription with a does not meet the data types and restrictions expressed by clause 5.2.12	a fragment that
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceSubscriptions/{subscription_id}' method set to 'PATCH' Update Context Source Registration Subscription with subscription id set to '\$ {subscription_id}' and subscription update fragment set to '\$ {subscription_update_fragment}' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Update Context Source Registration Subscription with Response Status Code set to 400 and Update Context Source Registration Subscription with Response Body containing the type 'problem_type=\$ {ERROR_TYPE_BAD_REQUEST_DATA}' and Response Body Title is a string and is not \${EMPTY} and Response Body Detail is a string and is not \${EMPTY} and Response Body Title is not equal to Response Body Detail }	SUT -> Client

TP Id	TD/NCCLLD/CC/DECCLID/020 04	
	TP/NGSI-LD/CS/REGSUB/039_01	
Test objective	Check that one can update a context source registration subscription	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.11.3	
Config Id	CF_05	
PICS Selection	PICS_CsoRegSubUpd	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing a set of Conte	ext Source
	Registration Subscriptions (CSRSs) }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/REGSUB/039_01_01	
Name	039_01_01 Update Context Source Registration Subscription	
Parent Release	v1.3.1	
Description	Check that one can update a context source registration subscription	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceSubscriptions/{subscription_id}' method set to 'PATCH' Update Context Source Registration Subscription with subscription id set to '\$ {subscription_id}' and subscription update fragment set to '\$ {subscription_update_fragment}' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Update Context Source Registration Subscription with Response Status Code set to 204 and Update Context Source Registration Subscription with Response does not contain a body and Update Context Source Registration Subscription with Response reason set to 'No Content' and	SUT -> Client



TP Id	TP/NGSI-LD/CS/REGSUB/039_03	
Test objective	Check that one cannot update an unknown context source registration subscri	ption
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.11.3	
Config Id	CF_05	
PICS Selection	PICS_CsoRegSubUpd	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/REGSUB/039_03_01	
Name	039_03_01 Update Unknown Context Source Registration Subscription	
Parent Release	v1.3.1	
Description	Check that one cannot update an unknown context source registration subscri	ption
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/csourceSubscriptions/{subscription_id}'	
	method set to 'PATCH'	
	Update Context Source Registration Subscription with subscription id	
	set to 'urn:ngsi-	
	Id:Subscription:unknowSubscription' and subscription update fragment set to	
	 '\$	
	{subscription_update_fragment}'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Update Context Source Registration Subscription with Response Status	
	Code set to	
	404 and	
	Update Context Source Registration Subscription with Response Body	
	containing the type 'problem_type=\$	
	{ERROR_TYPE_RESOURCE_NOT_FOUND}' and Response Body Title is a	
	string and is not	
	\${EMPTY}	
	and Response Body Detail is a string and is not	
	\$\{EMPTY\}	
	and Response Boty Title is not equal to Response Body Detail	
	}	

TP Id	TP/NGSI-LD/CS/REGSUB/039_05		
Test objective	Check that one cannot update a context source registration subscription with a	ın invalid	
	request body (invalid JSON document)		
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.11.3		
Config Id	CF_05		
PICS Selection	PICS_CsoRegSubUpd		
Initial conditions	with { the SUT being in the "initial state" and the SUT containing a set of Conte	ext Source	
	Registration Subscriptions (CSRSs) }		
	Permutations		
TP Permutation id	TP/NGSI-LD/CS/REGSUB/039_05_01		
Name	039_05_01 Update Context Source Registration Subscription With Invalid JSC	N Fragment	
Parent Release	v1.3.1		
Description	Check that one cannot update a context source registration subscription with an invalid		
	request body (invalid JSON document)		
Expected behaviour	Test events	Direction	
	when {	SUT <- Client	
	the SUT receives a Request from the client containing:		
	URL set to '/		
	ngsi-ld/v1/csourceSubscriptions/{subscription_id}'		

	1
method set to 'PATCH'	
Update Context Source Registration Subscription from file with	
subscription id set to '\$	
{subscription_id}' and subscription update fragment set to	
'csourceSubscriptions/	
fragments/subscription-update-invalid-json-sample.json'	
]	
then {	SUT -> Client
the SUT sends a valid Response for the operations:	
Update Context Source Registration Subscription From File with	
Response Status Code set to	
400 and	
Update Context Source Registration Subscription From File with	
Response Body containing the type 'problem_type=\$	
{ERROR_TYPE_BAD_REQUEST_DATA}' and Response Body Title is a	
string and is not	
\${EMPTY}	
and Response Body Detail is a string and is not	
\${EMPTY}	
and Response Boty Title is not equal to Response Body Detail	
}	

4.3.2.6 QueryContextSourceRegistrationSubscriptions

TP Id	TP/NGSI-LD/CS/REGSUB/041_01	
Test objective	Check that one can query context source registration subscriptions	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.11.5	
Config Id	CF_05	
PICS Selection	PICS_RetLisSubCsoReg	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing a set of Conte	ext Source
	Registration Subscriptions (CSRSs) }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/REGSUB/041_01_01	
Name	041_01_01 Query Context Source Registration Subscriptions	
Parent Release	v1.3.1	
Description	Check that one can query context source registration subscriptions	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/csourceSubscriptions'	
	method set to 'GET'	
	Query Context Source Registration Subscriptions and	
	Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	jsonld-contexts/ngsi-ld-test-suite-compound.jsonld'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Query Context Source Registration Subscriptions with Response Status	
	Code set to	
	200 and	
	Query Context Source Registration Subscriptions with Response	
	containing:	
	* file set to 'csourceSubscriptions/	
	expectations/subscriptions-035-01-expectation.json' * id set to '\$	
	{subscription_ids}'	
	* response set to '\$	
	{response.json()}'	
]	

TP Id	TP/NGSI-LD/CS/REGSUB/041_03
Test objective	Check that one can query context source registration subscriptions with providing page and
	limit parameters for pagination
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.11.5
Config Id	CF_05

PICS Selection	PICS_RetLisSubCsoReg	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing a set of Conto	ext Source
	Registration Subscriptions (CSRSs) }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/REGSUB/041_03_01	
Name	041_03_01 Query Second Subscription	
Parent Release	v1.3.1	
Description	Check that one can query context source registration subscriptions with provide	ling page and
	limit parameters for pagination	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/	
	ngsi-ld/v1/csourceSubscriptions' method set to 'GET'	
	Query Context Source Registration Subscriptions and	
	Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and	
	Query Parameter: limit set to '\${1}	
	'and	
	Query Parameter: page set to '\${2}	
	'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Query Context Source Registration Subscriptions with Response Status	
	Code set to	
	200 and	
	Query Context Source Registration Subscriptions with Response Body	
	containing a list of entities equal to '\${1}	
	of type 'Subscription' with response set to '\$ {response.json()}' and	
	Query Context Source Registration Subscriptions with Response	
	header with 'Prev' header equal to ' </th <th></th>	
	ngsi-Id/v1/csourceSubscriptions?	
	limit=1&page=3>;rel="next";type="application/	
	ld+json" and with 'Next' header equal to '\$	
	{response.headers}'	
)	
TP Permutation id	TP/NGSI-LD/CS/REGSUB/041_03_02	
Name	041_03_02 Query Last Subscription	
Parent Release	v1.3.1	
Description	Check that one can query context source registration subscriptions with provide	ling page and
	limit parameters for pagination	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/csourceSubscriptions' method set to 'GET'	
	Query Context Source Registration Subscriptions and	
	Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and	
	Query Parameter: limit set to '\${2}	
	'and	
	Query Parameter: page set to '\${2}	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Query Context Source Registration Subscriptions with Response Status	
	Code set to	
	200 and	
	Query Context Source Registration Subscriptions with Response Body	
	containing a list of entities equal to '\${1}	
	of type 'Subscription' with response set to '\$	

		1
	{response.json()}' and	
	Query Context Source Registration Subscriptions with Response	
	header with 'Prev' header equal to '\$	
	{EMPTY}' and with 'Next' header equal to '\$	
	{response.headers}'	
	}	
TP Permutation id	TP/NGSI-LD/CS/REGSUB/041_03_03	
Name	041_03_03 Query All Subscriptions	
Parent Release	v1.3.1	
Description	Check that one can query context source registration subscriptions with provice	ling page and
	limit parameters for pagination	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/csourceSubscriptions'	
	method set to 'GET'	
	Query Context Source Registration Subscriptions and	
	Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and	
	Query Parameter: limit set to '\${15}	
	'and	
	Query Parameter: page set to '\${1}	
	'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	0.0.1
	Query Context Source Registration Subscriptions with Response Status	
	Code set to	
	200 and	
	Query Context Source Registration Subscriptions with Response Body	
	containing a list of entities equal to '\${3}	
	of type 'Subscription' with response set to '\$	
	{response.ison()}' and	
	Query Context Source Registration Subscriptions with Response	
	header with 'Prev' header equal to '\$	
	(EMPTY)' and with 'Next' header equal to '\$	
	{response.headers}'	
	}	

TDII	TTD/N 0.00 / D 1.00 / D 5.00 /	
TP Id	TP/NGSI-LD/CS/REGSUB/041_02	
Test objective	Check that one can query context source registration subscriptions with a limit	parameter
	and it will be the maximum number of subscriptions to be retrieved	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.11.5	
Config Id	CF_05	
PICS Selection	PICS_RetLisSubCsoReg	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing a set of Conte	ext Source
	Registration Subscriptions (CSRSs) }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/REGSUB/041_02_01	
Name	041_02_01 Query One Subscription	
Parent Release	v1.3.1	
Description	Check that one can query context source registration subscriptions with a limit parameter	
	and it will be the maximum number of subscriptions to be retrieved	
Expected behaviour	Test events Direction	
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/csourceSubscriptions'	
	method set to 'GET'	
	Query Context Source Registration Subscriptions and	
	Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and	
	Query Parameter: limit set to '\${1}	

	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Query Context Source Registration Subscriptions with Response Status	
	Code set to	
	200 and	
	Query Context Source Registration Subscriptions with Response Body	
	containing a list of entities equal to '\${1}	
	of type 'Subscription' with response set to '\$	
	{response.json()}'	
	}	
TP Permutation id	TP/NGSI-LD/CS/REGSUB/041_02_02	
Name	041_02_02 Query Two Subscription	
Parent Release	v1.3.1	
Description	Check that one can query context source registration subscriptions with a limit	parameter
	and it will be the maximum number of subscriptions to be retrieved	
Expected behaviour	Test events	Direction
-	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/csourceSubscriptions'	
	method set to 'GET'	1
	Query Context Source Registration Subscriptions and	
	Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	jsonId-contexts/ngsi-Id-test-suite-compound.jsonId' and	
	Query Parameter: limit set to '\${2}	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Query Context Source Registration Subscriptions with Response Status	
	The state of the s	
	Code set to	
	200 and	
	Query Context Source Registration Subscriptions with Response Body	
	containing a list of entities equal to '\${2}	
	of type 'Subscription' with response set to '\$	
	{response.json()}'	
TD Day of the Co	TD N CO	
TP Permutation id	TP/NGSI-LD/CS/REGSUB/041_02_03	
Name	041_02_03 Query All Subscriptions	
Parent Release	v1.3.1	
Description	Check that one can query context source registration subscriptions with a limit	narameter
2000	and it will be the maximum number of subscriptions to be retrieved	paramotor
Expected behaviour		Direction
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	1
	ngsi-ld/v1/csourceSubscriptions'	
	method set to 'GET'	1
	Query Context Source Registration Subscriptions and	
	Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and	
	Query Parameter: limit set to '\${15}	
	13	
	thon (SUT -> Client
	then {	Jour -> Client
	the SUT sends a valid Response for the operations:	
	Query Context Source Registration Subscriptions with Response Status	
	Code set to	
	200 and	1
	Query Context Source Registration Subscriptions with Response Body	
	containing a list of entities equal to '\${3}	
	of type 'Subscription' with response set to '\$	
	{response.json()}'	1
	j, ' ''	
	- 0	1

TDIA	TD/NOOL D/OC/DEOCUD/OAA OA		
TP Id	TP/NGSI-LD/CS/REGSUB/041_04	ralid naga and	
Test objective	Check that one cannot query context source registration subscriptions with inv limit parameters	and page and	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.11.5		
Config Id	CF_05		
PICS Selection	PICS_RetLisSubCsoReg		
Initial conditions	with { the SUT containing an initial state }		
	Permutations		
TP Permutation id	TP/NGSI-LD/CS/REGSUB/041_04_01		
Name	041_04_01 Invalid Limit		
Parent Release	v1.3.1		
Description	Check that one cannot query context source registration subscriptions with inv limit parameters	Check that one cannot query context source registration subscriptions with invalid page and	
Expected behaviour	Test events	Direction	
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-Id/v1/csourceSubscriptions' method set to 'GET' Query Context Source Registration Subscriptions and Query Parameter: limit set to '\${-5}' and Query Parameter: page set to '\${2}' } then { the SUT sends a valid Response for the operations: Query Context Source Registration Subscriptions with Response Status Code set to 400 and Query Context Source Registration Subscriptions with Response Body containing the type 'https:// uri.etsi.org/ngsi-Id/errors/ BadRequestData' and Query Context Source Registration Subscriptions with Response body containing 'title' element	SUT <- Client SUT -> Client	
	}		
TP Permutation id	TP/NGSI-LD/CS/REGSUB/041_04_02		
Name	041_04_02 Invalid Page		
Parent Release	v1.3.1		
Description	Check that one cannot query context source registration subscriptions with invilimit parameters	alid page and	
Expected behaviour	Test events	Direction	
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-Id/v1/csourceSubscriptions' method set to 'GET' Query Context Source Registration Subscriptions and Query Parameter: limit set to '\${2}' and Query Parameter: page set to '\${-3}' }	SUT <- Client	
	then { the SUT sends a valid Response for the operations: Query Context Source Registration Subscriptions with Response Status Code set to 400 and Query Context Source Registration Subscriptions with Response Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/ BadRequestData' and Query Context Source Registration Subscriptions with Response body containing 'title' element }	SUT -> Client	
TP Permutation id	TP/NGSI-LD/CS/REGSUB/041_04_03	•	
Name	041_04_03 Invalid Limit And Page		
Parent Release	v1.3.1		
Description	Check that one cannot query context source registration subscriptions with inv	alid page and	
-	limit parameters	. •	

Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ngsi-ld/v1/csourceSubscriptions'	
	method set to 'GET'	
	Query Context Source Registration Subscriptions and	
	Query Parameter: limit set to '\${0}' and	
	Query Parameter: page set to '\${0}'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Query Context Source Registration Subscriptions with Response Status	
	Code set to	
	400 and	
	Query Context Source Registration Subscriptions with Response Body	
	containing the type 'https://	
	uri.etsi.org/ngsi-ld/errors/	
	BadRequestData' and	
	Query Context Source Registration Subscriptions with Response body	
	containing 'title' element	
	}	

4.3.3 Discovery

4.3.3.1 RetrieveContextSourceRegistration

TP ld	TP/NGSI-LD/CS/DISC/036_01	
Test objective	Check that one cannot retrieve a Context Source Registration, if the context source	
	registration id is not a valid URI	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.10.1	
Config Id	CF_05	
PICS Selection	PICS_CsoRegRet	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/DISC/036_01_01	
Name	036_01_01 Invalid Id	
Parent Release	v1.3.1	
Description	Check that one cannot retrieve a Context Source Registration, if the context s	source
	registration id is not a valid URI	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/csourceRegistrations/{context_source_registration_id}'	
	method set to 'GET'	
	Retrieve Context Source Registration with data: and	
	Query Parameter: context source registration id set to 'invalidUri'	
	}	01.17
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Retrieve Context Source Registration with Response Status Code set	
	to 400 and	
	Retrieve Context Source Registration with Response Body containing	
	the type 'https://	
	uri.etsi.org/ngsi-ld/errors/	
	BadRequestData' and	
	Retrieve Context Source Registration with Response body containing	
	'title' element	
	}	
		1

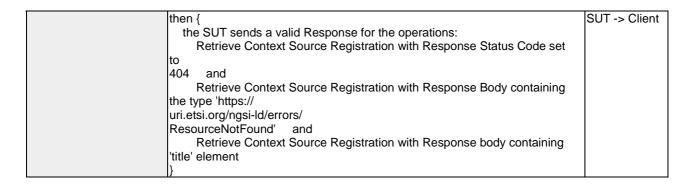
TP ld	TP/NGSI-LD/CS/DISC/036_03
Test objective	Check that one can retrieve a Context Source Registration. Term to URI expansion of
	Attribute names shall be observed.

Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.10.1	
Config Id	CF_05	
PICS Selection	PICS_CsoRegRet	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/DISC/036_03_01	
Name	036_03_01 Retrieve Context Source Registration With Default Core Context	
Parent Release	v1.3.1	
Description	Check that one can retrieve a Context Source Registration. Term to URI expan	nsion of
	Attribute names shall be observed.	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/csourceRegistrations/{context_source_registration_id}'	
	method set to 'GET'	
	Retrieve Context Source Registration with data: and	
	Query Parameter: context source registration id set to '\$	
	{context_source_registration_id}'	
	}	OUT OF C
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Retrieve Context Source Registration with Response Status Code set	
	to	
	Retrieve Context Source Registration with Response containing:	
	* file set to 'csourceRegistrations/	
	expectations/context-source-registration-expanded-format.json'	
	* id set to '\$	
	{context_source_registration_id}'	
	* response set to '\$	
	{response.json()}'	
	}	
	}	

TP ld	TP/NGSI-LD/CS/DISC/036_05		
Test objective	Check that the JSON-LD @context is obtained from a Link header if present	and that the	
_	default JSON-LD @context is used if not present		
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 6.3.5		
Config Id	CF_05		
PICS Selection	PICS_HTTP_CONTEXT_RESOLV		
Initial conditions	with { the SUT being in the "initial state" and the SUT contains a Context Sou	rce	
		Registration with id equal to \${context_source_registration_id} and payload set to	
	\${context_source_registration_payload_file_path}}		
	Permutations		
TP Permutation id	TP/NGSI-LD/CS/DISC/036_05_01		
Name	036_05_01 EmptyJsonLdContext		
Parent Release	v1.3.1		
Description	Check that the JSON-LD @context is obtained from a Link header if present and that the		
	default JSON-LD @context is used if not present		
Expected behaviour	Test events	Direction	
	when {	SUT <- Client	
	the SUT receives a Request from the client containing:		
	URL set to '/		
	ngsi-ld/v1/csourceRegistrations/{context_source_registration_id}'		
	method set to 'GET'		
	Retrieve Context Source Registration with data: and		
	Query Parameter: context source registration id set to '\$		
	{context_source_registration_id}' and		
	Query Parameter: context set to '\$		
	(EMPTY)'		
	then {	SUT -> Client	
	the SUT sends a valid Response for the operations:	301 -> Client	
	Retrieve Context Source Registration with Response Status Code set		
	to		
	200 and		

	Retrieve Context Source Registration with Response containing:	
	* file set to 'csourceRegistrations/	
	expectations/context-source-registration-expanded-format.json'	
	* id set to '\$	
	{context_source_registration_id}'	
	* response set to '\$	
	{response.json()}'	
	}	
TP Permutation id	TP/NGSI-LD/CS/DISC/036_05_02	
Name	036_05_02 CreationTimeJsonLdContext	
Parent Release	v1.3.1	
Description	Check that the JSON-LD @context is obtained from a Link header if present a	nd that the
	default JSON-LD @context is used if not present	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/csourceRegistrations/{context_source_registration_id}'	
	method set to 'GET'	
	Retrieve Context Source Registration with data: and	
	Query Parameter: context source registration id set to '\$	
	{context_source_registration_id}' and	
	Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	jsonld-contexts/ngsi-ld-test-suite-compound.jsonld'	
]	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Retrieve Context Source Registration with Response Status Code set	
	to	
	200 and	
	Retrieve Context Source Registration with Response containing:	
	* file set to 'csourceRegistrations/	
	expectations/context-source-registration.json'	
	* id set to '\$	
	{context_source_registration_id}'	
	* response set to '\$	
	{response.json()}'	
	}	

TP Id	TP/NGSI-LD/CS/DISC/036 02		
	<u>-</u>		
Test objective	Check that one cannot retrieve a Context Source Registration, if the NGSI-LD		
	not know about the target context source registration, because there is no exist	sting context	
	source registration whose id (URI) is equivalent		
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.10.1		
Config Id	CF_05		
PICS Selection	PICS_CsoRegRet		
Initial conditions	with { the SUT containing an initial state }		
	Permutations		
TP Permutation id	TP/NGSI-LD/CS/DISC/036_02_01		
Name	036_02_01 Retrieve Unknown Context Source Registration		
Parent Release	v1.3.1		
Description	Check that one cannot retrieve a Context Source Registration, if the NGSI-LD endpoint does		
	not know about the target context source registration, because there is no existing context		
	source registration whose id (URI) is equivalent	-	
Expected behaviour	Test events	Direction	
	when {	SUT <- Client	
	the SUT receives a Request from the client containing:		
	URL set to '/		
	ngsi-ld/v1/csourceRegistrations/{context_source_registration_id}'		
	method set to 'GET'		
	Retrieve Context Source Registration with data: and		
	Query Parameter: context source registration id set to 'urn:ngsi-		
	Id:ContextSourceRegistration:unknowRegistration'		
	}		
	IJ		



TP Id	TP/NGSI-LD/CS/DISC/036_04	
Test objective	Check that one can retrieve a Context Source Registration	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.10.1	
Config Id	CF_05	
PICS Selection	PICS_CsoRegRet	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/DISC/036_04_01	
Name	036_04_01 Retrieve Context Source Registration	
Parent Release	v1.3.1	
Description	Check that one can retrieve a Context Source Registration	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/csourceRegistrations/{context_source_registration_id}'	
	method set to 'GET'	
	Retrieve Context Source Registration with data: and	
	Query Parameter: context source registration id set to '\$	
	{context_source_registration_id}' and	
	Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	jsonld-contexts/ngsi-ld-test-suite-compound.jsonld'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Retrieve Context Source Registration with Response Status Code set	
	to	
	200 and	
	Retrieve Context Source Registration with Response containing:	
	* file set to 'csourceRegistrations/	
	expectations/context-source-registration.json' * id set to '\$	
	{context_source_registration_id}'	
	* response set to '\$	
	{response.json()}'	
	\tag{\tag{\tag{\tag{\tag{\tag{\tag{	
	If	

4.3.3.2 QueryContextSourceRegistrations

TP Id	TP/NGSI-LD/CS/DISC/037_08
Test objective	Check that one can query context source registrations. If no temporal query is present, only Context Source Registrations for Context Sources providing latest information are considered
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.10.2
Config Id	CF_05
PICS Selection	PICS_DisCsoReg
Initial conditions	with { the SUT containing an initial state }
Permutations Permutations	
TP Permutation id	TP/NGSI-LD/CS/DISC/037_08_01
Name	037_08_01 Query Context Source Registration Without Temporal Query

Parent Release	v1.3.1	
Description	Check that one can query context source registrations. If no temporal query is Context Source Registrations for Context Sources providing latest information considered	
Expected behaviour	Test events	Direction
Expected Bellaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: type set to 'Vehicle' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Query Context Source Registrations with Response Status Code set to 200 and Notification with Response body set to list of all matching Context Source Registrations resolved against the default JSON- LD context }	SUT -> Client

TD 1 1		
TP Id	TP/NGSI-LD/CS/DISC/037_11	
Test objective	Check that one can query context source registrations with providing page and	d limit
	parameters, pagination logic shall be in place as mandated by clause 5.5.9.	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.10.2	
Config Id	CF_05	
PICS Selection	PICS_DisCsoReg	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing a list of Conte	ext Source
	Registrations (CSRs) providing latest information about some entities }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/DISC/037_11_01	
Name	037_11_01 Query Second Subscription	
Parent Release	v1.3.1	
Description	Check that one can query context source registrations with providing page and	d limit
	parameters, pagination logic shall be in place as mandated by clause 5.5.9.	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/csourceRegistrations'	
	method set to 'GET'	
	Retrieve Temporal Representation of Entity and	
	Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	jsonId-contexts/ngsi-Id-test-suite-compound.jsonId' and	
	Query Parameter: type set to 'Building' and Query Parameter: limit set to '\${1}	
	'and	
	Query Parameter: offset set to '\${2}	
	Query Farameter. Onset set to \${2}	
	1	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	OUT -> OHEIR
	Query Context Source Registrations with Response Status Code set to	
	200 and	
	Query Context Source Registrations with Response Body containing a	
	list of entities equal to '\${1}	
	of type 'ContextSourceRegistration' with response set to '\$	
	{response.json()}' and	
	Query Context Source Registrations with Response header with 'Prev'	
	header equal to ' </th <th></th>	
	ngsi-ld/v1/csourceSubscriptions?	
	type=Building&limit=1&offset=3>;rel="next";type="application/	
	171	1

	11.1 W. 1.1 M. 1.1 M. 1.1 M.	1
	Id+json" and with 'Next' header equal to '\$	
	{response.json()}'	
TP Permutation id	TP/NGSI-LD/CS/DISC/037_11_02	
Name	037_11_02 Query Last Subscription	
Parent Release	v1.3.1	
Description	Check that one can query context source registrations with providing page and limit	
	parameters, pagination logic shall be in place as mandated by clause 5.5.9.	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: type set to 'Building' and Query Parameter: limit set to '\${2}	SUT <- Client
	I' and	
	Query Parameter: offset set to '\${2}	
	}	
	then { the SUT sends a valid Response for the operations: Query Context Source Registrations with Response Status Code set to 200 and	SUT -> Client
	Query Context Source Registrations with Response Body containing a list of entities equal to '\${1} ' of type 'ContextSourceRegistration' with response set to '\$	
	{response.json()}' and Query Context Source Registrations with Response header with 'Prev' header equal to '\$	
	{EMPTY}' and with 'Next' header equal to '\$ {response.json()}'	
TP Permutation id	TP/NGSI-LD/CS/DISC/037_11_03	
Name	037_11_03 Query All Subscriptions	
Parent Release	v1.3.1	
Description	Check that one can query context source registrations with providing page an parameters, pagination logic shall be in place as mandated by clause 5.5.9.	
Expected behaviour	Test events	Direction SUT <- Client
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: type set to 'Building' and Query Parameter: limit set to '\${15} ' and Query Parameter: offset set to '\${0}	
	then { the SUT sends a valid Response for the operations: Query Context Source Registrations with Response Status Code set to	SUT -> Client
	200 and Query Context Source Registrations with Response Body containing a list of entities equal to '\${2} ' of type 'ContextSourceRegistration' with response set to '\$ {response.json()}' and Query Context Source Registrations with Response header with 'Prev'	
	header equal to '\$	

{EMPTY}' and with 'Next' header equal to '\$	
{response.json()}'	
}	

TP/NGSI-LD/CS/DISC/037_09		
Test objective Check that one can query context source registrations. If present, the temp	oral query is	
matched against the observationInterval or the managementInterval		
Reference ETSI GS CIM 009 V1.5.1 [1], clause 5.10.2		
CF_05		
PICS Selection PICS_DisCsoReg		
nitial conditions with { the SUT containing an initial state }		
Permutations TRIVIOUS REPORT OF AN AND AND AND AND AND AND AND AND AND		
P Permutation id TP/NGSI-LD/CS/DISC/037_09_01		
lame 037_09_01 Observation Interval With observedAt		
Parent Release v1.3.1		
Check that one can query context source registrations. If present, the temp	oorai query is	
matched against the observationInterval or the managementInterval xpected behaviour Test events	Direction	
	SUT <- Client	
when { the SUT receives a Request from the client containing:	SOT <- CITETIL	
URL set to '/		
ngsi-ld/v1/csourceRegistrations'		
method set to 'GET'		
Retrieve Temporal Representation of Entity and		
Query Parameter: context set to 'https://		
forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/		
jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and		
Query Parameter: type set to 'Building' and		
Query Parameter: timeproperty set to 'observedAt' and		
Query Parameter: timerel set to 'before' and		
Query Parameter: timeAt set to '2021-08-01T22:00:00Z'		
then {	SUT -> Client	
the SUT sends a valid Response for the operations:	501 -> Client	
Query Context Source Registrations with Response Status Code set	to	
200 and		
Notification with Response body set to list of all matching Context		
Source Registrations resolved against the default JSON-		
LD context		
LD context }		
LD context } P Permutation id TP/NGSI-LD/CS/DISC/037_09_02		
LD context } P Permutation id TP/NGSI-LD/CS/DISC/037_09_02 Jame 037_09_02 Observation Interval Without timeproperty		
LD context } P Permutation id TP/NGSI-LD/CS/DISC/037_09_02 Jame 037_09_02 Observation Interval Without timeproperty Parent Release v1.3.1		
LD context } P Permutation id TP/NGSI-LD/CS/DISC/037_09_02 Jame 037_09_02 Observation Interval Without timeproperty Parent Release v1.3.1 Description Check that one can query context source registrations. If present, the temp	poral query is	
LD context } P Permutation id TP/NGSI-LD/CS/DISC/037_09_02 Jame 037_09_02 Observation Interval Without timeproperty Parent Release v1.3.1 Description Check that one can query context source registrations. If present, the temporation matched against the observationInterval or the managementInterval		
LD context } P Permutation id TP/NGSI-LD/CS/DISC/037_09_02 Jame 037_09_02 Observation Interval Without timeproperty Parent Release v1.3.1 Check that one can query context source registrations. If present, the temporate matched against the observationInterval or the managementInterval Expected behaviour Test events	Direction	
LD context } P Permutation id TP/NGSI-LD/CS/DISC/037_09_02 Jame 037_09_02 Observation Interval Without timeproperty Parent Release v1.3.1 Check that one can query context source registrations. If present, the temperature matched against the observationInterval or the managementInterval when {		
LD context } P Permutation id TP/NGSI-LD/CS/DISC/037_09_02 Jame 037_09_02 Observation Interval Without timeproperty V1.3.1 Check that one can query context source registrations. If present, the temperature matched against the observationInterval or the managementInterval Expected behaviour Test events when { the SUT receives a Request from the client containing:	Direction	
LD context } P Permutation id TP/NGSI-LD/CS/DISC/037_09_02 Jame 037_09_02 Observation Interval Without timeproperty V1.3.1 Check that one can query context source registrations. If present, the temperature matched against the observationInterval or the managementInterval Expected behaviour Test events when { the SUT receives a Request from the client containing:	Direction	
LD context } P Permutation id TP/NGSI-LD/CS/DISC/037_09_02 Jame 037_09_02 Observation Interval Without timeproperty V1.3.1 Check that one can query context source registrations. If present, the temporate matched against the observationInterval or the managementInterval Expected behaviour Test events when { the SUT receives a Request from the client containing:	Direction	
LD context } P Permutation id TP/NGSI-LD/CS/DISC/037_09_02 Jame 037_09_02 Observation Interval Without timeproperty Parent Release v1.3.1 Check that one can query context source registrations. If present, the temporate matched against the observationInterval or the managementInterval Expected behaviour Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceRegistrations' method set to 'GET'	Direction	
LD context } P Permutation id TP/NGSI-LD/CS/DISC/037_09_02 Jame 037_09_02 Observation Interval Without timeproperty V1.3.1 Check that one can query context source registrations. If present, the temporate matched against the observationInterval or the managementInterval Expected behaviour Test events when { the SUT receives a Request from the client containing:	Direction	
LD context } P Permutation id TP/NGSI-LD/CS/DISC/037_09_02 Jame 037_09_02 Observation Interval Without timeproperty V1.3.1 Check that one can query context source registrations. If present, the temporate description Expected behaviour Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and	Direction	
LD context } P Permutation id TP/NGSI-LD/CS/DISC/037_09_02 Jame 037_09_02 Observation Interval Without timeproperty V1.3.1 Check that one can query context source registrations. If present, the temporate matched against the observationInterval or the managementInterval Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-Id-test-suite-compound.jsonld' and	Direction	
LD context } TP Permutation id TP/NGSI-LD/CS/DISC/037_09_02 Jame 037_09_02 Observation Interval Without timeproperty Parent Release v1.3.1 Check that one can query context source registrations. If present, the temp matched against the observationInterval or the managementInterval Expected behaviour Test events When { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: type set to 'Building' and	Direction	
LD context } TP Permutation id TP/NGSI-LD/CS/DISC/037_09_02 Jame 037_09_02 Observation Interval Without timeproperty Parent Release v1.3.1 Check that one can query context source registrations. If present, the temperature and the observation of the management of the management of the management of the management of the sum o	Direction	
LD context } P Permutation id TP/NGSI-LD/CS/DISC/037_09_02 Jame 037_09_02 Observation Interval Without timeproperty Parent Release v1.3.1 Check that one can query context source registrations. If present, the temperature against the observationInterval or the managementInterval Expected behaviour Test events When { the SUT receives a Request from the client containing:	Direction	
LD context } P Permutation id TP/NGSI-LD/CS/DISC/037_09_02 Jame 037_09_02 Observation Interval Without timeproperty Parent Release v1.3.1 Check that one can query context source registrations. If present, the temp matched against the observationInterval or the managementInterval Expected behaviour Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: type set to 'Building' and Query Parameter: timeproperty set to '\$ {EMPTY}' and Query Parameter: timerel set to 'before' and	Direction	
LD context } P Permutation id TP/NGSI-LD/CS/DISC/037_09_02 Jame 037_09_02 Observation Interval Without timeproperty Parent Release v1.3.1 Check that one can query context source registrations. If present, the temperature against the observationInterval or the managementInterval Expected behaviour Test events When { the SUT receives a Request from the client containing:	Direction	
LD context } P Permutation id TP/NGSI-LD/CS/DISC/037_09_02 Jame 037_09_02 Observation Interval Without timeproperty V1.3.1 Check that one can query context source registrations. If present, the temp matched against the observationInterval or the managementInterval Test events When { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/ jsonId-contexts/ngsi-Id-test-suite-compound.jsonId' and Query Parameter: type set to 'Building' and Query Parameter: timeproperty set to '\$ {EMPTY}' and Query Parameter: timerel set to 'before' and Query Parameter: timeAt set to '2021-08-01T22:00:00Z' }	Direction SUT <- Client	
LD context } P Permutation id TP/NGSI-LD/CS/DISC/037_09_02 Jame 037_09_02 Observation Interval Without timeproperty V1.3.1 Check that one can query context source registrations. If present, the temp matched against the observationInterval or the managementInterval Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonId-contexts/ngsi-ld-test-suite-compound.jsonId' and Query Parameter: type set to 'Building' and Query Parameter: timeproperty set to '\$ {EMPTY}' and Query Parameter: timerel set to 'before' and Query Parameter: timeAt set to '2021-08-01T22:00:00Z' } then {	Direction	
LD context } P Permutation id TP/NGSI-LD/CS/DISC/037_09_02 Jame 037_09_02 Observation Interval Without timeproperty v1.3.1 Check that one can query context source registrations. If present, the temp matched against the observationInterval or the managementInterval When {	Direction SUT <- Client	
LD context } P Permutation id TP/NGSI-LD/CS/DISC/037_09_02 Jame 037_09_02 Observation Interval Without timeproperty V1.3.1 Check that one can query context source registrations. If present, the temp matched against the observationInterval or the managementInterval Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonId-contexts/ngsi-ld-test-suite-compound.jsonId' and Query Parameter: type set to 'Building' and Query Parameter: timeproperty set to '\$ {EMPTY}' and Query Parameter: timerel set to 'before' and Query Parameter: timeAt set to '2021-08-01T22:00:00Z' } then {	Direction SUT <- Client	

	Source Registrations resolved against the default JSON- LD context	
TD Dormutation id	TD/NGCL L D/GC/DIGG/027 00 02	
TP Permutation id Name	TP/NGSI-LD/CS/DISC/037_09_03	
Parent Release	037_09_03 Management Interval With createdAt	
Description	V1.3.1	ol augraio
Description	Check that one can query context source registrations. If present, the temporal query is matched against the observationInterval or the managementInterval	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: type set to 'Building' and Query Parameter: timeproperty set to 'createdAt' and Query Parameter: timerel set to 'before' and Query Parameter: timeAt set to '2021-08-01T22:00:00Z'	SUT <- Client
	then { the SUT sends a valid Response for the operations: Query Context Source Registrations with Response Status Code set to 200 and Notification with Response body set to list of all matching Context Source Registrations resolved against the default JSON- LD context	SUT -> Client
TP Permutation id	TP/NGSI-LD/CS/DISC/037_09_04	
Name	037_09_04 Management Interval With modifiedAt	
Parent Release	V1.3.1	
Description	Check that one can query context source registrations. If present, the tempore matched against the observationInterval or the managementInterval	al query is
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: type set to 'Building' and Query Parameter: timeproperty set to 'modifiedAt' and Query Parameter: timerel set to 'before' and Query Parameter: timeAt set to '2021-08-01T22:00:00Z' } then {	SUT <- Client
	then { the SUT sends a valid Response for the operations: Query Context Source Registrations with Response Status Code set to 200 and Notification with Response body set to list of all matching Context Source Registrations resolved against the default JSON-LD context }	SU1 -> Client

TP Id	TP/NGSI-LD/CS/DISC/037_10
	Check that one can query context source registrations. If present, the conditions specified by the context source query match the respective Context Source Properties
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.10.2
Config Id	CF_05
PICS Selection	PICS DisCsoRea

Initial conditions	with { the SUT being in the "initial state" and the SUT containing a list of Containing (CSRs) providing latest information about some entities }	ext Source
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/DISC/037_10_01	
Name	037_10_01 With list of Entity Ids	
Parent Release	v1.3.1	
Description	Check that one can query context source registrations. If present, the conditions specified by the context source query match the respective Context Source Properties	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: id set to '\$ {first_context_source_registration_id}, \$ {third_context_source_registration_id}' } then { the SUT sends a valid Response for the operations: Query Context Source Registrations with Response Status Code set to 200 and Notification with Response body set to list of all matching Context Source Registrations resolved against the default JSON-LD context	SUT -> Client
	}	
TP Permutation id	TP/NGSI-LD/CS/DISC/037_10_02	
Name	037_10_02 With NGSI-LD Query	
Parent Release	v1.3.1	
Description	Check that one can query context source registrations. If present, the condition the context source query match the respective Context Source Properties	ns specified by
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: q set to 'csourceProperty1=="aValue"'	SUT <- Client
	then { the SUT sends a valid Response for the operations: Query Context Source Registrations with Response Status Code set to 200 and Notification with Response body set to list of all matching Context Source Registrations resolved against the default JSON-LD context }	SUT -> Client

TP Id	TP/NGSI-LD/CS/DISC/037_01
Test objective	Check that one can query context source registrations if at least one of list of Entity Types or
	list of Attribute names is present
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.10.2
Config Id	CF_05
PICS Selection	PICS_DisCsoReg
Initial conditions	with { the SUT being in the "initial state" and the SUT containing a list of Context Source
	Registrations (CSRs) providing latest information about some entities }
Permutations Permutations	
TP Permutation id	TP/NGSI-LD/CS/DISC/037 01 01

Name	037_01_01 With list of entity types	
Parent Release	v1.3.1	
Description	Check that one can query context source registrations if at least one of list of	Entity Types or
	list of Attribute names is present	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and	SUT <- Client
	Query Parameter: type set to 'Building' } then { the SUT sends a valid Response for the operations: Query Context Source Registrations with Response Status Code set to 200 and Notification with Response body set to list of all matching Context Source Registrations resolved against the default JSON-LD context }	SUT -> Client
TP Permutation id	TP/NGSI-LD/CS/DISC/037_01_02	1
Name	037_01_02 With list of attribute names	
Parent Release	v1.3.1	
Description	Check that one can query context source registrations if at least one of list of list of Attribute names is present	Entity Types or
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: attrs set to 'name' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Query Context Source Registrations with Response Status Code set to 200 and Notification with Response body set to list of all matching Context Source Registrations resolved against the default JSON- LD context }	SUT -> Client

TP Id	TP/NGSI-LD/CS/DISC/037_03
Test objective	Check that one cannot query context source registrations, if the list of Entity identifiers includes a URI which it is not valid, or the query, geo-query or temporal query are not syntactically valid
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.10.2
Config Id	CF_05
PICS Selection	PICS_DisCsoReg
Initial conditions	with { the SUT containing an initial state }
	Permutations
TP Permutation id	TP/NGSI-LD/CS/DISC/037_03_01
Name	037_03_01 Invalid URI
Parent Release	v1.3.1
Description	Check that one cannot query context source registrations, if the list of Entity identifiers includes a URI which it is not valid, or the query, geo-query or temporal query are not syntactically valid

Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ ngsi-ld/v1/csourceRegistrations'	
	method set to 'GET'	
	Retrieve Temporal Representation of Entity and	
	Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: id set to 'invalidUri'	
	Query Parameter, id set to invalidon	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	OO 1 2 GHOIR
	Query Context Source Registrations with Response Status Code set to	
	400 and	
	Query Context Source Registrations with Response Body containing	
	the type 'https:// uri.etsi.org/ngsi-ld/errors/	
	BadRequestData' and	
	Query Context Source Registrations with Response body containing	
	'title' element	
TD D (. (
TP Permutation id Name	TP/NGSI-LD/CS/DISC/037_03_02 037_03_02 Invalid Query	
Parent Release	v1.3.1	
Description	Check that one cannot query context source registrations, if the list of Entity ic	dentifiers
F	includes a URI which it is not valid, or the query, geo-query or temporal query	
	syntactically valid	
Expected behaviour	Test events	Direction
	when { the SLIT receives a Request from the client containing:	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/	
	ngsi-ld/v1/csourceRegistrations'	
	method set to 'GET'	
	Retrieve Temporal Representation of Entity and	
	Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and	
	Query Parameter: q set to '(invalidQuery'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Query Context Source Registrations with Response Status Code set to	
	400 and Query Context Source Registrations with Response Body containing	
	the type 'https://	
	uri.etsi.org/ngsi-ld/errors/	
	BadRequestData' and	
	Query Context Source Registrations with Response body containing	
	'title' element	
TP Permutation id	TP/NGSI-LD/CS/DISC/037_03_03	
Name	037_03_03 Invalid GeoQuery	
Parent Release	v1.3.1	
Description	Check that one cannot query context source registrations, if the list of Entity ic	
	includes a URI which it is not valid, or the query, geo-query or temporal query	are not
Expected behaviour	syntactically valid Test events	Direction
-Apostou boliavioui	when {	SUT <- Client
	the SUT receives a Request from the client containing:	33.3.3.3
	URL set to '/	
	ngsi-ld/v1/csourceRegistrations'	
	method set to 'GET'	
	Retrieve Temporal Representation of Entity and	
	Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	jsonld-contexts/ngsi-ld-test-suite-compound.jsonld and	
		•

	Query Parameter: georel set to 'within'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Query Context Source Registrations with Response Status Code set to	
	400 and	
	Query Context Source Registrations with Response Body containing	
	the type 'https://	
	uri.etsi.org/ngsi-ld/errors/	
	BadRequestData' and	
	Query Context Source Registrations with Response body containing	
	'title' element	
	}	
TP Permutation id	TP/NGSI-LD/CS/DISC/037_03_04	
Name	037_03_04 Invalid Temporal Query	
Parent Release	v1.3.1	
Description	Check that one cannot query context source registrations, if the list of Entity ic	
	includes a URI which it is not valid, or the query, geo-query or temporal query	are not
	syntactically valid	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
		OO I V OHOIR
	the SUT receives a Request from the client containing:	
	the SUT receives a Request from the client containing: URL set to '/	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations'	oo i v olloini
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'GET'	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https://	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: timerel set to 'before' } then {	SUT -> Client
	the SUT receives a Request from the client containing:	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: timerel set to 'before' } then {	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/ jsonId-contexts/ngsi-Id-test-suite-compound.jsonId' and Query Parameter: timerel set to 'before' } then { the SUT sends a valid Response for the operations: Query Context Source Registrations with Response Status Code set to 400 and	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/ jsonId-contexts/ngsi-Id-test-suite-compound.jsonId' and Query Parameter: timerel set to 'before' } then { the SUT sends a valid Response for the operations: Query Context Source Registrations with Response Status Code set to 400 and Query Context Source Registrations with Response Body containing	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: timerel set to 'before' } then { the SUT sends a valid Response for the operations: Query Context Source Registrations with Response Status Code set to 400 and Query Context Source Registrations with Response Body containing the type 'https://	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: timerel set to 'before' } then { the SUT sends a valid Response for the operations: Query Context Source Registrations with Response Status Code set to 400 and Query Context Source Registrations with Response Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: timerel set to 'before' } then { the SUT sends a valid Response for the operations: Query Context Source Registrations with Response Status Code set to 400 and Query Context Source Registrations with Response Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/ BadRequestData' and	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: timerel set to 'before' } then { the SUT sends a valid Response for the operations: Query Context Source Registrations with Response Status Code set to 400 and Query Context Source Registrations with Response Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: timerel set to 'before' } then { the SUT sends a valid Response for the operations: Query Context Source Registrations with Response Status Code set to 400 and Query Context Source Registrations with Response Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/ BadRequestData' and	

TP Id	TP/NGSI-LD/CS/DISC/037_07	
Test objective	Check that one can query context source registrations. If present, the geoquery	y is matched
	against the GeoProperty programmatic parameter identified in the geoquery	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.10.2	
Config Id	CF_05	
PICS Selection	PICS_DisCsoReg	
Initial conditions	with { the SUT being in the "initial state" and the SUT contains a Context Source	ce
	Registration with id equal to \${context_source_registration_id} and payload set	t to
	\${context_source_registration_payload_file_path} }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/DISC/037_07_01	
Name	037_07_01 Near Point	
Parent Release	v1.3.1	
Description	Check that one can query context source registrations. If present, the geoquery	y is matched
	against the GeoProperty programmatic parameter identified in the geoquery	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/csourceRegistrations'	
	method set to 'GET'	
	Retrieve Temporal Representation of Entity and	
	Query Parameter: context set to 'https://	

	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and	
	Query Parameter: type set to 'Building' and	
	Query Parameter: georel set to 'near;maxDistance==2000' and	
	Query Parameter: geometry set to 'Point' and	
	Query Parameter: coordinates set to '[-8.503,41.202]	
	and Page Page 14 and 15 and 16	
	Query Parameter: geoproperty set to '\$	
	{EMPTY}' \	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	00.70
	Query Context Source Registrations with Response Status Code set to	
	200 and	
	Notification with Response body set to list of all matching Context	
	Source Registrations resolved against the default JSON-	
	LD context	
	}	
TP Permutation id	TP/NGSI-LD/CS/DISC/037_07_02	
Name	037_07_02 Within Polygon	
Parent Release	v1.3.1	
Description	Check that one can query context source registrations. If present, the geoquer	ry is matched
Formanda di bindi ancieroni	against the GeoProperty programmatic parameter identified in the geoquery	Discotion
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/	SOT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations'	SOT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'GET'	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https://	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: type set to 'Building' and	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: type set to 'Building' and Query Parameter: georel set to 'within' and	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: type set to 'Building' and Query Parameter: georel set to 'within' and Query Parameter: geometry set to 'Polygon' and	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: type set to 'Building' and Query Parameter: georel set to 'within' and Query Parameter: geometry set to 'Polygon' and Query Parameter: coordinates set to '[[-13.503,47.202],	SUT <- Client
	the SUT receives a Request from the client containing:	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: type set to 'Building' and Query Parameter: georel set to 'within' and Query Parameter: geometry set to 'Polygon' and Query Parameter: coordinates set to '[[-13.503,47.202], [6.541, 52.961],[20.37,44.653],[9.46,32.57], [-15.23,21.37]]' and	SUT <- Client
	the SUT receives a Request from the client containing:	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: type set to 'Building' and Query Parameter: georel set to 'within' and Query Parameter: geometry set to 'Polygon' and Query Parameter: coordinates set to '[[-13.503,47.202], [6.541, 52.961],[20.37,44.653],[9.46,32.57], [-15.23,21.37]]' and	SUT -> Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: type set to 'Building' and Query Parameter: georel set to 'within' and Query Parameter: geometry set to 'Polygon' and Query Parameter: coordinates set to '[[-13.503,47.202], [6.541, 52.961],[20.37,44.653],[9.46,32.57], [-15.23,21.37]]' and Query Parameter: geoproperty set to 'location' } then { the SUT sends a valid Response for the operations:	
	the SUT receives a Request from the client containing:	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: type set to 'Building' and Query Parameter: georel set to 'within' and Query Parameter: geometry set to 'Polygon' and Query Parameter: coordinates set to '[[-13.503,47.202], [6.541, 52.961],[20.37,44.653],[9.46,32.57], [-15.23,21.37]]' and Query Parameter: geoproperty set to 'location' } then { the SUT sends a valid Response for the operations:	
	the SUT receives a Request from the client containing:	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: type set to 'Building' and Query Parameter: georel set to 'within' and Query Parameter: geometry set to 'Polygon' and Query Parameter: coordinates set to '[[-13.503,47.202], [6.541, 52.961],[20.37,44.653],[9.46,32.57], [-15.23,21.37]]' and Query Parameter: geoproperty set to 'location' } then { the SUT sends a valid Response for the operations: Query Context Source Registrations with Response Status Code set to 200 and	
	the SUT receives a Request from the client containing:	

TP Id	TP/NGSI-LD/CS/DISC/037_05
Test objective	Check that one can query context source registrations matching EntityInfo of
	RegistrationInfo
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.10.2
Config Id	CF_05
PICS Selection	PICS_DisCsoReg
Initial conditions	with { the SUT containing an initial state }
	Permutations
TP Permutation id	TP/NGSI-LD/CS/DISC/037_05_01
Name	037_05_01 Registration With EntityInfo Matching The Query
Parent Release	v1.3.1
Description	Check that one can query context source registrations matching EntityInfo of
	RegistrationInfo

Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/csourceRegistrations'	
	method set to 'GET'	
	Retrieve Temporal Representation of Entity and	
	Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	ionge:etsi.org/rep/cin/ngsi-id-test-suite-compound.jsonId' and	
	Query Parameter: type set to 'OffStreetParking' and	
	Query Parameter: idPattern set to '.*downtown\$'	
	Query Parameter. IdPattern Set to . downtowns	
	About (CLIT . Oliana
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Query Context Source Registrations with Response Status Code set to	
	200 and	
	Notification with Response body set to list of all matching Context	
	Source Registrations resolved against the default JSON-	
	LD context	
	}	
TP Permutation id	TP/NGSI-LD/CS/DISC/037_05_02	
Name	037_05_02 Registration Without EntityInfo	
Parent Release	v1.3.1	
Description	Check that one can query context source registrations matching EntityInfo of	
-	RegistrationInfo	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/csourceRegistrations'	
	method set to 'GET'	
	Retrieve Temporal Representation of Entity and	
	Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	jsonId-contexts/ngsi-Id-test-suite-compound.jsonId' and	
	Query Parameter: type set to 'OffStreetParking' and	
	Query Parameter: idPattern set to '.*downtown\$'	
	l	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	301 -> Ciletit
	Query Context Source Registrations with Response Status Code set to	
	200 and	
	Notification with Response body set to list of all matching Context	
	Source Registrations resolved against the default JSON-	
	LD context	
	1)	1

TP ld	TP/NGSI-LD/CS/DISC/037 02		
Test objective	Check that one cannot query context source registrations, if neither Entity types nor Attribute names are provided, an error of type 400 shall be raised.		
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.10.2		
Config Id	CF_05		
PICS Selection	PICS_DisCsoReg		
Initial conditions	with { the SUT containing an initial state }		
	Permutations		
TP Permutation id	TP/NGSI-LD/CS/DISC/037_02_01	TP/NGSI-LD/CS/DISC/037_02_01	
Name	037_02_01 Query Context Source Registrations Without Entity Types and Attr	037 02 01 Query Context Source Registrations Without Entity Types and Attribute Names	
Parent Release	v1.3.1		
Description	Check that one cannot query context source registrations, if neither Entity type names are provided, an error of type 400 shall be raised.	es nor Attribute	
Expected behaviour	Test events	Direction	
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/csourceRegistrations'	SUT <- Client	

method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' }	
then { the SUT condo a valid Response for the operations:	SUT -> Client
the SUT sends a valid Response for the operations:	
Query Context Source Registrations with Response Status Code set to 400 and	
Query Context Source Registrations with Response Body containing	
the type 'https://	
uri.etsi.org/ngsi-ld/errors/	
BadRequestData' and	
Query Context Source Registrations with Response body containing	
'title' element	
}	

TP ld	TP/NGSI-LD/CS/DISC/037_04	
Test objective	Check that one can query context source registrations. If a JSON-LD context is	
	provided, then all the query terms shall be resolved against the default JSON-l	_D @context
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.10.2	
Config Id	CF_05	
PICS Selection	PICS_DisCsoReg	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CS/DISC/037_04_01	
Name	037_04_01 Query Context Source Registrations Without Context	
Parent Release	v1.3.1	
Description	Check that one can query context source registrations. If a JSON-LD context is	
	provided, then all the query terms shall be resolved against the default JSON-l	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: id set to '\${context_source_registration_id}' and Query Parameter: type set to 'https://ngsi-ld-test-suite/context#Building' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Query Context Source Registrations with Response Status Code set to 200 and Notification with Response body set to list of all matching Context Source Registrations resolved against the default JSON- LD context }	SUT -> Client

TP ld	TP/NGSI-LD/CS/DISC/037_06
Test objective	Check that one can query context source registrations matching property and relationship
	names of RegistrationInfo
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.10.2
Config Id	CF_05
PICS Selection	PICS_DisCsoReg
Initial conditions	with { the SUT being in the "initial state" and the SUT contains a Context Source
	Registration with id equal to \${context_source_registration_id} and payload set to
	\${context_source_registration_payload_file_path} }
	Permutations
TP Permutation id	TP/NGSI-LD/CS/DISC/037_06_01
Name	037_06_01 Query With Matching Properties And Relationships
Parent Release	v1.3.1
Description	Check that one can query context source registrations matching property and relationship
	names of RegistrationInfo

Expected behaviour	Test events	Direction
-	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/csourceRegistrations'	
	method set to 'GET'	
	Retrieve Temporal Representation of Entity and	
	Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and	
	Query Parameter: type set to 'Building' and	
	Query Parameter: attrs set to 'name,locatedAt'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Query Context Source Registrations with Response Status Code set to	
	200 and	
	Notification with Response body set to list of all matching Context	
	Source Registrations resolved against the default JSON-	
	LD context	
	}	
TP Permutation id	TP/NGSI-LD/CS/DISC/037_06_02	
Name	037_06_02 Query Without Properties And Relationships	
Parent Release	v1.3.1	
	Tall the state of	1 21 1 1
Description	Check that one can query context source registrations matching property and	relationship
	names of RegistrationInfo	·
Description Expected behaviour	names of RegistrationInfo Test events	Direction
	names of RegistrationInfo Test events when {	<u>.</u>
	names of RegistrationInfo Test events when { the SUT receives a Request from the client containing:	Direction
	names of RegistrationInfo Test events when { the SUT receives a Request from the client containing: URL set to '/	Direction
	names of RegistrationInfo Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceRegistrations'	Direction
	names of RegistrationInfo Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceRegistrations' method set to 'GET'	Direction
	names of RegistrationInfo Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and	Direction
	names of RegistrationInfo Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https://	Direction
	names of RegistrationInfo Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/	Direction
	names of RegistrationInfo Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/ jsonId-contexts/ngsi-Id-test-suite-compound.jsonId' and	Direction
	names of RegistrationInfo Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/ jsonId-contexts/ngsi-Id-test-suite-compound.jsonId' and Query Parameter: type set to 'Building' and	Direction
	names of RegistrationInfo Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/ jsonId-contexts/ngsi-Id-test-suite-compound.jsonId' and Query Parameter: type set to 'Building' and Query Parameter: attrs set to '\$	Direction
	names of RegistrationInfo Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/ jsonId-contexts/ngsi-Id-test-suite-compound.jsonId' and Query Parameter: type set to 'Building' and	Direction
	names of RegistrationInfo Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/ jsonId-contexts/ngsi-Id-test-suite-compound.jsonId' and Query Parameter: type set to 'Building' and Query Parameter: attrs set to '\$ {EMPTY}' }	Direction SUT <- Client
•	names of RegistrationInfo Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/ jsonId-contexts/ngsi-Id-test-suite-compound.jsonId' and Query Parameter: type set to 'Building' and Query Parameter: attrs set to '\$ {EMPTY}' } then {	Direction
	names of RegistrationInfo Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/ jsonId-contexts/ngsi-Id-test-suite-compound.jsonId' and Query Parameter: type set to 'Building' and Query Parameter: attrs set to '\$ {EMPTY}' } then { the SUT sends a valid Response for the operations:	Direction SUT <- Client
	names of RegistrationInfo Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/ jsonId-contexts/ngsi-Id-test-suite-compound.jsonId' and Query Parameter: type set to 'Building' and Query Parameter: attrs set to '\$ {EMPTY}' } then { the SUT sends a valid Response for the operations: Query Context Source Registrations with Response Status Code set to	Direction SUT <- Client
	names of RegistrationInfo Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/ jsonId-contexts/ngsi-Id-test-suite-compound.jsonId' and Query Parameter: type set to 'Building' and Query Parameter: attrs set to '\$ {EMPTY}' } then { the SUT sends a valid Response for the operations: Query Context Source Registrations with Response Status Code set to 200 and	Direction SUT <- Client
	names of RegistrationInfo Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/ jsonId-contexts/ngsi-Id-test-suite-compound.jsonId' and Query Parameter: type set to 'Building' and Query Parameter: attrs set to '\$ {EMPTY}' } then { the SUT sends a valid Response for the operations: Query Context Source Registrations with Response Status Code set to 200 and Notification with Response body set to list of all matching Context	Direction SUT <- Client
	names of RegistrationInfo Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/ jsonId-contexts/ngsi-Id-test-suite-compound.jsonId' and Query Parameter: type set to 'Building' and Query Parameter: attrs set to '\$ {EMPTY}' } then { the SUT sends a valid Response for the operations: Query Context Source Registrations with Response Status Code set to 200 and Notification with Response body set to list of all matching Context Source Registrations resolved against the default JSON-	Direction SUT <- Client
•	names of RegistrationInfo Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/csourceRegistrations' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/ jsonId-contexts/ngsi-Id-test-suite-compound.jsonId' and Query Parameter: type set to 'Building' and Query Parameter: attrs set to '\$ {EMPTY}' } then { the SUT sends a valid Response for the operations: Query Context Source Registrations with Response Status Code set to 200 and Notification with Response body set to list of all matching Context	Direction SUT <- Client

4.4 ContextInformation

4.4.1 Consumption

4.4.1.1 TemporalEntity

4.4.1.1.1 RetrieveTemporalEvolutionOfEntity

TP Id	TP/NGSI-LD/CI/Cons/TE/020_08
Test objective	Check that one cannot retrieve the temporal evolution of non-existing entity attributes
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.3
Config Id	CF_01
PICS Selection	PICS_TemEvoEntRet
Initial conditions	with { the SUT containing an initial state }

Permutations		
TP Permutation id	TP/NGSI-LD/CI/Cons/TE/020_08_01	
Name	020_08_01 Retrieve the temporal evolution of non-existing entity attributes	
Parent Release	v1.3.1	
Description	Check that one cannot retrieve the temporal evolution of non-existing entity attributes	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attrs set to '\$ {temporal_attributes_to_be_retrieved}' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld'	SUT <- Client
	then { the SUT sends a valid Response for the operations: Retrieve Temporal Representation Of Entity with Response Status Code set to 404 and Retrieve Temporal Representation Of Entity with Response Body containing the type 'https:// uri.etsi.org/ngsi-Id/errors/ ResourceNotFound' and Retrieve Temporal Representation Of Entity with Response body containing 'title' element }	SUT -> Client

TP Id	TP/NGSI-LD/CI/Cons/TE/020_11	
Test objective	Check that one can retrieve the temporal evolution of an entity with the aggreg	ated temporal
	representation	•
Reference	ETSI GS CIM 009 V1.4.1 [1], clauses 4.5.19, 5.7.3	
Config Id	CF_01	
PICS Selection	PICS_AGGRTEMPORAL and PICS_TemEvoEntRet	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/TE/020_11_01	
Name	020_11_01 One aggregate method aggregated by one hour duration	
Parent Release	v1.4.1	
Description	Check that one can retrieve the temporal evolution of an entity with the aggreg	ated temporal
	representation	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}'	
	method set to 'GET'	
	Retrieve Temporal Representation of Entity and	
	Query Parameter: id set to '\$	
	{temporal_entity_representation_id}' and	
	Query Parameter: attrs set to '\$	
	(EMPTY)' and	
	Query Parameter: options set to '\$ {options}' and	
	Query Parameter: aggrMethods set to 'avg' and	
	Query Parameter: aggrPeriodDuration set to 'PT1H' and	
	Query Parameter: context set to 'https://	
	Itorne etsi org/ren/cim/ngsi-ig-test-suite/-/raw/gevelon/resources/	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld'	

<u></u>		
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Retrieve Temporal Representation Of Entity with Response Status	
	Code set to	
	200 and	
	Retrieve Temporal Representation Of Entity with Response Body containing EntityTemporal element containing attribute instances in the time	
	range specified by the NGSI-	
	LD temporal query:	
	* the payload is defined in the file set to 'vehicle-	
	temporal-representation-020-11-01-expectation.json'	
	* the id was changed to '\$	
	{temporal_entity_representation_id}'	
	* response body to be checked set to '\$	
	{response.json()}'	
	}	
TP Permutation id	TP/NGSI-LD/CI/Cons/TE/020_11_02	
Name	020_11_02 One aggregate method aggregated by one hour duration asking f	or one attribute
Parent Release	v1.4.1	
Description	Check that one can retrieve the temporal evolution of an entity with the aggre representation	gated temporal
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}'	
	method set to 'GET'	
	Retrieve Temporal Representation of Entity and	
	Query Parameter: id set to '\$	
	{temporal_entity_representation_id}' and	
	Query Parameter: attrs set to 'fuelLevel' and	
	Query Parameter: options set to '\$	
	(options)' and	
	Query Parameter: aggrMethods set to 'avg' and Query Parameter: aggrPeriodDuration set to 'PT1H' and	
	Query Parameter: aggire nouburation set to Firm and Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	isonId-contexts/ngsi-Id-test-suite-compound.jsonId	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Retrieve Temporal Representation Of Entity with Response Status	
	Code set to	
	200 and	
	Retrieve Temporal Representation Of Entity with Response Body	
	containing EntityTemporal element containing attribute instances in the time	
	range specified by the NGSI-	
	LD temporal query: * the payload is defined in the file set to 'vehicle-	1
	temporal-representation-020-11-02-expectation.json	1
	* the id was changed to '\$	1
	the id was changed to \$ [temporal_entity_representation_id]'	1
	* response body to be checked set to '\$	
	{response.json()}'	
	}	
TP Permutation id	TP/NGSI-LD/CI/Cons/TE/020_11_03	
Name	020_11_03 Multiple aggregate methods aggregated by one hour duration	
Parent Release Description	v1.4.1 Check that one can retrieve the temporal evolution of an entity with the aggre	nated temporal
Description	representation	gated temporar
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	1
	ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}'	
	method set to 'GET'	1
	Retrieve Temporal Representation of Entity and	
	Query Parameter: id set to '\$	

	{temporal_entity_representation_id}' and	
	Query Parameter: attrs set to '\$	
	{EMPTY}' and	
	Query Parameter: options set to '\$	
	{options}' and	
	Query Parameter: aggrMethods set to 'avg,max' and	
	Query Parameter: aggrPeriodDuration set to 'PT1H' and	
	Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	jsonld-contexts/ngsi-ld-test-suite-compound.jsonld'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Retrieve Temporal Representation Of Entity with Response Status	
	Code set to	
	200 and	
	Retrieve Temporal Representation Of Entity with Response Body	
	containing EntityTemporal element containing attribute instances in the time	
	range specified by the NGSI-	
	LD temporal query:	
	* the payload is defined in the file set to 'vehicle-	
	temporal-representation-020-11-03-expectation.json'	
	* the id was changed to '\$	
	{temporal_entity_representation_id}'	
	* response body to be checked set to '\$	
	{response.json()}'	
	tiesponse.json();	
TP Permutation id	TP/NGSI-LD/CI/Cons/TE/020_11_04	
Name	020_11_04 Multiple aggregate methods aggregated by one day duration	
Parent Release	v1.4.1	
Description	Check that one can retrieve the temporal evolution of an entity with the aggreg	rated temporal
Description	representation	gated temporar
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	IMITELL 1	
	the SLIT receives a Request from the client containing:	
	the SUT receives a Request from the client containing:	out a choin
	URL set to '/	
	URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}'	
	URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'GET'	Ser v Gilorik
	URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'GET' Retrieve Temporal Representation of Entity and	
	URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: id set to '\$	
	URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: id set to '\$ {temporal_entity_representation_id}' and	
	URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attrs set to '\$	
	URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attrs set to '\$ {EMPTY}' and	
	URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attrs set to '\$ {EMPTY}' and Query Parameter: options set to '\$	
	URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attrs set to '\$ {EMPTY}' and Query Parameter: options set to '\$ {options}' and	
	URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attrs set to '\$ {EMPTY}' and Query Parameter: options set to '\$ {options}' and Query Parameter: aggrMethods set to 'min,max' and	
	URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attrs set to '\$ {EMPTY}' and Query Parameter: options set to '\$ {options}' and Query Parameter: aggrMethods set to 'min,max' and Query Parameter: aggrPeriodDuration set to 'P1D' and	
	URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attrs set to '\$ {EMPTY}' and Query Parameter: options set to '\$ {options}' and Query Parameter: aggrMethods set to 'min,max' and Query Parameter: aggrPeriodDuration set to 'P1D' and Query Parameter: context set to 'https://	
	URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attrs set to '\$ {EMPTY}' and Query Parameter: options set to '\$ {options}' and Query Parameter: aggrMethods set to 'min,max' and Query Parameter: aggrPeriodDuration set to 'P1D' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attrs set to '\$ {EMPTY}' and Query Parameter: options set to '\$ {options}' and Query Parameter: aggrMethods set to 'min,max' and Query Parameter: aggrPeriodDuration set to 'P1D' and Query Parameter: context set to 'https://	
	URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attrs set to '\$ {EMPTY}' and Query Parameter: options set to '\$ {options}' and Query Parameter: aggrMethods set to 'min,max' and Query Parameter: aggrPeriodDuration set to 'P1D' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' }	
	URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attrs set to '\$ {EMPTY}' and Query Parameter: options set to '\$ {options}' and Query Parameter: aggrMethods set to 'min,max' and Query Parameter: aggrPeriodDuration set to 'P1D' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' } then {	SUT -> Client
	URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attrs set to '\$ {EMPTY}' and Query Parameter: options set to '\$ {options}' and Query Parameter: aggrMethods set to 'min,max' and Query Parameter: aggrPeriodDuration set to 'P1D' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' } then { the SUT sends a valid Response for the operations:	
	URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attrs set to '\$ {EMPTY}' and Query Parameter: options set to '\$ {options}' and Query Parameter: aggrMethods set to 'min,max' and Query Parameter: aggrPeriodDuration set to 'P1D' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' } then { the SUT sends a valid Response for the operations: Retrieve Temporal Representation Of Entity with Response Status	
	URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attrs set to '\$ {EMPTY}' and Query Parameter: options set to '\$ {options}' and Query Parameter: aggrMethods set to 'min,max' and Query Parameter: aggrPeriodDuration set to 'P1D' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' } then { the SUT sends a valid Response for the operations: Retrieve Temporal Representation Of Entity with Response Status Code set to	
	URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attrs set to '\$ {EMPTY}' and Query Parameter: options set to '\$ {options}' and Query Parameter: aggrMethods set to 'min,max' and Query Parameter: aggrPeriodDuration set to 'P1D' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' } then { the SUT sends a valid Response for the operations: Retrieve Temporal Representation Of Entity with Response Status Code set to 200 and	
	URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attrs set to '\$ {EMPTY}' and Query Parameter: options set to '\$ {options}' and Query Parameter: aggrMethods set to 'min,max' and Query Parameter: aggrPeriodDuration set to 'P1D' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' } then { the SUT sends a valid Response for the operations: Retrieve Temporal Representation Of Entity with Response Status Code set to and Retrieve Temporal Representation Of Entity with Response Body	
	URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attrs set to '\$ {EMPTY}' and Query Parameter: options set to '\$ {options}' and Query Parameter: aggrMethods set to 'min,max' and Query Parameter: aggrPeriodDuration set to 'P1D' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' } then { the SUT sends a valid Response for the operations: Retrieve Temporal Representation Of Entity with Response Status Code set to and Retrieve Temporal Representation Of Entity with Response Body containing EntityTemporal element containing attribute instances in the time	
	URL set to '/ ngsi-Id/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attrs set to '\$ {EMPTY}' and Query Parameter: options set to '\$ {options}' and Query Parameter: aggrMethods set to 'min,max' and Query Parameter: aggrPeriodDuration set to 'P1D' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/ jsonId-contexts/ngsi-Id-test-suite-compound.jsonId' } then { the SUT sends a valid Response for the operations: Retrieve Temporal Representation Of Entity with Response Status Code set to 200 and Retrieve Temporal Representation Of Entity with Response Body containing EntityTemporal element containing attribute instances in the time range specified by the NGSI-	
	URL set to '/ ngsi-Id/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attrs set to '\$ {EMPTY}' and Query Parameter: options set to '\$ {options}' and Query Parameter: aggrMethods set to 'min,max' and Query Parameter: aggrPeriodDuration set to 'P1D' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/ jsonId-contexts/ngsi-Id-test-suite-compound.jsonId' } then { the SUT sends a valid Response for the operations: Retrieve Temporal Representation Of Entity with Response Status Code set to 200 and Retrieve Temporal Representation Of Entity with Response Body containing EntityTemporal element containing attribute instances in the time range specified by the NGSI- LD temporal query:	
	URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attrs set to '\$ {EMPTY}' and Query Parameter: options set to '\$ {options}' and Query Parameter: aggrMethods set to 'min,max' and Query Parameter: aggrPeriodDuration set to 'P1D' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' } then { the SUT sends a valid Response for the operations: Retrieve Temporal Representation Of Entity with Response Status Code set to 200 and Retrieve Temporal Representation Of Entity with Response Body containing EntityTemporal element containing attribute instances in the time range specified by the NGSI- LD temporal query: * the payload is defined in the file set to 'vehicle-	
	URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attrs set to '\$ {EMPTY}' and Query Parameter: options set to '\$ {options}' and Query Parameter: aggrMethods set to 'min,max' and Query Parameter: aggrPeriodDuration set to 'P1D' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' } then { the SUT sends a valid Response for the operations: Retrieve Temporal Representation Of Entity with Response Status Code set to 200 and Retrieve Temporal Representation Of Entity with Response Body containing EntityTemporal element containing attribute instances in the time range specified by the NGSI- LD temporal query: * the payload is defined in the file set to 'vehicle- temporal-representation-020-11-04-expectation.json'	
	URL set to '/ ngsi-Id/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attrs set to '\$ {EMPTY}' and Query Parameter: options set to '\$ {options}' and Query Parameter: aggrMethods set to 'min,max' and Query Parameter: aggrPeriodDuration set to 'P1D' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-Id-test-suite-compound.jsonld' } then { the SUT sends a valid Response for the operations: Retrieve Temporal Representation Of Entity with Response Status Code set to 200 and Retrieve Temporal Representation Of Entity with Response Body containing EntityTemporal element containing attribute instances in the time range specified by the NGSI- LD temporal query: * the payload is defined in the file set to 'vehicle- temporal-representation-020-11-04-expectation.json' * the id was changed to '\$	
	URL set to '/ ngsi-Id/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attrs set to '\$ {EMPTY}' and Query Parameter: options set to '\$ {options}' and Query Parameter: aggrMethods set to 'min,max' and Query Parameter: aggrPeriodDuration set to 'P1D' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-Id-test-suite-compound.jsonld' } then { the SUT sends a valid Response for the operations: Retrieve Temporal Representation Of Entity with Response Status Code set to 200 and Retrieve Temporal Representation Of Entity with Response Body containing EntityTemporal element containing attribute instances in the time range specified by the NGSI- LD temporal query: * the payload is defined in the file set to 'vehicle- temporal-representation-020-11-04-expectation.json' * the id was changed to '\$ {temporal_entity_representation_id}'	
	URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attrs set to '\$ {EMPTY}' and Query Parameter: options set to '\$ {options}' and Query Parameter: aggrMethods set to 'min,max' and Query Parameter: aggrPeriodDuration set to 'P1D' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' } then { the SUT sends a valid Response for the operations: Retrieve Temporal Representation Of Entity with Response Status Code set to 200 and Retrieve Temporal Representation Of Entity with Response Body containing EntityTemporal element containing attribute instances in the time range specified by the NGSI- LD temporal query: * the payload is defined in the file set to 'vehicle- temporal-representation-020-11-04-expectation.json' * the id was changed to '\$ {temporal_entity_representation_id}' * response body to be checked set to '\$	
	URL set to '/ ngsi-Id/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attrs set to '\$ {EMPTY}' and Query Parameter: options set to '\$ {options}' and Query Parameter: aggrMethods set to 'min,max' and Query Parameter: aggrPeriodDuration set to 'P1D' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-Id-test-suite-compound.jsonld' } then { the SUT sends a valid Response for the operations: Retrieve Temporal Representation Of Entity with Response Status Code set to 200 and Retrieve Temporal Representation Of Entity with Response Body containing EntityTemporal element containing attribute instances in the time range specified by the NGSI- LD temporal query: * the payload is defined in the file set to 'vehicle- temporal-representation-020-11-04-expectation.json' * the id was changed to '\$ {temporal_entity_representation_id}'	

TP/NGSI-LD/CI/Cons/TE/020 09	
	valid request
ETSI GS CIM 009 V1.5.1 [1], clause 5.7.3	
CF_01	
\${entity} with an id set to \${temporal_entity_representation_id} and an attributo \${atrld} }	
Permutations	
TP/NGSI-LD/CI/Cons/TE/020_09_01	
020_09_01 After	
v1.3.1	
Check that one cannot retrieve the temporal evolution of an entity with an incontent	valid request
Test events	Direction
when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: id set to '\$ {temporal_entity_representation_id}' and Query Parameter: timerel set to 'after' and Query Parameter: timeAt set to '\$ {EMPTY}' and Query Parameter: endTimeAt set to '\$ {EMPTY}' } then { the SUT sends a valid Response for the operations: Retrieve Temporal Representation Of Entity with Response Status Code set to 400 and Retrieve Temporal Representation Of Entity with Response Body containing the type 'https:// uri.etsi.org/ngsi-Id/errors/ BadRequestData' and Retrieve Temporal Representation Of Entity with Response body containing 'title' element }	SUT -> Client
TP/NGSLLD/CI/Cons/TE/020_00_02	
Check that one cannot retrieve the temporal evolution of an entity with an in	valid request
	Direction
when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: id set to '\$ {temporal_entity_representation_id}' and Query Parameter: timerel set to 'before' and Query Parameter: timeAt set to '\$ {EMPTY}' and Query Parameter: endTimeAt set to '\$ {EMPTY}' } then { the SUT sends a valid Response for the operations:	SUT <- Client
	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.3 CF_01 PICS_TemEvoEntRet with (the SUT being in the "initial state" and the SUT containing an initial Te \$(entity) with an id set to \$(temporal_entity_representation_id) and an attribit to \$(atrid) } Permutations TP/NGSI-LD/CI/Cons/TE/020_09_01 020_09_01 After V1.3.1 Check that one cannot retrieve the temporal evolution of an entity with an in content Test events when { the SUT receives a Request from the client containing: URL set to // ngsi-Id/V1/temporal/entities/{temporal_entity_representation_id}' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: id set to '\$ {temporal_entity_representation_id}' and Query Parameter: timeAt set to 'after' and Query Parameter: timeAt set to 'after' and Query Parameter: endTimeAt set to '\$ {EMPTY}' and Query Parameter: endTimeAt set to '\$ {EMPTY}' } then { the SUT sends a valid Response for the operations: Retrieve Temporal Representation Of Entity with Response Body containing the type 'https:// uri.etsi.org/ngsi-Id/errors/ BadRequestData' and Retrieve Temporal Representation Of Entity with Response body containing the type 'https:// uri.etsi.org/ngsi-Id/errors/ BadRequestData' and Retrieve Temporal Representation Of Entity with Response body containing title' element } Pr/NGSI-LD/CI/Cons/TE/020_09_02 020_09_02 Before V1.3.1 Check that one cannot retrieve the temporal evolution of an entity with an incontent Test events when { the SUT receives a Request from the client containing:

	Retrieve Temporal Representation Of Entity with Response Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/ BadRequestData' and Retrieve Temporal Representation Of Entity with Response body	
	containing 'title' element	
TP Permutation id	TP/NGSI-LD/CI/Cons/TE/020_09_03	
Name	020_09_03 Between	
Parent Release	v1.3.1	
Description	Check that one cannot retrieve the temporal evolution of an entity with an invacontent	lid request
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: id set to '\$ {temporal_entity_representation_id}' and Query Parameter: timerel set to 'between' and Query Parameter: timeAt set to '2020-08-01T12:00:00Z' and Query Parameter: endTimeAt set to '\$ {EMPTY}' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Retrieve Temporal Representation Of Entity with Response Status Code set to 400 and Retrieve Temporal Representation Of Entity with Response Body containing the type 'https:// uri.etsi.org/ngsi-Id/errors/ BadRequestData' and Retrieve Temporal Representation Of Entity with Response body containing 'title' element }	SUT -> Client

TP ld	TP/NGSI-LD/CI/Cons/TE/020_10	
Test objective	Check that one can retrieve the temporal evolution of an entity with the simple	ified temporal
	representation	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.3	
Config Id	CF_01	
PICS Selection	PICS_TemEvoEntRet	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/TE/020_10_01	
Name	020_10_01 Retrieve the temporal evolution of an entity with the simplified ter	mporal
	representation	
Parent Release	v1.3.1	
Description	Check that one can retrieve the temporal evolution of an entity with the simplified temporal	
	representation	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}'	
	method set to 'GET'	
	Retrieve Temporal Representation of Entity and	
	Query Parameter: id set to '\$	
	{temporal_entity_representation_id}' and	
	Query Parameter: options set to '\$	
	(options)' and	
	Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	

jsonld-contexts/ngsi-ld-test-suite-compound.jsonld'	
jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' } then { the SUT sends a valid Response for the operations: Retrieve Temporal Representation Of Entity with Response Status Code set to 200 and Retrieve Temporal Representation Of Entity with Response Body containing EntityTemporal element containing attribute instances in the time range specified by the NGSI- LD temporal query: * the payload is defined in the file set to 'vehicle- temporal-representation-020-10-expectation.jsonld' * the id was changed to '\$ {temporal_entity_representation_id}'	SUT -> Client
{temporal_entity_representation_id}' * response body to be checked set to '\$	
{response.json()}' }	

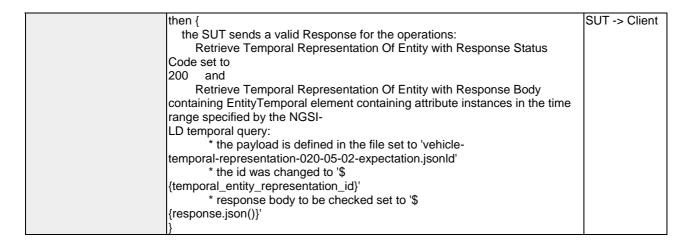
TP Id	TP/NGSI-LD/CI/Cons/TE/020_12	
Test objective	Check that one can retrieve the temporal evolution of an entity with a Language	neProperty
l est objective	property	ger roperty
Reference	ETSI GS CIM 009 V1.4.1 [1], clauses 4.5.18, 4.5.7, 4.5.9, 5.7.3	
Config Id	CF_01	
PICS Selection	PICS_LANG and PICS_P_TEMPORAL_REP and	
FICS Selection	PICS_P_SIMPTEMPORAL_REP_AND_R_SIMPTEMPORAL_REP and	
	PICS TemEvoEntRet	
Initial conditions	with { the SUT containing an initial state }	
initial conditions	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/TE/020_12_01	
Name	020 12 01 Patricys the normalized temporal representation of an antity with	
Name	020_12_01 Retrieve the normalized temporal representation of an entity with	а
Devent Deleges	LanguageProperty property	
Parent Release	v1.4.1	D
Description	Check that one can retrieve the temporal evolution of an entity with a Language	уенторепту
Everated habovis	property	Diversian
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}'	
	method set to 'GET'	
	Retrieve Temporal Representation of Entity and Query Parameter: id set to '\$	
	{temporal_entity_representation_id}' and	
	Query Parameter: options set to '\$	
	{EMPTY}' and	
	Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	isonId-contexts/ngsi-Id-test-suite-compound.jsonId	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	OO1 > OHOIR
	Retrieve Temporal Representation Of Entity with Response Status	
	Code set to	
	200 and	
	Retrieve Temporal Representation Of Entity with Response Body	
	containing EntityTemporal element containing attribute instances in the time	
	range specified by the NGSI-	
	LD temporal query:	
	* the payload is defined in the file set to 'vehicle-	
	language-property-normalized-temporal-representation-expectation.jsonld' * the id was changed to '\$	
	{temporal_entity_representation_id}'	
	* response body to be checked set to '\$	
	{response.json()}'	
	}	
TP Permutation id	TP/NGSI-LD/CI/Cons/TE/020_12_02	1
ir remutation id		

Name	020_12_02 Retrieve the simplified temporal representation of an entity with a	
	LanguageProperty property	
Parent Release	v1.4.1	
Description	Check that one can retrieve the temporal evolution of an entity with a LanguagePr	
	property	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}'	
	method set to 'GET'	
	Retrieve Temporal Representation of Entity and	
	Query Parameter: id set to '\$	
	{temporal_entity_representation_id}' and	
	Query Parameter: options set to 'temporalValues' and	
	Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	jsonld-contexts/ngsi-ld-test-suite-compound.jsonld'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Retrieve Temporal Representation Of Entity with Response Status	
	Code set to	
	200 and	
	Retrieve Temporal Representation Of Entity with Response Body	
	containing EntityTemporal element containing attribute instances in the time	
	range specified by the NGSI-	
	LD temporal query:	
	* the payload is defined in the file set to 'vehicle-	
	language-property-simplified-temporal-representation-expectation.jsonld'	
	* the id was changed to '\$	
	{temporal_entity_representation_id}'	
	* response body to be checked set to '\$	
	{response.json()}'	
	}	

TP Id	TP/NGSI-LD/CI/Cons/TE/020_07	
Test objective	Check that one cannot retrieve the temporal evolution of a non-existing entity	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.3	
Config Id	CF_01	
PICS Selection	PICS_TemEvoEntRet	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/TE/020_07_01	
Name	020_07_01 Retrieve the temporal evolution of a non-existing entity	
Parent Release	v1.3.1	
Description	Check that one cannot retrieve the temporal evolution of a non-existing entity	
Expected behaviour	Test events	Direction SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: id set to 'urn:ngsi- ld:Vehicle:unknowEntity' }	
	then { the SUT sends a valid Response for the operations: Retrieve Temporal Representation Of Entity with Response Status Code set to 404 and Retrieve Temporal Representation Of Entity with Response Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/ ResourceNotFound' and	SUT -> Client

Retrieve Temporal Representation Of Entity with Response body	
containing 'title' element	
}	

TP Id	TP/NGSI-LD/CI/Cons/TE/020_05	
Test objective	Check that one can retrieve the temporal evolution of the last N instances of e	ntity attributes
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.3	
Config Id	CF_01	
PICS Selection	PICS_TemEvoEntRet	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing an initial Tem	poral Entity
	\${entity} with an id set to \${temporal_entity_representation_id} and an attribute	
	to \${atrld} }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/TE/020_05_01	
Name	020_05_01 Retrieve Some Instances	
Parent Release	V1.3.1	
Description	Check that one can retrieve the temporal evolution of the last N instances of e	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}'	
	method set to 'GET'	
	Retrieve Temporal Representation of Entity and	
	Query Parameter: id set to '\$	
	{temporal_entity_representation_id}' and	
	Query Parameter: lastN set to '10' and	
	Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	jsonld-contexts/ngsi-ld-test-suite-compound.jsonld'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Retrieve Temporal Representation Of Entity with Response Status	
	Code set to	
	200 and	
	Retrieve Temporal Representation Of Entity with Response Body containing EntityTemporal element containing attribute instances in the time	
	range specified by the NGSI-	
	LD temporal query:	
	* the payload is defined in the file set to 'vehicle-	
	temporal-representation-020-05-01-expectation.jsonld'	
	* the id was changed to '\$	
	{temporal_entity_representation_id}'	
	* response body to be checked set to '\$	
	{response.json()}'	
	}	
TP Permutation id	TP/NGSI-LD/CI/Cons/TE/020_05_02	
Name	020_05_02 Retrieve All Instances	
Parent Release	V1.3.1	atitu attributas
Description Expected behaviour	Check that one can retrieve the temporal evolution of the last N instances of e Test events	Direction
Expected beliaviour	when {	SUT <- Client
	the SUT receives a Request from the client containing:	SOT <- Client
	URL set to '/	
	ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}'	
	method set to 'GET'	
	Retrieve Temporal Representation of Entity and	
	Query Parameter: id set to '\$	
	{temporal_entity_representation_id}' and	
	Query Parameter: lastN set to '20' and	
	Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	jsonld-contexts/ngsi-ld-test-suite-compound.jsonld'	
]}	



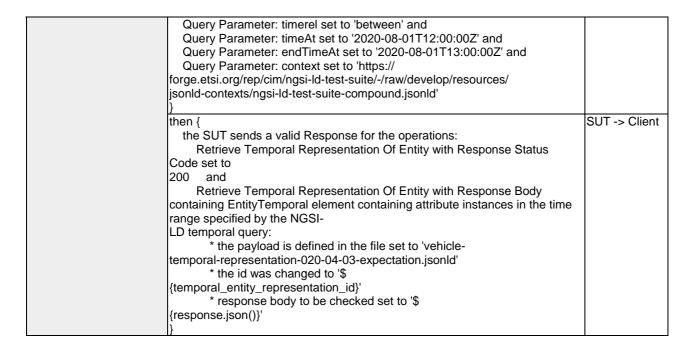
TP ld	TP/NGSI-LD/CI/Cons/TE/020_01	
Test objective	Check that one can retrieve the temporal evolution of an entity	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.3	
Config Id	CF_01	
PICS Selection	PICS_TemEvoEntRet	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/TE/020_01_01	
Name	020_01_01 Retrieve the temporal evolution of an entity	
Parent Release	v1.3.1	
Description	Check that one can retrieve the temporal evolution of an entity	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}'	
	method set to 'GET'	
	Retrieve Temporal Representation of Entity and	
	Query Parameter: id set to '\$	
	{temporal_entity_representation_id}'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Retrieve Temporal Representation Of Entity with Response Status	
	Code set to	
	200 and	
	Retrieve Temporal Representation Of Entity with Response Body	
	containing EntityTemporal element containing attribute instances in the time	
	range specified by the NGSI-	
	LD temporal query:	
	* the payload is defined in the file set to 'vehicle- temporal-representation-020-01-expectation.jsonId'	
	* the id was changed to '\$	
	{temporal_entity_representation_id}'	
	* response body to be checked set to '\$	
	{response.json()}'	
	[[tesponse.json()]	
	If	

TP ld	TP/NGSI-LD/CI/Cons/TE/020_03	
Test objective	Check that one can retrieve the temporal evolution of certain attributes of an entity	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.3	
Config Id	CF_01	
PICS Selection	PICS_TemEvoEntRet	
	with { the SUT being in the "initial state" and the SUT containing an initial Temporal Entity \${entity} with an id set to \${temporal_entity_representation_id} and an attribute with an id set to \${atrld} }	
Permutations Permutations		
TP Permutation id	TP/NGSI-LD/CI/Cons/TE/020 03 01	

Name	020_03_01 Retrieve the temporal evolution of certain attributes of an entity	
Parent Release	v1.3.1	
Description	Check that one can retrieve the temporal evolution of certain attributes of an e	entity
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attrs set to '\$ {temporal_attributes_to_be_retrieved}' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' } then {	SUT <- Client
	the SUT sends a valid Response for the operations: Retrieve Temporal Representation Of Entity with Response Status Code set to 200 and Retrieve Temporal Representation Of Entity with Response Body containing EntityTemporal element containing attribute instances in the time range specified by the NGSI- LD temporal query: * the payload is defined in the file set to 'vehicle- temporal-representation-020-03-expectation.jsonld' * the id was changed to '\$ {temporal_entity_representation_id}' * response body to be checked set to '\$ {response.json()}' }	SOT -> Client

TP ld	TP/NGSI-LD/CI/Cons/TE/020_04	
Test objective	Check that one can retrieve the temporal evolution of an entity matching the g	iven NGSI-LD
	temporal query	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.3	
Config Id	CF_01	
PICS Selection	PICS_TemEvoEntRet	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing an initial Tem	
	\${entity} with an id set to \${temporal_entity_representation_id} and an attribute	e with an id set
	to \${atrld} }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/TE/020_04_01	
Name	020_04_01 After	
Parent Release	v1.3.1	
Description	Check that one can retrieve the temporal evolution of an entity matching the g	iven NGSI-LD
	temporal query	1 5: .:
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}'	
	method set to 'GET'	
	Retrieve Temporal Representation of Entity and Query Parameter: id set to '\$	
	{temporal_entity_representation_id}' and	
	Query Parameter: timerel set to 'after' and	
	Query Parameter: timeAt set to '2020-08-01T13:03:00Z' and	
	Query Parameter: endTimeAt set to '\$	
	{EMPTY}' and	
	Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	jsonld-contexts/ngsi-ld-test-suite-compound.jsonld'	
	٠ - ١٠٠ - ١٠٠٠ - ١٠٠ - ١٠٠ - ١٠٠٠ - ١٠٠ - ١٠٠٠ - ١	1

	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Retrieve Temporal Representation Of Entity with Response Status	
	Code set to	
	200 and	
	Retrieve Temporal Representation Of Entity with Response Body	
	containing EntityTemporal element containing attribute instances in the time	
	range specified by the NGSI-	
	LD temporal query:	
	* the payload is defined in the file set to 'vehicle-	
	temporal-representation-020-04-01-expectation.jsonId'	
	* the id was changed to '\$	
	{temporal_entity_representation_id}'	
	* response body to be checked set to '\$	
	{response.json()}'	
TD Dames tation in	} 	
TP Permutation id	TP/NGSI-LD/CI/Cons/TE/020_04_02	
Name	020_04_02 Before	
Parent Release	v1.3.1	
Description	Check that one can retrieve the temporal evolution of an entity matching the g	iven NGSI-LD
	temporal query	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}	
	method set to 'GET'	
	Retrieve Temporal Representation of Entity and	
	Query Parameter: id set to '\$	
	{temporal_entity_representation_id}' and	
	Query Parameter: timerel set to 'before' and	
	Query Parameter: timeAt set to '2020-08-01T12:05:00Z' and	
	Query Parameter: endTimeAt set to '\$	
	{EMPTY}' and	
	Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	jsonld-contexts/ngsi-ld-test-suite-compound.jsonld'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Retrieve Temporal Representation Of Entity with Response Status	
	Code set to	
	200 and	
	Retrieve Temporal Representation Of Entity with Response Body	
	containing EntityTemporal element containing attribute instances in the time	
	range specified by the NGSI-	
	II D temporal query:	
	LD temporal query: * the payload is defined in the file set to 'vehicle-	
	* the payload is defined in the file set to 'vehicle-	
	* the payload is defined in the file set to 'vehicle-temporal-representation-020-04-02-expectation.jsonId'	
	* the payload is defined in the file set to 'vehicle- temporal-representation-020-04-02-expectation.jsonId' * the id was changed to '\$	
	* the payload is defined in the file set to 'vehicle- temporal-representation-020-04-02-expectation.jsonId' * the id was changed to '\$ {temporal_entity_representation_id}'	
	* the payload is defined in the file set to 'vehicle- temporal-representation-020-04-02-expectation.jsonld' * the id was changed to '\$ {temporal_entity_representation_id}' * response body to be checked set to '\$	
	* the payload is defined in the file set to 'vehicle- temporal-representation-020-04-02-expectation.jsonId' * the id was changed to '\$ {temporal_entity_representation_id}'	
	* the payload is defined in the file set to 'vehicle- temporal-representation-020-04-02-expectation.jsonId'	
TP Permutation id	* the payload is defined in the file set to 'vehicle- temporal-representation-020-04-02-expectation.jsonld'	
Name	* the payload is defined in the file set to 'vehicle- temporal-representation-020-04-02-expectation.jsonId'	
Name Parent Release	* the payload is defined in the file set to 'vehicle- temporal-representation-020-04-02-expectation.jsonId'	
Name	* the payload is defined in the file set to 'vehicle- temporal-representation-020-04-02-expectation.jsonId'	iven NGSI-LD
Name Parent Release Description	* the payload is defined in the file set to 'vehicle- temporal-representation-020-04-02-expectation.jsonId'	
Name Parent Release	* the payload is defined in the file set to 'vehicle- temporal-representation-020-04-02-expectation.jsonId'	Direction
Name Parent Release Description	* the payload is defined in the file set to 'vehicle- temporal-representation-020-04-02-expectation.jsonId'	
Name Parent Release Description	* the payload is defined in the file set to 'vehicle- temporal-representation-020-04-02-expectation.jsonId'	Direction
Name Parent Release Description	* the payload is defined in the file set to 'vehicle- temporal-representation-020-04-02-expectation.jsonId'	Direction
Name Parent Release Description	* the payload is defined in the file set to 'vehicle- temporal-representation-020-04-02-expectation.jsonId'	Direction
Name Parent Release Description	* the payload is defined in the file set to 'vehicle- temporal-representation-020-04-02-expectation.jsonId'	Direction
Name Parent Release Description	* the payload is defined in the file set to 'vehicle- temporal-representation-020-04-02-expectation.jsonId' * the id was changed to '\$ {temporal_entity_representation_id}' * response body to be checked set to '\$ {response.json()}' } TP/NGSI-LD/CI/Cons/TE/020_04_03 020_04_03 Between v1.3.1 Check that one can retrieve the temporal evolution of an entity matching the g temporal query Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'GET'	Direction
Name Parent Release Description	* the payload is defined in the file set to 'vehicle- temporal-representation-020-04-02-expectation.jsonId'	Direction
Name Parent Release Description	* the payload is defined in the file set to 'vehicle- temporal-representation-020-04-02-expectation.jsonId' * the id was changed to '\$ {temporal_entity_representation_id}' * response body to be checked set to '\$ {response.json()}' } TP/NGSI-LD/CI/Cons/TE/020_04_03 020_04_03 Between v1.3.1 Check that one can retrieve the temporal evolution of an entity matching the g temporal query Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'GET'	Direction



TP ld	TP/NGSI-LD/CI/Cons/TE/020_06	
Test objective	Check that one cannot retrieve the temporal evolution of an entity with an invalid id (invalid	
D. C	URI)	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.3	
Config Id	CF_01	
PICS Selection	PICS_TemEvoEntRet	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/TE/020_06_01	
Name	020_06_01 Retrieve the temporal evolution of an entity with an invalid id	
Parent Release	v1.3.1	
Description	Check that one cannot retrieve the temporal evolution of an entity with an inv URI)	alid id (invalid
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: id set to 'invalidUri' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Retrieve Temporal Representation Of Entity with Response Status Code set to 400 and Retrieve Temporal Representation Of Entity with Response Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/ BadRequestData' and Retrieve Temporal Representation Of Entity with Response body containing 'title' element }	SUT -> Client

TP Id	TP/NGSI-LD/CI/Cons/TE/020_02
Test objective	Check that one can retrieve the temporal evolution of an entity using a context
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.3
Config Id	CF_01
PICS Selection	PICS_TemEvoEntRet
Initial conditions	with { the SUT containing an initial state }

	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/TE/020_02_01	
Name	020_02_01 Retrieve the temporal evolution of an entity using a context	
Parent Release	v1.3.1	
Description	Check that one can retrieve the temporal evolution of an entity using a context	t
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'GET' Retrieve Temporal Representation of Entity and Query Parameter: id set to '\$ {temporal_entity_representation_id}' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Retrieve Temporal Representation Of Entity with Response Status Code set to 200 and Retrieve Temporal Representation Of Entity with Response Body containing EntityTemporal element containing attribute instances in the time range specified by the NGSI- LD temporal query: * the payload is defined in the file set to 'vehicle- temporal-representation-020-02-expectation.jsonId' * the id was changed to '\$ {temporal_entity_representation_id}' * response body to be checked set to '\$ {response.json()}' }	SUT -> Client

4.4.1.1.2 QueryTemporalEvolutionOfEntities

TP ld	TP/NGSI-LD/CI/Cons/TE/021_13	
Test objective	Check that one can query the temporal evolution of entities using the entityOperations method	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.4	
Config Id	CF_01	
PICS Selection	PICS_QueTemEvoEnt_AND_QueTemEvoEntPos	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing an a list of Te	mporal
	Entities }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/TE/021_13_01	
Name	021_13_01 After	
Parent Release	v1.3.1	
Description	Check that one can query the temporal evolution of entities using the entityOp	erations
	method	_
Expected behaviour	Test events	Direction
=xpootoa sonarioa:	1000 040110	
Exposion somerion	when {	SUT <- Client
	when { the SUT receives a Request from the client containing:	
	when { the SUT receives a Request from the client containing: URL set to '/	
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entityOperations/query'	
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entityOperations/query' method set to 'POST'	
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entityOperations/query' method set to 'POST' Query Temporal Representation of Entities Via POST Request: and	
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entityOperations/query' method set to 'POST' Query Temporal Representation of Entities Via POST Request: and Query Parameter: query_file_name set to 'entity-	
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entityOperations/query' method set to 'POST' Query Temporal Representation of Entities Via POST Request: and Query Parameter: query_file_name set to 'entity-operations-after-query.jsonld' and	
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entityOperations/query' method set to 'POST' Query Temporal Representation of Entities Via POST Request: and Query Parameter: query_file_name set to 'entity- operations-after-query.jsonId' and Query Parameter: Link set to '<	
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entityOperations/query' method set to 'POST' Query Temporal Representation of Entities Via POST Request: and Query Parameter: query_file_name set to 'entity- operations-after-query.jsonld' and Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/	
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entityOperations/query' method set to 'POST' Query Temporal Representation of Entities Via POST Request: and Query Parameter: query_file_name set to 'entity- operations-after-query.jsonld' and Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/ resources/jsonld-contexts/ngsi-ld-test-suite-	
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entityOperations/query' method set to 'POST' Query Temporal Representation of Entities Via POST Request: and Query Parameter: query_file_name set to 'entity- operations-after-query.jsonld' and Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/ resources/jsonld-contexts/ngsi-ld-test-suite- compound.jsonld>; rel="http://www.w3.org/ns/json-	
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entityOperations/query' method set to 'POST' Query Temporal Representation of Entities Via POST Request: and Query Parameter: query_file_name set to 'entity- operations-after-query.jsonld' and Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/ resources/jsonld-contexts/ngsi-ld-test-suite-	

•		
	json'	
	then { the SUT sends a valid Response for the operations:	SUT -> Client
	Query Temporal Representation Of Entities Via Post with Response	
	Status Code set to 200 and	
	Query Temporal Representation Of Entities Via Post with Request	
	response body containing a list that contains Entity Temporal Elements	
	compared with file 'vehicles-	
	temporal-representation-021-13-01-expectation.jsonld'	
	and using the list of entity ids define in '\$	
	{temporal_entities_representation_ids}'	
TP Permutation id	TP/NGSI-LD/CI/Cons/TE/021_13_02	
Name	021_13_02 Before	
Parent Release	V1.3.1	arations
Description	Check that one can query the temporal evolution of entities using the entityOp method	erations
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/	
	ngsi-ld/v1/temporal/entityOperations/query' method set to 'POST'	
	Query Temporal Representation of Entities Via POST Request: and	
	Query Parameter: query_file_name set to 'entity-	
	operations-before-query.jsonld' and	
	Query Parameter: Link set to '<	
	\$https://forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/	
	resources/jsonld-contexts/ngsi-ld-test-suite-compound.jsonld>; rel="http://www.w3.org/ns/json-	
	Id#context";type="application/Id+json" and	
	Query Parameter: content_type set to 'application/	
	ison'	
	[}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Query Temporal Representation Of Entities Via Post with Response	
	Status Code set to 200 and	
	Query Temporal Representation Of Entities Via Post with Request	
	response body containing a list that contains Entity Temporal Elements	
	compared with file 'vehicles-	
	temporal-representation-021-13-02-expectation.jsonId'	
	and using the list of entity ids define in '\$	
	{temporal_entities_representation_ids}'	
	}	

I== · ·	
TP ld	TP/NGSI-LD/CI/Cons/TE/021_08
Test objective	Check that one can query the temporal evolution of entities matching the given NGSI-LD
-	query
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.4
Config Id	CF_01
PICS Selection	PICS_QueTemEvoEnt_AND_QueTemEvoEntPos
Initial conditions	with { the SUT containing an initial state }
	Permutations
TP Permutation id	TP/NGSI-LD/CI/Cons/TE/021_08_01
Name	021_08_01 Query the temporal evolution of entities matching the given NGSI-LD query
Parent Release	v1.3.1
Description	Check that one can query the temporal evolution of entities matching the given NGSI-LD
	query

Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/temporal/entities'	
	method set to 'GET'	
	Query Temporal Representation of Entities and	
	Query Parameter: entity_types set to '\$	
	{entity_types_to_be_retrieved}' and Query Parameter: ngsild_query set to 'speed>	
	90' and	
	Query Parameter: timerel set to 'after' and	
	Query Parameter: timeAt set to '2020-07-01T12:05:00Z' and	
	Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	jsonld-contexts/ngsi-ld-test-suite-compound.jsonld	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Query Temporal Representation Of Entities with Response Status	
	Code set to	
	200 and	
	Query Temporal Representation Of Entities with Request response	
	body containing a list that contains Entity Temporal Elements compared with file 'vehicles-	
	temporal-representation-021-08-expectation.jsonId	
	and using the list of entity ids define in '\$	
	{temporal_entities_representation_ids}'	
	}	

TP ld	TP/NGSI-LD/CI/Cons/TE/021_11	
Test objective	Check that one can guery the temporal evolution of entities with a limit to the number of	
,	entities to be retrieved	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.4	
Config Id	CF 01	
PICS Selection	PICS_QueTemEvoEnt_AND_QueTemEvoEntPos	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing an a list of	Temporal
	Entities }	·
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/TE/021_11_01	
Name	021_11_01 Query Some entities	
Parent Release	v1.3.1	
Description	Check that one can query the temporal evolution of entities with a limit to the	ne number of
	entities to be retrieved	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/temporal/entities'	
	method set to 'GET'	
	Query Temporal Representation of Entities and	
	Query Parameter: entity_types set to '\$	
	{entity_types_to_be_retrieved}' and	
	Query Parameter: limit set to '\${2}	
	and	
	Query Parameter: timerel set to 'after' and	
	Query Parameter: timeAt set to '2020-07-01T12:05:00Z' and	
	Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	jsonld-contexts/ngsi-ld-test-suite-compound.jsonld'	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	SUT -> CITEFIT
	Query Temporal Representation Of Entities with Response Status	
	Code set to	
	200 and	
	Query Temporal Representation Of Entities with Response Body	

	containing a list of entities equal to '\${2}	
	of type 'Vehicle' with response set to '\$	
	{response.json()}'	
	}	
TP Permutation id	TP/NGSI-LD/CI/Cons/TE/021_11_02	1
Name	021_11_02 Query All entities	
Parent Release	v1.3.1	
Description	Check that one can query the temporal evolution of entities with a limit to the	number of
•	entities to be retrieved	
Expected behaviour	Test events	Direction
•	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/temporal/entities'	
	method set to 'GET'	
	Query Temporal Representation of Entities and	
	Query Parameter: entity_types set to '\$	
	{entity_types_to_be_retrieved}' and	
	Query Parameter: limit set to '\${3}	
	' and	
	Query Parameter: timerel set to 'after' and	
	Query Parameter: timeAt set to '2020-07-01T12:05:00Z' and	
	Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	jsonld-contexts/ngsi-ld-test-suite-compound.jsonld'	
	j	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Query Temporal Representation Of Entities with Response Status	
	Code set to	
	200 and	
	Query Temporal Representation Of Entities with Response Body	
	containing a list of entities equal to '\${3}	
	of type 'Vehicle' with response set to '\$	
	{response.json()}'	
	}	
	I)	ı

TP ld	TP/NGSI-LD/CI/Cons/TE/021_14	
Test objective	Check that one can query the temporal evolution of entities with the simplified	representation
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.4	
Config Id	CF_01	
PICS Selection	PICS_QueTemEvoEnt_AND_QueTemEvoEntPos	
Initial conditions	with { the SUT being in the "initial state" and and containing a list of entities }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/TE/021_14_01	·
Name	021_14_01 After	
Parent Release	v1.3.1	
Description	Check that one can query the temporal evolution of entities with the simplified	representation
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'GET' Query Temporal Representation of Entities and Query Parameter: entity_types set to '\$ {entity_types_to_be_retrieved}' and Query Parameter: timerel set to 'after' and Query Parameter: timeAt set to '2020-08-01T12:04:00Z' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: options set to 'temporalValues' }	SUT <- Client

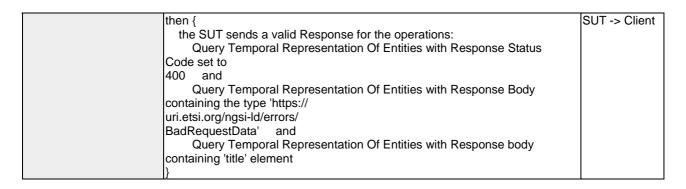
	lu (OUT OF
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Query Temporal Representation Of Entities with Response Status	
	Code set to	
	200 and	
	Query Temporal Representation Of Entities with Request response	
	body containing a list that contains Entity Temporal Elements	
	compared with file 'vehicles-	
	temporal-representation-021-14-01-expectation.json'	
	and using the list of entity ids define in '\$	
	{temporal_entities_representation_ids}'	
	{temporal_emilies_represemation_ids}	
TP Permutation id	TP/NGSI-LD/CI/Cons/TE/021_14_02	
Name	021_14_02 Before	
Parent Release	v1.3.1	
Description	Check that one can query the temporal evolution of entities with the simplified	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/temporal/entities'	
	method set to 'GET'	
	Query Temporal Representation of Entities and	
	Query Parameter: entity_types set to '\$	
	{entity_types_to_be_retrieved}' and	
	Query Parameter: timerel set to 'before' and	
	Query Parameter: timeAt set to '2020-09-01T13:06:00Z' and	
	Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and	
	Query Parameter: options set to 'temporalValues'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Query Temporal Representation Of Entities with Response Status	
	Code set to	
	200 and	
	Query Temporal Representation Of Entities with Request response	
	body containing a list that contains Entity Temporal Elements	
	compared with file 'vehicles-	
	temporal-representation-021-14-02-expectation.json'	
	and using the list of entity ids define in '\$	
	{temporal_entities_representation_ids}'	

TP ld	TP/NGSI-LD/CI/Cons/TE/021_09	
Test objective	Check that one can query the temporal evolution of entities matching the give	n NGSI-LD
	geo-query	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.4	
Config Id	CF_01	
PICS Selection	PICS_QueTemEvoEnt_AND_QueTemEvoEntPos	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing an a list of Te Entities }	emporal
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/TE/021_09_01	
Name	021_09_01 Near Point	
Parent Release	v1.3.1	
Description	Check that one can query the temporal evolution of entities matching the given NGSI-LD	
	geo-query	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/temporal/entities'	
	method set to 'GET'	
	Query Temporal Representation of Entities and	

	Query Parameter: entity_types set to '\$	
	{entity_types_to_be_retrieved}' and	
	Query Parameter: georel set to 'near;maxDistance==2000' and	
	Query Parameter: geometry set to 'Point' and	
	Query Parameter: coordinates set to '[-8.503,41.202]	
	'and	
	Query Parameter: geoproperty set to '\$	
	{EMPTY}' and	
	Query Parameter: timerel set to 'after' and	
	Query Parameter: timeAt set to '2020-07-01T12:05:00Z' and	
	Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	jsonld-contexts/ngsi-ld-test-suite-compound.jsonld'	
	<u>}</u>	OUT Olises
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Query Temporal Representation Of Entities with Response Status	
	Code set to	
	200 and	
	Query Temporal Representation Of Entities with Request response	
	body containing a list that contains Entity Temporal Elements]
	compared with file 'vehicles-	
	temporal-representation-021-09-expectation.jsonId	
	and using the list of entity ids define in '\$	
	{temporal_entities_representation_ids}'	
TP Permutation id	TP/NGSI-LD/CI/Cons/TE/021_09_02	
Name	021_09_02 Within Polygon	
Parent Release	v1.3.1	
Description	Check that one can query the temporal evolution of entities matching the give	en NGSI-I D
Description	geo-query	SII NOOI-LD
Expected behaviour	Test events	Direction
=xpootou ponaviou:		SLIT - Cliont
_xpootou bonarrour	when {	SUT <- Client
	when { the SUT receives a Request from the client containing:	SUT <- Client
	when { the SUT receives a Request from the client containing: URL set to '/	SUT <- Client
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities'	SUT <- Client
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'GET'	SUT <- Client
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities' method set to 'GET' Query Temporal Representation of Entities and	SUT <- Client
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities' method set to 'GET' Query Temporal Representation of Entities and Query Parameter: entity_types set to '\$	SUT <- Client
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities' method set to 'GET' Query Temporal Representation of Entities and Query Parameter: entity_types set to '\$ {entity_types_to_be_retrieved}' and	SUT <- Client
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'GET' Query Temporal Representation of Entities and Query Parameter: entity_types set to '\$ {entity_types_to_be_retrieved}' and Query Parameter: georel set to 'contains' and	SUT <- Client
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'GET' Query Temporal Representation of Entities and Query Parameter: entity_types set to '\$ {entity_types_to_be_retrieved}' and Query Parameter: georel set to 'contains' and Query Parameter: geometry set to 'Polygon' and	SUT <- Client
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'GET' Query Temporal Representation of Entities and Query Parameter: entity_types set to '\$ {entity_types_to_be_retrieved}' and Query Parameter: georel set to 'contains' and	SUT <- Client
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'GET' Query Temporal Representation of Entities and Query Parameter: entity_types set to '\$ {entity_types_to_be_retrieved}' and Query Parameter: georel set to 'contains' and Query Parameter: geometry set to 'Polygon' and	SUT <- Client
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'GET' Query Temporal Representation of Entities and Query Parameter: entity_types set to '\$ {entity_types_to_be_retrieved}' and Query Parameter: georel set to 'contains' and Query Parameter: geometry set to 'Polygon' and Query Parameter: coordinates set to '[[[-13.503,47.202],	SUT <- Client
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'GET' Query Temporal Representation of Entities and Query Parameter: entity_types set to '\$ {entity_types_to_be_retrieved}' and Query Parameter: georel set to 'contains' and Query Parameter: geometry set to 'Polygon' and Query Parameter: coordinates set to '[[[-13.503,47.202], [6.541, 52.961],[20.37,44.653],[9.46,32.57],[-13.503,32.57], [-13.503,47.202]]]' and	SUT <- Client
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'GET' Query Temporal Representation of Entities and Query Parameter: entity_types set to '\$ {entity_types_to_be_retrieved}' and Query Parameter: georel set to 'contains' and Query Parameter: geometry set to 'Polygon' and Query Parameter: coordinates set to '[[[-13.503,47.202], [6.541, 52.961],[20.37,44.653],[9.46,32.57],[-13.503,32.57], [-13.503,47.202]]]' and Query Parameter: geoproperty set to 'location' and	SUT <- Client
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'GET' Query Temporal Representation of Entities and Query Parameter: entity_types set to '\$ {entity_types_to_be_retrieved}' and Query Parameter: georel set to 'contains' and Query Parameter: geometry set to 'Polygon' and Query Parameter: coordinates set to '[[[-13.503,47.202], [6.541, 52.961],[20.37,44.653],[9.46,32.57],[-13.503,32.57], [-13.503,47.202]]]' and Query Parameter: geoproperty set to 'location' and Query Parameter: timerel set to 'after' and	SUT <- Client
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'GET' Query Temporal Representation of Entities and Query Parameter: entity_types set to '\$ {entity_types_to_be_retrieved}' and Query Parameter: georel set to 'contains' and Query Parameter: geometry set to 'Polygon' and Query Parameter: coordinates set to '[[[-13.503,47.202], [6.541, 52.961],[20.37,44.653],[9.46,32.57],[-13.503,32.57], [-13.503,47.202]]]' and Query Parameter: geoproperty set to 'location' and Query Parameter: timerel set to 'after' and Query Parameter: timeAt set to '2020-07-01T12:05:00Z' and	SUT <- Client
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'GET' Query Temporal Representation of Entities and Query Parameter: entity_types set to '\$ {entity_types_to_be_retrieved}' and Query Parameter: georel set to 'contains' and Query Parameter: geometry set to 'Polygon' and Query Parameter: coordinates set to '[[[-13.503,47.202], [6.541, 52.961],[20.37,44.653],[9.46,32.57],[-13.503,32.57], [-13.503,47.202]]]' and Query Parameter: geoproperty set to 'location' and Query Parameter: timerel set to 'after' and Query Parameter: timeAt set to '2020-07-01T12:05:00Z' and Query Parameter: context set to 'https://	SUT <- Client
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'GET' Query Temporal Representation of Entities and Query Parameter: entity_types set to '\$ {entity_types_to_be_retrieved}' and Query Parameter: georel set to 'contains' and Query Parameter: geometry set to 'Polygon' and Query Parameter: coordinates set to '[[[-13.503,47.202], [6.541, 52.961],[20.37,44.653],[9.46,32.57],[-13.503,32.57], [-13.503,47.202]]]' and Query Parameter: geoproperty set to 'location' and Query Parameter: timerel set to 'after' and Query Parameter: timeAt set to '2020-07-01T12:05:00Z' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	SUT <- Client
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'GET' Query Temporal Representation of Entities and Query Parameter: entity_types set to '\$ {entity_types_to_be_retrieved}' and Query Parameter: georel set to 'contains' and Query Parameter: geometry set to 'Polygon' and Query Parameter: coordinates set to '[[[-13.503,47.202], [6.541, 52.961],[20.37,44.653],[9.46,32.57],[-13.503,32.57], [-13.503,47.202]]]' and Query Parameter: geoproperty set to 'location' and Query Parameter: timerel set to 'after' and Query Parameter: timeAt set to '2020-07-01T12:05:00Z' and Query Parameter: context set to 'https://	SUT <- Client
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'GET' Query Temporal Representation of Entities and Query Parameter: entity_types set to '\$ {entity_types_to_be_retrieved}' and Query Parameter: georel set to 'contains' and Query Parameter: geometry set to 'Polygon' and Query Parameter: coordinates set to '[[[-13.503,47.202], [6.541, 52.961],[20.37,44.653],[9.46,32.57],[-13.503,32.57], [-13.503,47.202]]]' and Query Parameter: geoproperty set to 'location' and Query Parameter: timerel set to 'after' and Query Parameter: timeAt set to '2020-07-01T12:05:00Z' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' }	
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'GET' Query Temporal Representation of Entities and Query Parameter: entity_types set to '\$ {entity_types_to_be_retrieved}' and Query Parameter: georel set to 'contains' and Query Parameter: geometry set to 'Polygon' and Query Parameter: coordinates set to '[[[-13.503,47.202], [6.541, 52.961],[20.37,44.653],[9.46,32.57],[-13.503,32.57], [-13.503,47.202]]]' and Query Parameter: geoproperty set to 'location' and Query Parameter: timerel set to 'after' and Query Parameter: timeAt set to '2020-07-01T12:05:00Z' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' } then {	SUT <- Client SUT -> Client
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'GET' Query Temporal Representation of Entities and Query Parameter: entity_types set to '\$ {entity_types_to_be_retrieved}' and Query Parameter: georel set to 'contains' and Query Parameter: geometry set to 'Polygon' and Query Parameter: coordinates set to '[[[-13.503,47.202], [6.541, 52.961],[20.37,44.653],[9.46,32.57],[-13.503,32.57], [-13.503,47.202]]]' and Query Parameter: geoproperty set to 'location' and Query Parameter: timerel set to 'after' and Query Parameter: timeAt set to '2020-07-01T12:05:00Z' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' } then { the SUT sends a valid Response for the operations:	
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'GET' Query Temporal Representation of Entities and Query Parameter: entity_types set to '\$ {entity_types_to_be_retrieved}' and Query Parameter: georel set to 'contains' and Query Parameter: geometry set to 'Polygon' and Query Parameter: coordinates set to '[[[-13.503,47.202], [6.541, 52.961],[20.37,44.653],[9.46,32.57],[-13.503,32.57], [-13.503,47.202]]]' and Query Parameter: geoproperty set to 'location' and Query Parameter: timerel set to 'after' and Query Parameter: timeAt set to '2020-07-01T12:05:00Z' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' } then { the SUT sends a valid Response for the operations:	
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities' method set to 'GET' Query Temporal Representation of Entities and Query Parameter: entity_types set to '\$ {entity_types_to_be_retrieved}' and Query Parameter: georel set to 'contains' and Query Parameter: geometry set to 'Polygon' and Query Parameter: coordinates set to '[[[-13.503,47.202], [6.541, 52.961],[20.37,44.653],[9.46,32.57],[-13.503,32.57], [-13.503,47.202]]]' and Query Parameter: geoproperty set to 'location' and Query Parameter: timerel set to 'after' and Query Parameter: timeAt set to '2020-07-01T12:05:00Z' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-Id-test-suite-compound.jsonld' } then { the SUT sends a valid Response for the operations: Query Temporal Representation Of Entities with Response Status Code set to	
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities' method set to 'GET' Query Temporal Representation of Entities and Query Parameter: entity_types set to '\$ {entity_types_to_be_retrieved}' and Query Parameter: georel set to 'contains' and Query Parameter: geometry set to 'Polygon' and Query Parameter: coordinates set to '[[[-13.503,47.202], [6.541, 52.961],[20.37,44.653],[9.46,32.57],[-13.503,32.57], [-13.503,47.202]]]' and Query Parameter: geoproperty set to 'location' and Query Parameter: timerel set to 'after' and Query Parameter: timeAt set to '2020-07-01T12:05:00Z' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-Id-test-suite-compound.jsonld' } then { the SUT sends a valid Response for the operations: Query Temporal Representation Of Entities with Response Status Code set to 200 and	
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'GET' Query Temporal Representation of Entities and Query Parameter: entity_types set to '\$ {entity_types_to_be_retrieved}' and Query Parameter: georel set to 'contains' and Query Parameter: geometry set to 'Polygon' and Query Parameter: coordinates set to '[[[-13.503,47.202], [6.541, 52.961],[20.37,44.653],[9.46,32.57],[-13.503,32.57], [-13.503,47.202]]]' and Query Parameter: geoproperty set to 'location' and Query Parameter: timerel set to 'after' and Query Parameter: timeAt set to '2020-07-01T12:05:00Z' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' } then { the SUT sends a valid Response for the operations: Query Temporal Representation Of Entities with Response Status Code set to 200 and Query Temporal Representation Of Entities with Request response	
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities' method set to 'GET' Query Temporal Representation of Entities and Query Parameter: entity_types set to '\$ {entity_types_to_be_retrieved}' and Query Parameter: georel set to 'contains' and Query Parameter: geometry set to 'Polygon' and Query Parameter: coordinates set to '[[[-13.503,47.202], [6.541, 52.961],[20.37,44.653],[9.46,32.57],[-13.503,32.57], [-13.503,47.202]]]' and Query Parameter: geoproperty set to 'location' and Query Parameter: timerel set to 'after' and Query Parameter: timeAt set to '2020-07-01T12:05:00Z' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-Id-test-suite-compound.jsonld' } then { the SUT sends a valid Response for the operations: Query Temporal Representation Of Entities with Response Status Code set to 200 and Query Temporal Representation Of Entities with Request response body containing a list that contains Entity Temporal Elements	
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities' method set to 'GET' Query Temporal Representation of Entities and Query Parameter: entity_types set to '\$ {entity_types_to_be_retrieved}' and Query Parameter: geometry set to 'Polygon' and Query Parameter: geometry set to 'Polygon' and Query Parameter: coordinates set to '[[[-13.503,47.202], [6.541, 52.961],[20.37,44.653],[9.46,32.57],[-13.503,32.57], [-13.503,47.202]]]' and Query Parameter: geoproperty set to 'location' and Query Parameter: timerel set to 'after' and Query Parameter: timeAt set to '2020-07-01T12:05:00Z' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' } then { the SUT sends a valid Response for the operations: Query Temporal Representation Of Entities with Response Status Code set to 200 and Query Temporal Representation Of Entities with Request response body containing a list that contains Entity Temporal Elements compared with file 'vehicles-	
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities' method set to 'GET' Query Temporal Representation of Entities and Query Parameter: entity_types set to '\$ {entity_types_to_be_retrieved}' and Query Parameter: georel set to 'contains' and Query Parameter: geometry set to 'Polygon' and Query Parameter: coordinates set to '[[[-13.503,47.202], [6.541, 52.961],[20.37,44.653],[9.46,32.57],[-13.503,32.57], [-13.503,47.202]]]' and Query Parameter: geoproperty set to 'location' and Query Parameter: timerel set to 'after' and Query Parameter: timerel set to 'after' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' } then { the SUT sends a valid Response for the operations: Query Temporal Representation Of Entities with Response Status Code set to 200 and Query Temporal Representation Of Entities with Request response body containing a list that contains Entity Temporal Elements compared with file 'vehicles- temporal-representation-021-09-expectation.jsonld'	
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities' method set to 'GET' Query Temporal Representation of Entities and Query Parameter: entity_types set to '\$ {entity_types_to_be_retrieved}' and Query Parameter: georel set to 'contains' and Query Parameter: geometry set to 'Polygon' and Query Parameter: coordinates set to '[[[-13.503,47.202], [6.541, 52.961],[20.37,44.653],[9.46,32.57],[-13.503,32.57], [-13.503,47.202]]' and Query Parameter: geoproperty set to 'location' and Query Parameter: timerel set to 'after' and Query Parameter: timerel set to 'location' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' } then { the SUT sends a valid Response for the operations: Query Temporal Representation Of Entities with Response Status Code set to 200 and Query Temporal Representation Of Entities with Request response body containing a list that contains Entity Temporal Elements compared with file 'vehicles- temporal-representation-021-09-expectation.jsonld' and using the list of entity ids define in '\$	
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities' method set to 'GET' Query Temporal Representation of Entities and Query Parameter: entity_types set to '\$ {entity_types_to_be_retrieved}' and Query Parameter: georel set to 'contains' and Query Parameter: geometry set to 'Polygon' and Query Parameter: coordinates set to '[[[-13.503,47.202], [6.541, 52.961],[20.37,44.653],[9.46,32.57],[-13.503,32.57], [-13.503,47.202]]]' and Query Parameter: geoproperty set to 'location' and Query Parameter: timerel set to 'after' and Query Parameter: timerel set to 'after' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' } then { the SUT sends a valid Response for the operations: Query Temporal Representation Of Entities with Response Status Code set to 200 and Query Temporal Representation Of Entities with Request response body containing a list that contains Entity Temporal Elements compared with file 'vehicles- temporal-representation-021-09-expectation.jsonld'	

TP/NGSI-LD/CI/Cons/TE/021_10	
Check that one can query the temporal evolution of entities matching the giver Context Source filter	n NGSI-LD
ETSI GS CIM 009 V1.5.1 [1], clause 5.7.4	
CF_01	
PICS_QueTemEvoEnt_AND_QueTemEvoEntPos	
with { the SUT containing a Context Source Registration of a context source (temporal information of two entities of type Building between 2020-08-01T22:02021-08-01T21:07:00Z and CS1 containing two temporal entities of type Build temporal evolution of those entities in the mentioned interval. }	07:00Z and
	LD context
Check that one can query the temporal evolution of entities matching the given Context Source filter	n NGSI-LD
Test events	Direction
the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'GET' Query Temporal Representation of Entities and Query Parameter: entity_types set to '\$ {entity_types_to_be_retrieved}' and Query Parameter: csf set to 'endpoint' and Query Parameter: timerel set to 'after' and Query Parameter: timeAt set to '2020-07-01T12:05:00Z' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' } then { the SUT sends a valid Response for the operation: Query Temporal Representation Of Entities with Response Status	SUT <- Client
	Check that one can query the temporal evolution of entities matching the giver Context Source filter ETSI GS CIM 009 V1.5.1 [1], clause 5.7.4 CF_01 PICS_QueTemEvoEnt_AND_QueTemEvoEntPos with { the SUT containing a Context Source Registration of a context source (temporal information of two entities of type Building between 2020-08-01T22:02021-08-01T21:07:00Z and CS1 containing two temporal entities of type Build temporal evolution of those entities in the mentioned interval. } Permutations TP/NGSI-LD/CI/Cons/TE/021_10_01 021_10_01 Query the temporal evolution of entities matching the given NGSI-source filter V1.3.1 Check that one can query the temporal evolution of entities matching the given Context Source filter Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/V1/temporal/entities' method set to 'GET' Query Temporal Representation of Entities and Query Parameter: entity_types set to '\$ {entity_types_to_be_retrieved}' and Query Parameter: cof set to 'endpoint' and Query Parameter: timerel set to 'after' and Query Parameter: timerel set to 'after' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/ jsonId-contexts/ngsi-Id-test-suite-compound.jsonId' } then SUT sends a valid Response for the operation:

TP Id	TP/NGSI-LD/CI/Cons/TE/021_12	
Test objective	Check that one cannot query the temporal evolution of entities with an invalid r	equest or
•	invalid request content	•
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.4	
Config Id	CF_01	
PICS Selection	PICS_QueTemEvoEnt_AND_QueTemEvoEntPos	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/TE/021_12_01	
Name	021_12_01 Query the temporal evolution of entities with an invalid request	
Parent Release	v1.3.1	
Description	Check that one cannot query the temporal evolution of entities with an invalid r	request
Expected behaviour	Test events	Direction
Expected behaviour	Test events when {	Direction SUT <- Client
Expected behaviour	when { the SUT receives a Request from the client containing:	
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/	
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities'	
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'GET'	
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'GET' Query Temporal Representation of Entities and	
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'GET' Query Temporal Representation of Entities and Query Parameter: timerel set to 'after' and	
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'GET' Query Temporal Representation of Entities and Query Parameter: timerel set to 'after' and Query Parameter: timeAt set to '2020-07-01T12:05:00Z' and	
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'GET' Query Temporal Representation of Entities and Query Parameter: timerel set to 'after' and Query Parameter: timeAt set to '2020-07-01T12:05:00Z' and Query Parameter: context set to 'https://	
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'GET' Query Temporal Representation of Entities and Query Parameter: timerel set to 'after' and Query Parameter: timeAt set to '2020-07-01T12:05:00Z' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'GET' Query Temporal Representation of Entities and Query Parameter: timerel set to 'after' and Query Parameter: timeAt set to '2020-07-01T12:05:00Z' and Query Parameter: context set to 'https://	



TP Id	TP/NGSI-LD/CI/Cons/TE/021_07	
Test objective	Check that one can query the temporal evolution of entities matching the giver	id pattern
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.4	
Config Id	CF_01	
PICS Selection	PICS_QueTemEvoEnt_AND_QueTemEvoEntPos	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/TE/021_07_01	
Name	021_07_01 Query the temporal evolution of entities matching the given id patte	ern
Parent Release	v1.3.1	
Description	Check that one can query the temporal evolution of entities matching the giver	id pattern
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'GET' Query Temporal Representation of Entities and Query Parameter: entity_types set to '\$ {entity_types_to_be_retrieved}' and Query Parameter: entity_id_pattern set to 'urn:ngsi- ld:Vehicle:.*' and Query Parameter: timerel set to 'after' and Query Parameter: timeAt set to '2020-07-01T12:05:00Z' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Query Temporal Representation Of Entities with Response Status Code set to 200 and Query Temporal Representation Of Entities with Request response body containing a list that contains Entity Temporal Elements compared with file 'vehicles- temporal-representation-021-07-expectation.jsonId' and using the list of entity ids define in '\$ {temporal_entities_representation_ids}' }	SUT -> Client

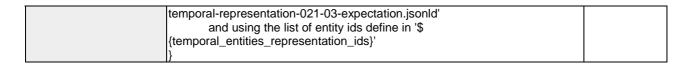
TP Id	TP/NGSI-LD/CI/Cons/TE/021_05
Test objective	Check that one can query the temporal evolution of entities matching the given type(s)
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.4
Config Id	CF_01
PICS Selection	PICS_QueTemEvoEnt_AND_QueTemEvoEntPos
Initial conditions	with { the SUT containing an initial state }
	Permutations
TP Permutation id	TP/NGSI-LD/CI/Cons/TE/021_05_01
Name	021_05_01 Query the temporal evolution of entities matching the given type(s)
Parent Release	v1.3.1
Description	Check that one can query the temporal evolution of entities matching the given type(s)

Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/temporal/entities'	
	method set to 'GET'	
	Query Temporal Representation of Entities and	
	Query Parameter: entity_types set to '\$	
	{entity_types_to_be_retrieved}' and	
	Query Parameter: timerel set to 'after' and	
	Query Parameter: timeAt set to '2020-07-01T12:05:00Z' and	
	Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	jsonld-contexts/ngsi-ld-test-suite-compound.jsonld'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Query Temporal Representation Of Entities with Response Status	
	Code set to	
	200 and	
	Query Temporal Representation Of Entities with Request response	
	body containing a list that contains Entity Temporal Elements	
	compared with file 'vehicles-	
	temporal-representation-021-05-expectation.jsonld'	
	and using the list of entity ids define in '\$	
	{temporal_entities_representation_ids}'	
	}	

TP ld	TP/NGSI-LD/CI/Cons/TE/021_01	
Test objective	Check that one can query the temporal evolution of entities	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.4	
Config Id	CF_01	
PICS Selection	PICS_QueTemEvoEnt_AND_QueTemEvoEntPos	
Initial conditions	with { the SUT being in the "initial state" and and containing a list of entities }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/TE/021_01_01	
Name	021_01_01 After	
Parent Release	v1.3.1	
Description	Check that one can query the temporal evolution of entities	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'GET' Query Temporal Representation of Entities and Query Parameter: entity_types set to '\$ {entity_types_to_be_retrieved}' and Query Parameter: timerel set to 'after' and Query Parameter: timeAt set to '2020-08-01T12:04:00Z' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Query Temporal Representation Of Entities with Response Status Code set to 200 and Query Temporal Representation Of Entities with Request response body containing a list that contains Entity Temporal Elements compared with file 'vehicles- temporal-representation-021-01-01-expectation.jsonld' and using the list of entity ids define in '\$ {temporal_entities_representation_ids}' }	SUT -> Client
TP Permutation id	TP/NGSI-LD/CI/Cons/TE/021_01_02	I

Name	021_01_02 Before	
Parent Release	v1.3.1	
Description	Check that one can query the temporal evolution of entities	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'GET' Query Temporal Representation of Entities and Query Parameter: entity_types set to '\$ {entity_types_to_be_retrieved}' and Query Parameter: timerel set to 'before' and Query Parameter: timeAt set to '2020-09-01T13:06:00Z' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Query Temporal Representation Of Entities with Response Status Code set to 200 and Query Temporal Representation Of Entities with Request response body containing a list that contains Entity Temporal Elements compared with file 'vehicles- temporal-representation-021-01-02-expectation.jsonld' and using the list of entity ids define in '\$ {temporal_entities_representation_ids}' }	SUT -> Client

TP Id	TP/NGSI-LD/CI/Cons/TE/021 03	
Test objective	Check that one can query the temporal evolution of the last N instances of enti	ties attributes
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.4	tics attributes
Config Id	CF 01	
PICS Selection	PICS_QueTemEvoEnt_AND_QueTemEvoEntPos	
Initial conditions	with { the SUT containing an initial state }	
initial conditions	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/TE/021_03_01	
Name	021_03_01 Query the temporal evolution of the last N instances of entities attr	ihutes
Parent Release	v1.3.1	ibatoo
Description	Check that one can query the temporal evolution of the last N instances of enti	ties attributes
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/temporal/entities'	
	method set to 'GET'	
	Query Temporal Representation of Entities and	
	Query Parameter: entity_types set to '\$	
	{entity_types_to_be_retrieved}' and	
	Query Parameter: timerel set to 'after' and	
	Query Parameter: timeAt set to '2020-07-01T12:05:00Z' and	
	Query Parameter: lastN set to '14' and	
	Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	jsonld-contexts/ngsi-ld-test-suite-compound.jsonld'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Query Temporal Representation Of Entities with Response Status	
	Code set to	
	200 and	
	Query Temporal Representation Of Entities with Request response	
	body containing a list that contains Entity Temporal Elements	
	compared with file 'vehicles-	



TDII	TRIVIOUS RIGHT	1
TP Id	TP/NGSI-LD/CI/Cons/TE/021_04	
Test objective	Check that one can query the temporal evolution of entities using a context	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.4	
Config Id	CF_01	
PICS Selection	PICS_QueTemEvoEnt_AND_QueTemEvoEntPos	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/TE/021_04_01	
Name	021_04_01 Query the temporal evolution of entities using a context	
Parent Release	v1.3.1	
Description	Check that one can query the temporal evolution of entities using a context	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/temporal/entities'	
	method set to 'GET'	
	Query Temporal Representation of Entities and	
	Query Parameter: entity_types set to '\$	
	{entity_types_to_be_retrieved}' and	
	Query Parameter: timerel set to 'after' and	
	Query Parameter: timeAt set to '2020-07-01T12:05:00Z' and	
	Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	jsonld-contexts/ngsi-ld-test-suite-compound.jsonld'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Query Temporal Representation Of Entities with Response Status	
	Code set to	
	200 and	
	Query Temporal Representation Of Entities with Request response	
	body containing a list that contains Entity Temporal Elements	
	compared with file 'vehicles-	
	temporal-representation-021-04-expectation.jsonld'	
	and using the list of entity ids define in '\$	
	{temporal_entities_representation_ids}'	
	}	

1		
TP ld	TP/NGSI-LD/CI/Cons/TE/021_06	
Test objective	Check that one can query the temporal evolution of entities matching the giver	identifier(s)
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.4	
Config Id	CF_01	
PICS Selection	PICS_QueTemEvoEnt_AND_QueTemEvoEntPos	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/TE/021_06_01	
Name	021_06_01 Query the temporal evolution of entities matching the given identifi	er(s)
Parent Release	v1.3.1	
Description	Check that one can query the temporal evolution of entities matching the given	identifier(s)
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/temporal/entities'	
	method set to 'GET'	
	Query Temporal Representation of Entities and	
	Query Parameter: entity_types set to '\$	
	{entity_types_to_be_retrieved}' and	
	Query Parameter: entity_ids set to '\$	

{entity_ids_to_be_retrieved}' and Query Parameter: timerel set to 'after' and Query Parameter: timeAt set to '2020-07-01T12:05:00Z' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' }	
then { the SUT sends a valid Response for the operations: Query Temporal Representation Of Entities with Response Status Code set to 200 and Query Temporal Representation Of Entities with Request response body containing a list that contains Entity Temporal Elements compared with file 'vehicles- temporal-representation-021-06-expectation.jsonId' and using the list of entity ids define in '\$ {temporal_entities_representation_ids}' }	SUT -> Client

TP Id	TP/NGSI-LD/CI/Cons/TE/021_02	
Test objective	Check that one can query the temporal evolution of certain attributes of entitie	S
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.4	
Config Id	CF_01	
PICS Selection	PICS_QueTemEvoEnt_AND_QueTemEvoEntPos	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/TE/021_02_01	
Name	021_02_01 Query the temporal evolution of certain attributes of entities	
Parent Release	v1.3.1	
Description	Check that one can query the temporal evolution of certain attributes of entitie	S
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'GET' Query Temporal Representation of Entities and Query Parameter: entity_types set to '\$	SUT <- Client
	{entity_types_to_be_retrieved}' and Query Parameter: timerel set to 'after' and Query Parameter: timeAt set to '2020-07-01T12:05:00Z' and Query Parameter: attrs set to '\$ {temporal_attributes_to_be_retrieved}' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' }	
	then { the SUT sends a valid Response for the operations: Query Temporal Representation Of Entities with Response Status Code set to 200 and Query Temporal Representation Of Entities with Request response body containing a list that contains Entity Temporal Elements compared with file 'vehicles- temporal-representation-021-02-expectation.jsonId' and using the list of entity ids define in '\$ {temporal_entities_representation_ids}' }	SUT -> Client

4.4.1.2 Discovery

4.4.1.2.1 RetrieveAvailableEntityTypes

TP Id	TP/NGSI-LD/CI/Cons/DISC/022_01	
Test objective	Check that one can retrieve a list of NGSI-LD entity types	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.5	
Config Id	CF_01	
PICS Selection	PICS_RetAvaEntTyp	
Initial conditions	with { the SUT being in the "initial state" and and containing a list of entities }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/DISC/022_01_01	
Name	022_01_01 WithoutJsonLdContext	
Parent Release	v1.3.1	
Description	Check that one can retrieve a list of NGSI-LD entity types	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/types' method set to 'GET' Retrieve entity types: and Query Parameter: context set to '\${EMPTY}' } then { the SUT sends a valid Response for the operations: Retrieve Entity Types with Response Status Code set to 200 and Retrieve Entity Types with Response Body containing an Entity Type	SUT <- Client SUT -> Client
TD Downstation id	List with expectation body equal to file: 'types/ expectations/entity-type-list-022-01-01-expectation.json' }	
TP Permutation id	TP/NGSI-LD/CI/Cons/DISC/022_01_02	
Name Daniel Balance	022_01_02 WithJsonLdContext	
Parent Release	V1.3.1	
Description	Check that one can retrieve a list of NGSI-LD entity types	T 51 //
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/types' method set to 'GET' Retrieve entity types: and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Retrieve Entity Types with Response Status Code set to 200 and Retrieve Entity Types with Response Body containing an Entity Type List with expectation body equal to file: 'types/ expectations/entity-type-list-022-01-02-expectation.json' }	SUT -> Client

4.4.1.2.2 RetrieveAvailableAttributes

TP Id	TP/NGSI-LD/CI/Cons/DISC/025_01
Test objective	Check that one can retrieve a list of NGSI-LD attributes
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.8
Config Id	CF_01
PICS Selection	PICS_AvaAtt
Initial conditions	with { the SUT being in the "initial state" and the SUT containing an initial Entity \${entity} with
	an id set to \${entityId} }

	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/DISC/025_01_01	
Name	025_01_01 WithoutJsonLdContext	
Parent Release	v1.3.1	
Description	Check that one can retrieve a list of NGSI-LD attributes	
Expected behaviour	Test events	Direction
•	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ngsi-ld/v1/attributes'	
	method set to 'GET'	
	Retrieve attributes: and	
	Query Parameter: context set to '\${EMPTY}'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Retrieve Attributes with Response Status Code set to	
	200 and	
	Retrieve Attributes with Response Body containing an Attribute List	
	element	
	* with filename set to 'types/	
	expectations/attribute-list-025-01-01-expectation.json'	
	* response set to '\$	
	{response.json()}'	
TP Permutation id	TP/NGSI-LD/CI/Cons/DISC/025_01_02	
Name	025_01_02 WithJsonLdContext	
Parent Release	v1.3.1	
Description	Check that one can retrieve a list of NGSI-LD attributes	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/attributes'	
	method set to 'GET'	
	Retrieve attributes: and	
	Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	jsonld-contexts/ngsi-ld-test-suite-compound.jsonld'	
	<u>}</u>	OUT Oli a sat
	then {	SUT -> Client
	the SUT sends a valid Response for the operations: Retrieve Attributes with Response Status Code set to	
	laaa .	
	200 and Retrieve Attributes with Response Body containing an Attribute List	
	element	
	* with filename set to 'types/	
	expectations/attribute-list-025-01-02-expectation.json'	
	* response set to '\$	
		1
	{response.ison()}'	
	{response.json()}'	

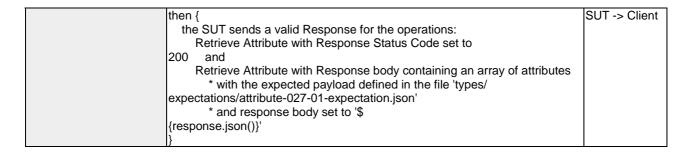
4.4.1.2.3 RetrieveDetailsOfAvailableEntityTypes

TP Id	TP/NGSI-LD/CI/Cons/DISC/023_01		
Test objective	Check that one can retrieve a list with a detailed representation of NGSI-LD entity types		
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.6		
Config Id	CF_01		
PICS Selection	PICS_RetAvaEntTyp		
Initial conditions	with { the SUT being in the "initial state" and and containing a list of entities }		
	Permutations		
TP Permutation id	TP/NGSI-LD/CI/Cons/DISC/023_01_01		
Name	023_01_01 WithoutJsonLdContext		
Parent Release	v1.3.1		
Description	Check that one can retrieve a list with a detailed representation of NGSI-LD entity types		

Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ngsi-ld/v1/types'	
	method set to 'GET'	
	Retrieve entity types: and	
	Query Parameter: context set to '\${EMPTY}' and	
	Query Parameter: details set to 'true'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Retrieve Entity Types with Response Status Code set to	
	200 and	
	Retrieve Entity Types with Response Body containing an Entity Type	
	Element with expectation body equal to file: 'types/	
	expectations/entity-type-023-01-01-expectation.json'	
TD Dammartation in		
TP Permutation id	TP/NGSI-LD/CI/Cons/DISC/023_01_02	
Name	023_01_02 WithJsonLdContext	
Parent Release	V1.3.1	***
Description	Check that one can retrieve a list with a detailed representation of NGSI-LD e	Direction
Expected behaviour		SUT <- Client
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/types' method set to 'GET'	
	Retrieve entity types: and	
	Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	jsonId-contexts/ngsi-Id-test-suite-compound.jsonId' and	
	Query Parameter: details set to 'true'	
	Rederly i diameter. details set to true	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	301 -> Client
	Retrieve Entity Types with Response Status Code set to	
	200 and	
	Retrieve Entity Types with Response Body containing an Entity Type	
	Element with expectation body equal to file: 'types/	
	expectations/entity-type-023-01-02-expectation.json'	

4.4.1.2.4 RetrieveAvailableAttributeInformation

TP ld	TP/NGSI-LD/CI/Cons/DISC/027_02	
Test objective	Check that one can retrieve a list with a detailed representation of NGSI-LD at	tributes
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.10	
Config Id	CF_01	
PICS Selection	PICS_DetAvaAtt	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/DISC/027_02_01	
Name	027_02_01 Retrieve Detailed Representation Of Available Attribute	
Parent Release	v1.3.1	
Description	Check that one can retrieve a list with a detailed representation of NGSI-LD at	tributes
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/attributes/{attribute_name}'	
	method set to 'GET'	
	Retrieve Attribute with attributeName set to 'airQualityLevel'	
	}	



TP/NGSI-LD/CI/Cons/DISC/027_01	
Check that one cannot retrieve a detailed representation of an unknown NGSI	-LD attribute
ETSI GS CIM 009 V1.5.1 [1], clause 5.7.10	
CF_01	
PICS_DetAvaAtt	
with { the SUT containing an initial state }	
Permutations	
TP/NGSI-LD/CI/Cons/DISC/027_01_01	
027_01_01 Retrieve Detailed Representation Of Available Attribute Without Co	ontext
v1.3.1	
Check that one cannot retrieve a detailed representation of an unknown NGSI	-LD attribute
Test events	Direction
the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/attributes/{attribute_name}' method set to 'GET' Retrieve Attribute with attributeName set to 'airQualityLevel' } then { the SUT sends a valid Response for the operations: Retrieve Attribute with Response Status Code set to 404 and Retrieve Attribute with Response Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/	SUT <- Client SUT -> Client
	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.10 CF_01 PICS_DetAvaAtt with { the SUT containing an initial state } Permutations TP/NGSI-LD/CI/Cons/DISC/027_01_01 027_01_01 Retrieve Detailed Representation Of Available Attribute Without Cov1.3.1 Check that one cannot retrieve a detailed representation of an unknown NGSI- Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/attributes/{attribute_name}' method set to 'GET' Retrieve Attribute with attributeName set to 'airQualityLevel' } then { the SUT sends a valid Response for the operations: Retrieve Attribute with Response Status Code set to 404

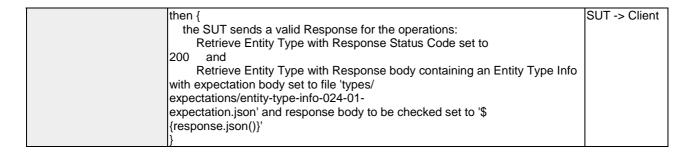
4.4.1.2.5 RetrieveDetailsOfAvailableAttributes

TP ld	TP/NGSI-LD/CI/Cons/DISC/026_01	
Test objective	Check that one can retrieve a list with a detailed representation of NGSI-LD att	ributes
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.9	
Config Id	CF_01	
PICS Selection	PICS_AvaAtt	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing an initial Entity	/ \${entity} with
	an id set to \${entityId} }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/DISC/026_01_01	
Name	026_01_01 WithoutJsonLdContext	
Parent Release	v1.3.1	
Description	Check that one can retrieve a list with a detailed representation of NGSI-LD att	ributes
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ngsi-ld/v1/attributes'	
	method set to 'GET'	
	Retrieve attributes: and	
	Query Parameter: context set to '\${EMPTY}' and	
	Query Parameter: details set to 'true'	
	}	

	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	SOT -> Client
	Retrieve Attributes with Response Status Code set to	
	200 and	
	Retrieve Attributes with Response body containing an array of	
	attributes	
	* with the expected payload defined in the file 'types/	
	expectations/attribute-026-01-01-expectation.json'	
	* and response body set to '\$	
	{response.json()}'	
	{response.json()}	
TP Permutation id	TP/NGSI-LD/CI/Cons/DISC/026_01_02	
Name	026_01_02 WithJsonLdContext	
Parent Release	v1.3.1	
Description	Check that one can retrieve a list with a detailed representation of NGSI-LI	O attributes
Expected behaviour	Test events	Direction
-	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/attributes'	
	method set to 'GET'	
	Retrieve attributes: and	
	Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and	
	Query Parameter: details set to 'true'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Retrieve Attributes with Response Status Code set to	
	200 and	
	Retrieve Attributes with Response body containing an array of	
	attributes	
	* with the expected payload defined in the file 'types/	
	expectations/attribute-026-01-02-expectation.json'	
	* and response body set to '\$	
	{response.json()}'	
	[(************************************	
	- U	

4.4.1.2.6 RetrieveAvailableEntityTypeInformation

TP ld	TP/NGSI-LD/CI/Cons/DISC/024_02	
Test objective	Check that one can retrieve a detailed representation of a specified NGSI-LD	entity type
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.7	
Config Id	CF_01	
PICS Selection	PICS_DetAvaEntTyp	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/DISC/024_02_01	
Name	024_02_01 Retrieve Detailed Representation Of Available Entity Type	
Parent Release	v1.3.1	
Description	Check that one can retrieve a detailed representation of a specified NGSI-LD	entity type
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	URL set to '/ ngsi-ld/v1/types/{type}'	
	ngsi-ld/v1/types/{type}' method set to 'GET' Retrieve Entity Type, with type set to 'Building', with Header['Link']	
	ngsi-ld/v1/types/{type}' method set to 'GET' Retrieve Entity Type, with type set to 'Building', with Header['Link'] containing 'https://	
	ngsi-ld/v1/types/{type}' method set to 'GET' Retrieve Entity Type, with type set to 'Building', with Header['Link'] containing 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	ngsi-ld/v1/types/{type}' method set to 'GET' Retrieve Entity Type, with type set to 'Building', with Header['Link'] containing 'https://	



TP Id	TP/NGSI-LD/CI/Cons/DISC/024_01	
Test objective	Check that one cannot retrieve a detailed representation of an unknown NGSI-	LD entity type
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.7	
Config Id	CF_01	
PICS Selection	PICS_DetAvaEntTyp	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/DISC/024_01_01	
Name	024_01_01 Retrieve Detailed Representation Of Available Entity Type Without	Context
Parent Release	v1.3.1	
Description	Check that one cannot retrieve a detailed representation of an unknown NGSI-	LD entity type
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/types/{type}'	
	method set to 'GET'	
	Retrieve Entity Type, with type set to 'Building'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Retrieve Entity Type with Response Status Code set to	
	404 and	
	Retrieve Entity Type with Response Body containing the type 'https://	
	uri.etsi.org/ngsi-ld/errors/	
	ResourceNotFound' and	
	Retrieve Entity Type with Response body containing 'title' element	
]}	

4.4.1.3 Entity

4.4.1.3.1 QueryEntities

TP ld	TP/NGSI-LD/CI/Cons/E/019_03_04	
Test objective	Check that one cannot query entities if the requested attribute names are inco	rrect
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.2	
Config Id	CF_01	
PICS Selection	PICS_QueEnt_AND_QueEntPos	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/E/019_03_04	
Name	019_03_04 Query several entities based on incorrect attribute names	
Parent Release	v1.3.1	
Description	Check that one cannot query entities if the requested attribute names are inco	rrect
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ngsi-ld/v1/entities/'	
	method set to 'GET'	
	Get Entities Request: and	
	Query Parameter: attrs set to '\${attributes_to_be_retrieved}'	
]}	

then {		SUT -> Client
the SUT sends	a valid Response for the operations:	
Create Entit	y Selecting Content Type with Response Status Code set	
to		
201 and		
Create Entit	y Selecting Content Type with Response Status Code set	
to		
201 and		
Query Entition	es with Response Status Code set to	
400 and		
Query Entition	es with Response Body containing the type 'https://	
uri.etsi.org/ngsi-lo	d/errors/	
BadRequestData	' and	
Query Entition	es with Response body containing 'title' element	
}		

TP ld	TP/NGSI-LD/CI/Cons/E/019_02_04	
Test objective	Check that one can query several entities via POST Interaction based on attrib	ute names
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.2	ato namoo
Config Id	CF 01	
PICS Selection	PICS_QueEnt_AND_QueEntPos	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/E/019_02_04	
Name	019_02_04 Query several entities via POST Interaction based on attribute name	nes
Parent Release	v1.3.1	
Description	Check that one can query several entities via POST Interaction based on attrib	oute names
Expected behaviour	Test events	Direction
•	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ngsi-ld/v1/entityOperations/query'	
	method set to 'POST'	
	Get Entities Via POST Request: and	
	Query Parameter: attrs set to '\${attributes_to_be_retrieved}' and	
	Query Parameter: content_type set to 'application/json' and	
	Query Parameter: accept set to 'application/json'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	oo i y onom
	ı '	
	· ·	
	}	
	Query Entities Via POST with Response Status Code set to 200 and Query Entities Via POST with Response Body containing a list containing Entity Elements and Query Parameter: expectation_filename set to 'vehicle- simple-attributes-core-context-expectation.json' and Query Parameter: entities_ids set to 'entities_ids_to_be_retrieved' and Query Parameter: ignore_core_context_version set to 'False' }	

TP ld	TP/NGSI-LD/CI/Cons/E/019_03_02		
Test objective	Check that one cannot query entities if the requested entity types are incorrect		
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.2		
Config Id	CF_01		
PICS Selection	PICS_QueEnt_AND_QueEntPos		
Initial conditions	with { the SUT containing an initial state }		
	Permutations		
TP Permutation id	TP/NGSI-LD/CI/Cons/E/019_03_02		
Name	019_03_02 Query entities based on incorrect entity types		
Parent Release	v1.3.1		
Description	Check that one cannot query entities if the requested entity types are incorrect		

Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ngsi-ld/v1/entities/'	
	method set to 'GET'	
	Get Entities Request: and	
	Query Parameter: entity_types set to '\${entity_types_to_be_retrieved}'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Create Entity Selecting Content Type with Response Status Code set	
	to	
	201 and	
	Create Entity Selecting Content Type with Response Status Code set	
	to	
	201 and	
	Query Entities with Response Status Code set to	
	400 and	
	Query Entities with Response Body containing the type 'https://	
	uri.etsi.org/ngsi-ld/errors/	
	BadRequestData' and	
	Query Entities with Response body containing 'title' element	
	[}	

_		
TP ld	TP/NGSI-LD/CI/Cons/E/019_02_02	
Test objective	Check that one can query several entities via POST Interaction based on the	entity type
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.2	
Config Id	CF_01	
PICS Selection	PICS_QueEnt_AND_QueEntPos	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/E/019_02_02	
Name	019_02_02 Query several entities via POST Interaction based on the entities t	ypes
Parent Release	v1.3.1	
Description	Check that one can query several entities via POST Interaction based on the	entity type
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entityOperations/query'	
	method set to 'POST'	
	Get Entities Via POST Request: and	
	Query Parameter: entity_types set to 'https://	
	ngsi-ld-test-suite/context#Vehicle' and	
	Query Parameter: accept set to 'application/	
	ld+json' and	
	Query Parameter: content_type set to 'application/	
	json'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Query Entities Via POST with Response Status Code set to	
	200 and	
	Query Entities Via POST with Response body containing a list	
	containing entity elements with different types and	
	Query Parameter: filename set to 'two-	
	vehicles-expectation.jsonld' and	
	Query Parameter: entities_representation_ids set to	
	'entities_ids_to_be_compared' and	
	Query Parameter: response_body set to 'response.json()' and	
	Query Parameter: ignore_core_context_version set to 'True'	
	}	

TP Id	TP/NGSI-LD/CI/Cons/E/019_03_05	
Test objective	Check that one cannot query entities if the request has a wrong geometryProp	perty
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.2	
Config Id	CF_01	
PICS Selection	PICS_QueEnt_AND_QueEntPos	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/E/019_03_05	
Name	019_03_05 Query entities when the request has a wrong geometryProperty	
Parent Release	v1.3.1	
Description	Check that one cannot query entities if the request has a wrong geometryProp	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/'	
	method set to 'GET'	
	Get Entities Request: and	
	Query Parameter: entity_types set to '\$	
	{entity_types_to_be_retrieved}' and	
	Query Parameter: geoproperty set to 'location' and	
	Query Parameter: georel set to 'invalid' and	
	Query Parameter: geometry set to 'Point' and Query Parameter: coordinates set to '[-8.503,41.202]	
	and	
	Query Parameter: Link set to '<	
	\$https://forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/	
	resources/jsonId-contexts/ngsi-Id-test-suite-	
	compound.jsonld>; rel="http://www.w3.org/ns/json-	
	ld#context";type="application/ld+json"	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Create Entity Selecting Content Type with Response Status Code set	
	to	
	201 and	
	Create Entity Selecting Content Type with Response Status Code set	
	to	
	201 and	
	Query Entities with Response Status Code set to	
	400 and	
	Query Entities with Response Body containing the type 'https://	
	uri.etsi.org/ngsi-ld/errors/	
	BadRequestData' and	
	Query Entities with Response body containing 'title' element	
	1}	

TP Id	TP/NGSI-LD/CI/Cons/E/019_02_05	
Test objective	Check that one can query entitites via POST Interaction asking for a GeoJSON	
	representation	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.2	
Config Id	CF_01	
PICS Selection	PICS_QueEnt_AND_QueEntPos	
Initial conditions	with { the SUT containing an initial state }	
Permutations		
TP Permutation id	TP/NGSI-LD/CI/Cons/E/019_02_05	
Name	019_02_05 Query several entities via POST Interaction asking for a GeoJSON	
	representation	
Parent Release	v1.3.1	
Description	Check that one can query entities via POST Interaction asking for a GeoJSON	
	representation	

Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entityOperations/query'	
	method set to 'POST'	
	Get Entities Via POST Request: and	
	Query Parameter: entity_types set to '\$	
	{entity_types_to_be_retrieved}' and	
	Query Parameter: accept set to 'application/	
	geo+json' and Ouery Peremeter: context set to 'https://	
	Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	isonld-contexts/ngsi-ld-test-suite-compound.jsonld' and	
	Query Parameter: content_type set to 'application/	
	ison'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Query Entities Via POST with Response Status Code set to	
	200 and	
	Query Entities Via POST with Check Response Body Content and	
	Query Parameter: expectation_filename set to 'vehicle-	
	parking-simple-expectation.geojson' and	
	Query Parameter: response_body set to 'response.json()' and	
	Query Parameter: additional_ignored_path set to '\$	
	[{EMPTY}'	
]}	

1		
TP ld	TP/NGSI-LD/CI/Cons/E/019_08	
Test objective	Query entities with Entity Type Selection Language.	
	ETSI GS CIM 009 V1.5.1 [1], clauses 4.17 and 5.7.2	
	CF_01	
	PICS_TYPESELECTLANG and PICS_QueEnt_AND_QueEntPos	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/E/019_08_01	
Name	019_08_01 query with one type	
Parent Release	v1.5.1	
Description	Query entities with Entity Type Selection Language.	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/' method set to 'GET' Get Entities Request: and Query Parameter: entity_types set to '\$ {entity_types_selection}' and Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/ resources/jsonld-contexts/ngsi-ld-test-suite- compound.jsonld>; rel="http://www.w3.org/ns/json- ld#context";type="application/ld+json"' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Query Entities with Response Status Code set to 200 and Query Entities with Response Body contains entities ids }	SUT -> Client
TP Permutation id	TP/NGSI-LD/CI/Cons/E/019_08_02	
	019_08_02 query with the AND operator	
Parent Release	v1.5.1	
Description	Query entities with Entity Type Selection Language.	

Expected behaviour	Test events	Direction
•	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/'	
	method set to 'GET'	
	Get Entities Request: and	
	Query Parameter: entity_types set to '\$	
	{entity_types_selection}' and	
	Query Parameter: Link set to '<	
	\$https://forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/	
	resources/jsonld-contexts/ngsi-ld-test-suite-	
	compound.jsonld>; rel="http://www.w3.org/ns/json-	
	d#context";type="application/ld+json"	
	<u>}</u>	OLUT OU
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Query Entities with Response Status Code set to	
	200 and	
	Query Entities with Response Body contains entities ids	
TP Permutation id	TP/NGSI-LD/CI/Cons/E/019_08_03	
Name	019_08_03 query with the OR operator	
Parent Release	v1.5.1	
Description	Query entities with Entity Type Selection Language.	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/'	
	method set to 'GET'	
	Get Entities Request: and	
	Query Parameter: entity_types set to '\$	
	{entity_types_selection}' and	
	Query Parameter: Link set to '<	
	\$https://forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/	
	resources/jsonld-contexts/ngsi-ld-test-suite-	
	compound.jsonId>; rel="http://www.w3.org/ns/json-	
	ld#context";type="application/ld+json"	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Query Entities with Response Status Code set to	
	200 and	
	Query Entities with Response Body contains entities ids	
	}	
TP Permutation id	TP/NGSI-LD/CI/Cons/E/019_08_04	
Name	019_08_04 different query with the OR operator	
Parent Release	v1.5.1	
Description	Query entities with Entity Type Selection Language.	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/	
	method set to 'GET'	
	Get Entities Request: and	
	Query Parameter: entity_types set to '\$	
	{entity_types_selection}' and	
	Query Parameter: Link set to '<	
	\$https://forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/	
	resources/jsonId-contexts/ngsi-Id-test-suite- compound.jsonId>; rel="http://www.w3.org/ns/json-	
	compound.jsonid>; rei= http://www.w3.org/ns/json- ld#context";type="application/ld+json"	
	Ш	I

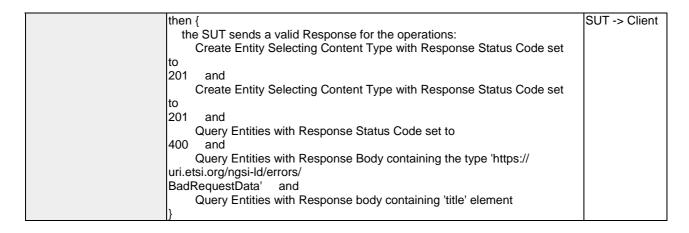
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Query Entities with Response Status Code set to	
	200 and	
	Query Entities with Response Body contains entities ids	
	}	
TP Permutation id	TP/NGSI-LD/CI/Cons/E/019_08_05	·
Name	019_08_05 query with two operators	
Parent Release	v1.5.1	
Description	Query entities with Entity Type Selection Language.	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/'	
	method set to 'GET'	
	Get Entities Request: and	
	Query Parameter: entity_types set to '\$	
	{entity_types_selection}' and	
	Query Parameter: Link set to '<	
	\$https://forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/	
	resources/jsonld-contexts/ngsi-ld-test-suite-	
	compound.jsonld>; rel="http://www.w3.org/ns/json-	
	ld#context";type="application/ld+json"	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Query Entities with Response Status Code set to	
	200 and	
	Query Entities with Response Body contains entities ids	
	}	

TP ld	TP/NGSI-LD/CI/Cons/E/019_03_01	
Test objective	Check that one cannot query entities if the requested ids are incorrect	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.2	
Config Id	CF_01	
PICS Selection	PICS_QueEnt_AND_QueEntPos	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/E/019_03_01	
Name	019_03_01 Query entities based on incorrect ids	
Parent Release	v1.3.1	
Description	Check that one cannot query entities if the requested ids are incorrect	
Expected behaviour	Test events	Direction
	the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/entities/' method set to 'GET' Get Entities Request: and Query Parameter: entity_ids set to '\${entities_ids_to_be_retrieved}' and Query Parameter: entity_types set to '\${entity_types_to_be_retrieved}' }	
	then { the SUT sends a valid Response for the operations: Create Entity Selecting Content Type with Response Status Code set to 201 and Create Entity Selecting Content Type with Response Status Code set to 201 and Query Entities with Response Status Code set to 400 and Query Entities with Response Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/ BadRequestData' and	SUT -> Client

Query Entities with Response body containing 'title' element	
J.	

TP Id	TP/NGSI-LD/CI/Cons/E/019_02_01	
Test objective	Check that one can query one entity via POST Interaction based on id	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.2	
Config Id	CF_01	
PICS Selection	PICS_QueEnt_AND_QueEntPos	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/E/019_02_01	
Name	019_02_01 Query one entity via POST Interaction based on id	
Parent Release	v1.3.1	
Description	Check that one can query one entity via POST Interaction based on id	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entityOperations/query' method set to 'POST' Get Entities Via POST Request: and Query Parameter: entity_ids set to '\$ {first_entity_id}' and Query Parameter: entity_types set to 'https:// ngsi-ld-test-suite/context#Building' and Query Parameter: content_type set to 'application/ json' and Query Parameter: accept set to 'application/ json' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Query Entities Via POST with Response Status Code set to 200 and Query Entities Via POST with Response Body containing a list containing Entity Elements and Query Parameter: expectation_filename set to 'building- minimal-expectation.json' and Query Parameter: entities_ids set to 'entities_ids_to_be_compared' and Query Parameter: ignore_core_context_version set to 'False' }	SUT -> Client

	·	
TP ld	TP/NGSI-LD/CI/Cons/E/019_03_03	
Test objective	Check that one cannot query entities if the requested id pattern is incorrect	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.2	
Config Id	CF_01	
PICS Selection	PICS_QueEnt_AND_QueEntPos	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/E/019_03_03	
Name	019_03_03 Query several entities based on incorrect id pattern	
Parent Release	v1.3.1	
Description	Check that one cannot query entities if the requested id pattern is incorrect	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/'	
	method set to 'GET'	
	Get Entities Request: and	
	Query Parameter: entity_id_pattern set to 'invalid_entity_id_pattern**' and	
	Query Parameter: entity_types set to '\$	
	{entity_types_to_be_retrieved}'	
	}	

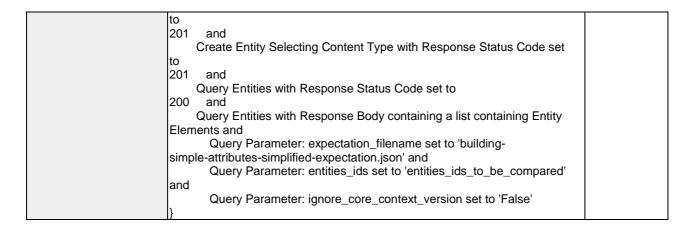


TP ld	TP/NGSI-LD/CI/Cons/E/019_02_03	
Test objective	Check that one can query several entities via POST Interaction based on the g	jiven id
•	pattern	,
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.2	
Config Id	CF_01	
PICS Selection	PICS QueEnt_AND_QueEntPos	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/E/019_02_03	
Name	019_02_03 Query several entities via POST Interaction based on the given id	pattern
Parent Release	v1.3.1	
Description	Check that one can query several entities via POST Interaction based on the g	given id
•	pattern	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entityOperations/query'	
	method set to 'POST'	
	Get Entities Via POST Request: and	
	Query Parameter: entity_id_pattern set to 'urn:ngsi-	
	ld:Building:.*' and	
	Query Parameter: entity_types set to 'https://	
	ngsi-ld-test-suite/context#Building' and	
	Query Parameter: content_type set to 'application/	
	json' and _	
	Query Parameter: accept set to 'application/	
	json'	
	}	OLIT OII (
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Query Entities Via POST with Response Status Code set to	
	200 and	
	Query Entities Via POST with Response Body containing a list	
	containing Entity Elements and	
	Query Parameter: expectation_filename set to 'building-minimal-expectation.json' and	
	Query Parameter: entities_ids set to 'entities_ids_to_be_compared'	
	and Query Parameter: ignore_core_context_version set to 'False'	
	Query i arameter. ignore_core_context_version set to Faise	
	- I/	

TP Id	TP/NGSI-LD/CI/Cons/E/019_01_02
Test objective	Check that one can query several entities based on the entities types
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.2
Config Id	CF_01
PICS Selection	PICS_QueEnt_AND_QueEntPos
Initial conditions	with { the SUT containing an initial state }

Permutations		
TP Permutation id	TP/NGSI-LD/CI/Cons/E/019_01_02	
Name	019_01_02 Query several entities based on the entities types	
Parent Release	v1.3.1	
Description	Check that one can query several entities based on the entities types	
Expected behaviour	Test events	Direction
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/entities/' method set to 'GET' Get Entities Request: and Query Parameter: entity_types set to '\${entity_types_to_be_retrieved}' and Query Parameter: accept set to 'application/ld+json' } then { the SUT sends a valid Response for the operations: Create Entity Selecting Content Type with Response Status Code set to 201 and Create Entity Selecting Content Type with Response Status Code set to 201 and	SUT -> Client
	Create Entity Selecting Content Type with Response Status Code set to 201 and Query Entities with Response Status Code set to 200 and Query Entities with Response body containing a list containing entity elements with different types and Query Parameter: filename set to 'two-types-vehicle-offstreetparking-expectation.jsonld' and Query Parameter: entities_representation_ids set to 'entities_ids_to_be_compared' and Query Parameter: response_body set to 'response.json()' and Query Parameter: ignore_core_context_version set to 'True' }	

TP Id	TP/NGSI-LD/CI/Cons/E/019_04	
Test objective	Check that the queried entities by Id can be returned in a simplified representa	ation
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 6.3.7	
Config Id	CF_01	
PICS Selection	PICS_HTTP_SIMP_REPR	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/E/019_04_01	
Name	019_04_01 Query entities in a simplified representation	
Parent Release	v1.3.1	
Description	Check that the queried entities by Id can be returned in a simplified representation	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ngsi-ld/v1/entities/'	
	method set to 'GET'	
	Get Entities Request: and	
	Query Parameter: entity_ids set to '\${entities_ids_to_be_retrieved}' and	
	Query Parameter: entity_types set to '\${entity_types_to_be_retrieved}'	
	and Query Parameter: options set to 'keyValues'	
	ly Query Farameter, options set to key values	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	301 -> Ciletit
	Create Entity Selecting Content Type with Response Status Code set	
	to	
	201 and	
	Create Entity Selecting Content Type with Response Status Code set	



TP ld	TP/NGSI-LD/CI/Cons/E/019_01_06	
Test objective	Check that one can query several entities based on scopes	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 4.19	
Config Id	CF_01	
PICS Selection	PICS_SCOPESELECTLANG	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/E/019_01_06	
Name	019_01_06 QueryWithFullScope	
Parent Release	v1.5.1	
Description	Check that one can query several entities based on scopes	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entities/' method set to 'GET' Get Entities Request: and Query Parameter: scopeq set to '/ Madrid/Gardens/ParqueNorte' and Query Parameter: entity_types set to 'https:// ngsi-Id-test-suite/context#Building' } then { the SUT sends a valid Response for the operations: Query Entities with Response Status Code set to 200 and Query Entities with Response Body containing a list of entities equal to '\${2} ' of type 'https://ngsi-Id-test-suite/ context#Building' with response set to '\$ {response.json()}'	SUT <- Client SUT -> Client
TP Permutation id	TP/NGSI-LD/CI/Cons/E/019_01_06	
Name	019_01_06 QueryWithPlusMatching	
Parent Release	v1.5.1	
Description	Check that one can query several entities based on scopes	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entities/' method set to 'GET' Get Entities Request: and Query Parameter: scopeq set to '/ Madrid/+/ParqueNorte' and Query Parameter: entity_types set to 'https:// ngsi-Id-test-suite/context#Building' }	SUT <- Client

	La ,	lour or
	then { the SUT sends a valid Response for the operations: Query Entities with Response Status Code set to	SUT -> Client
	200 and Query Entities with Response Body containing a list of entities equal to	
	\${2}	
	of type 'https://ngsi-ld-test-suite/	
	context#Building' with response set to '\$ {response.json()}'	
	{response.json()}	
TP Permutation id	TP/NGSI-LD/CI/Cons/E/019_01_06	l.
Name	019_01_06 QueryWithHashMatching	
Parent Release	v1.5.1	
Description	Check that one can query several entities based on scopes	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/	
	method set to 'GET'	
	Get Entities Request: and	
	Query Parameter: scopeq set to '/	
	CompanyA/#' and	
	Query Parameter: entity_types set to 'https://	
	ngsi-ld-test-suite/context#Building'	
	hon (CLIT . Client
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Query Entities with Response Status Code set to	
	200 and Query Entities with Response Body containing a list of entities equal to	
	\ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \	
	Φξ γ γ γ γ γ γ γ γ γ	
	context#Building' with response set to '\$	
	{response.json()}'	
	}	
TP Permutation id	TP/NGSI-LD/CI/Cons/E/019_01_06	1
Name	019_01_06 QueryNonEmptyScope	
Parent Release	v1.5.1	
Description	Check that one can query several entities based on scopes	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/	
	method set to 'GET'	
	Get Entities Request: and	
	Query Parameter: scopeq set to '/	
	#' and	
	Query Parameter: entity_types set to 'https://	
	Query Parameter: entity_types set to 'https:// ngsi-ld-test-suite/context#Building' }	
	Query Parameter: entity_types set to 'https:// ngsi-ld-test-suite/context#Building' } then {	SUT -> Client
	Query Parameter: entity_types set to 'https:// ngsi-ld-test-suite/context#Building' } then { the SUT sends a valid Response for the operations:	SUT -> Client
	Query Parameter: entity_types set to 'https:// ngsi-ld-test-suite/context#Building' } then { the SUT sends a valid Response for the operations: Query Entities with Response Status Code set to	SUT -> Client
	Query Parameter: entity_types set to 'https:// ngsi-ld-test-suite/context#Building' } then { the SUT sends a valid Response for the operations: Query Entities with Response Status Code set to 200 and	SUT -> Client
	Query Parameter: entity_types set to 'https:// ngsi-ld-test-suite/context#Building' } then { the SUT sends a valid Response for the operations: Query Entities with Response Status Code set to 200 and Query Entities with Response Body containing a list of entities equal to	SUT -> Client
	Query Parameter: entity_types set to 'https:// ngsi-ld-test-suite/context#Building' } then { the SUT sends a valid Response for the operations: Query Entities with Response Status Code set to 200 and Query Entities with Response Body containing a list of entities equal to '\${2}	SUT -> Client
	Query Parameter: entity_types set to 'https:// ngsi-ld-test-suite/context#Building' } then { the SUT sends a valid Response for the operations: Query Entities with Response Status Code set to 200 and Query Entities with Response Body containing a list of entities equal to '\${2} ' of type 'https://ngsi-ld-test-suite/	SUT -> Client
	Query Parameter: entity_types set to 'https:// ngsi-ld-test-suite/context#Building' } then { the SUT sends a valid Response for the operations: Query Entities with Response Status Code set to 200 and Query Entities with Response Body containing a list of entities equal to '\${2} ' of type 'https://ngsi-ld-test-suite/ context#Building' with response set to '\$	SUT -> Client
	Query Parameter: entity_types set to 'https:// ngsi-ld-test-suite/context#Building' } then { the SUT sends a valid Response for the operations: Query Entities with Response Status Code set to 200 and Query Entities with Response Body containing a list of entities equal to '\${2} ' of type 'https://ngsi-ld-test-suite/	SUT -> Client
TP Permutation id	Query Parameter: entity_types set to 'https:// ngsi-ld-test-suite/context#Building' } then { the SUT sends a valid Response for the operations: Query Entities with Response Status Code set to 200 and Query Entities with Response Body containing a list of entities equal to '\${2} ' of type 'https://ngsi-ld-test-suite/ context#Building' with response set to '\$ {response.json()}' }	SUT -> Client
TP Permutation id	Query Parameter: entity_types set to 'https:// ngsi-ld-test-suite/context#Building' } then { the SUT sends a valid Response for the operations: Query Entities with Response Status Code set to 200 and Query Entities with Response Body containing a list of entities equal to '\${2} ' of type 'https://ngsi-ld-test-suite/ context#Building' with response set to '\$ {response.json()}' } TP/NGSI-LD/CI/Cons/E/019_01_06	SUT -> Client
Name	Query Parameter: entity_types set to 'https:// ngsi-ld-test-suite/context#Building' } then { the SUT sends a valid Response for the operations: Query Entities with Response Status Code set to 200 and Query Entities with Response Body containing a list of entities equal to '\${2} ' of type 'https://ngsi-ld-test-suite/ context#Building' with response set to '\$ {response.json()}' } TP/NGSI-LD/CI/Cons/E/019_01_06 019_01_06 QueryWithAndScope	SUT -> Client
	Query Parameter: entity_types set to 'https:// ngsi-ld-test-suite/context#Building' } then { the SUT sends a valid Response for the operations: Query Entities with Response Status Code set to 200 and Query Entities with Response Body containing a list of entities equal to '\${2} ' of type 'https://ngsi-ld-test-suite/ context#Building' with response set to '\$ {response.json()}' } TP/NGSI-LD/CI/Cons/E/019_01_06	SUT -> Client

Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/'	
	method set to 'GET'	
	Get Entities Request: and	
	Query Parameter: scopeq set to '/	
	Madrid/Gardens/ParqueNorte;/CompanyA/OrganizationB/UnitC' and	
	Query Parameter: entity_types set to 'https://	
	ngsi-ld-test-suite/context#Building'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Query Entities with Response Status Code set to	
	200 and	
	Query Entities with Response Body containing a list of entities equal to	
	 \${1}	
	of type 'https://ngsi-ld-test-suite/	
	context#Building' with response set to '\$	
	{response.json()}'	
TP Permutation id	TP/NGSI-LD/CI/Cons/E/019_01_06	
Name	019_01_06 QueryWithOrScope	
Parent Release	v1.5.1	
Description	Check that one can query several entities based on scopes	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	URL set to '/ ngsi-ld/v1/entities/'	
	URL set to '/ ngsi-ld/v1/entities/' method set to 'GET'	
	URL set to '/ ngsi-ld/v1/entities/' method set to 'GET' Get Entities Request: and	
	URL set to '/ ngsi-Id/v1/entities/' method set to 'GET' Get Entities Request: and Query Parameter: scopeq set to '/	
	URL set to '/ ngsi-Id/v1/entities/' method set to 'GET' Get Entities Request: and Query Parameter: scopeq set to '/ Madrid/Gardens/ParqueNorte,/CompanyA/OrganizationB/UnitC' and	
	URL set to '/ ngsi-ld/v1/entities/' method set to 'GET' Get Entities Request: and Query Parameter: scopeq set to '/ Madrid/Gardens/ParqueNorte,/CompanyA/OrganizationB/UnitC' and Query Parameter: entity_types set to 'https://	
	URL set to '/ ngsi-Id/v1/entities/' method set to 'GET' Get Entities Request: and Query Parameter: scopeq set to '/ Madrid/Gardens/ParqueNorte,/CompanyA/OrganizationB/UnitC' and	
	URL set to '/ ngsi-ld/v1/entities/' method set to 'GET' Get Entities Request: and Query Parameter: scopeq set to '/ Madrid/Gardens/ParqueNorte,/CompanyA/OrganizationB/UnitC' and Query Parameter: entity_types set to 'https:// ngsi-ld-test-suite/context#Building' }	CLIT - Client
	URL set to '/ ngsi-Id/v1/entities/' method set to 'GET' Get Entities Request: and Query Parameter: scopeq set to '/ Madrid/Gardens/ParqueNorte,/CompanyA/OrganizationB/UnitC' and Query Parameter: entity_types set to 'https:// ngsi-Id-test-suite/context#Building' } then {	SUT -> Client
	URL set to '/ ngsi-Id/v1/entities/' method set to 'GET' Get Entities Request: and Query Parameter: scopeq set to '/ Madrid/Gardens/ParqueNorte,/CompanyA/OrganizationB/UnitC' and Query Parameter: entity_types set to 'https:// ngsi-Id-test-suite/context#Building' } then { the SUT sends a valid Response for the operations:	SUT -> Client
	URL set to '/ ngsi-Id/v1/entities/' method set to 'GET' Get Entities Request: and Query Parameter: scopeq set to '/ Madrid/Gardens/ParqueNorte,/CompanyA/OrganizationB/UnitC' and Query Parameter: entity_types set to 'https:// ngsi-Id-test-suite/context#Building' } then { the SUT sends a valid Response for the operations: Query Entities with Response Status Code set to	SUT -> Client
	URL set to '/ ngsi-Id/v1/entities/' method set to 'GET' Get Entities Request: and Query Parameter: scopeq set to '/ Madrid/Gardens/ParqueNorte,/CompanyA/OrganizationB/UnitC' and Query Parameter: entity_types set to 'https:// ngsi-Id-test-suite/context#Building' } then { the SUT sends a valid Response for the operations: Query Entities with Response Status Code set to	SUT -> Client
	URL set to '/ ngsi-Id/v1/entities/' method set to 'GET' Get Entities Request: and Query Parameter: scopeq set to '/ Madrid/Gardens/ParqueNorte,/CompanyA/OrganizationB/UnitC' and Query Parameter: entity_types set to 'https:// ngsi-Id-test-suite/context#Building' } then { the SUT sends a valid Response for the operations: Query Entities with Response Status Code set to and Query Entities with Response Body containing a list of entities equal to	SUT -> Client
	URL set to '/ ngsi-Id/v1/entities/' method set to 'GET' Get Entities Request: and Query Parameter: scopeq set to '/ Madrid/Gardens/ParqueNorte,/CompanyA/OrganizationB/UnitC' and Query Parameter: entity_types set to 'https:// ngsi-Id-test-suite/context#Building' } then { the SUT sends a valid Response for the operations: Query Entities with Response Status Code set to and Query Entities with Response Body containing a list of entities equal to '\${2}	SUT -> Client
	URL set to '/ ngsi-ld/v1/entities/' method set to 'GET' Get Entities Request: and Query Parameter: scopeq set to '/ Madrid/Gardens/ParqueNorte,/CompanyA/OrganizationB/UnitC' and Query Parameter: entity_types set to 'https:// ngsi-ld-test-suite/context#Building' } then { the SUT sends a valid Response for the operations: Query Entities with Response Status Code set to and Query Entities with Response Body containing a list of entities equal to '\${2} ' of type 'https://ngsi-ld-test-suite/	SUT -> Client
	URL set to '/ ngsi-ld/v1/entities/' method set to 'GET' Get Entities Request: and Query Parameter: scopeq set to '/ Madrid/Gardens/ParqueNorte,/CompanyA/OrganizationB/UnitC' and Query Parameter: entity_types set to 'https:// ngsi-ld-test-suite/context#Building' } then { the SUT sends a valid Response for the operations: Query Entities with Response Status Code set to and Query Entities with Response Body containing a list of entities equal to '\${2} ' of type 'https://ngsi-ld-test-suite/ context#Building' with response set to '\$	SUT -> Client
	URL set to '/ ngsi-ld/v1/entities/' method set to 'GET' Get Entities Request: and Query Parameter: scopeq set to '/ Madrid/Gardens/ParqueNorte,/CompanyA/OrganizationB/UnitC' and Query Parameter: entity_types set to 'https:// ngsi-ld-test-suite/context#Building' } then { the SUT sends a valid Response for the operations: Query Entities with Response Status Code set to and Query Entities with Response Body containing a list of entities equal to '\${2} ' of type 'https://ngsi-ld-test-suite/	SUT -> Client

TP ld	TP/NGSI-LD/CI/Cons/E/019_01_04	
Test objective	Check that one can query several entities based on attribute names	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.2	
Config Id	CF_01	
PICS Selection	PICS_QueEnt_AND_QueEntPos	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/E/019_01_04	
Name	019_01_04 Query several entities based on attribute names	
Parent Release	v1.3.1	
Description	Check that one can query several entities based on attribute names	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing.	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ngsi-ld/v1/entities/' method set to 'GET'	
	Get Entities Request: and	
	Get Entitles Nequest, and	

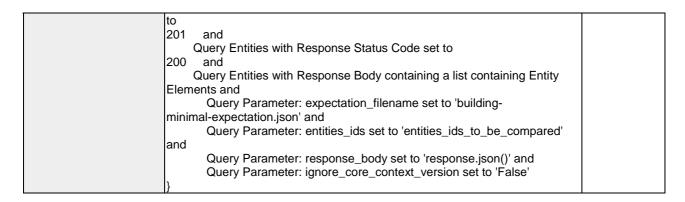
	Query Parameter: attrs set to '\${attributes_to_be_retrieved}'	
}	1	OUT Oliver
	en {	SUT -> Client
	the SUT sends a valid Response for the operations:	
to	Create Entity Selecting Content Type with Response Status Code set	
to 20		
20	01 and Create Entity Selecting Content Type with Response Status Code set	
to		
to 20		
20	Query Entities with Response Status Code set to	
20	· ·	
20	Query Entities with Response Body containing a list containing Entity	
Ele	ements and	
	Query Parameter: expectation_filename set to 'building-	
att	tributes-query-expectation.json' and	
	Query Parameter: entities_ids set to 'entities_ids_to_be_compared'	
an	nd	
	Query Parameter: ignore_core_context_version set to 'False'	
}		

TP Id	TP/NGSI-LD/CI/Cons/E/019_06	
Test objective	Check that one can query entities specifying a maximum number of results	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 6.3.10	
Config Id	CF_01	
PICS Selection	PICS_HTTP_PAGINATION_LIMIT	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/E/019_06_01	
Name	019_06_01 Query entities specifying a maximum number of results	
Parent Release	v1.3.1	
Description	Check that one can query entities specifying a maximum number of results	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/entities/' method set to 'GET' Get Entities Request: and Query Parameter: entity_ids set to '\${entities_ids_to_be_retrieved}' and Query Parameter: entity_types set to '\${entity_types_to_be_retrieved}' and Query Parameter: limit set to '2' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Create Entity Selecting Content Type with Response Status Code set to 201 and Create Entity Selecting Content Type with Response Status Code set to 201 and Create Entity Selecting Content Type with Response Status Code set to 201 and Create Entity Selecting Content Type with Response Status Code set to 201 and Query Entities with Response Status Code set to 200 and Query Entities with Response Body containing a list of entities equal to '\${2} ' of type 'https://ngsi-ld-test-suite/ context#Building' with response set to '\$ {response.json()}' }	SUT -> Client

TP Id	TP/NGSI-LD/CI/Cons/E/019_01_03
Test objective	Check that one can query several entities based on the given id pattern
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.2

Config Id	CF_01	
PICS Selection	PICS_QueEnt_AND_QueEntPos	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/E/019_01_03	
Name	019_01_03 Query several entities based on the given id pattern	
Parent Release	v1.3.1	
Description	Check that one can query several entities based on the given id pattern	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/entities/' method set to 'GET' Get Entities Request: and Query Parameter: entity_id_pattern set to 'urn:ngsi-ld:Building:.*' and Query Parameter: entity_types set to '\${entity_types_to_be_retrieved}' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Create Entity Selecting Content Type with Response Status Code set to 201 and Create Entity Selecting Content Type with Response Status Code set to 201 and Query Entities with Response Status Code set to 200 and Query Entities with Response Body containing a list containing Entity Elements and Query Parameter: expectation_filename set to 'building- minimal-expectation.json' and Query Parameter: entities_ids set to 'entities_ids_to_be_compared' and Query Parameter: response_body set to 'response.json()' and Query Parameter: ignore_core_context_version set to 'False' }	SUT -> Client

TP ld	TP/NGSI-LD/CI/Cons/E/019_01_01	
Test objective	Check that one can query several entities based on ids	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.2	
Config Id	CF_01	
PICS Selection	PICS_QueEnt_AND_QueEntPos	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/E/019_01_01	
Name	019_01_01 Query several entities based on ids	
Parent Release	v1.3.1	
Description	Check that one can query several entities based on ids	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/entities/' method set to 'GET' Get Entities Request: and Query Parameter: entity_ids set to '\${entities_ids_to_be_retrieved}' and Query Parameter: entity_types set to '\${entity_types_to_be_retrieved}' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Create Entity Selecting Content Type with Response Status Code set to 201 and Create Entity Selecting Content Type with Response Status Code set to 201 and Create Entity Selecting Content Type with Response Status Code set to 201 and Create Entity Selecting Content Type with Response Status Code set	SUT -> Client



TP ld	TP/NGSI-LD/CI/Cons/E/019_01_05	_
Test objective	Check that one can query entities based on a geoquery	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.2	
Config Id	CF_01	
PICS Selection	PICS_QueEnt_AND_QueEntPos	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/E/019_01_05	
Name	019_01_05 Query several entities based on a geoquery	
Parent Release	v1.3.1	
Description	Check that one can query entities based on a geoquery	
Expected behaviour	Test events	Direction SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/entities/' method set to 'GET' Get Entities Request: and Query Parameter: entity_types set to '\${entity_types_to_be_retrieved}' and Query Parameter: accept set to 'application/ld+json' and Query Parameter: georel set to 'equals' and Query Parameter: geometry set to 'Point' and Query Parameter: coordinates set to '[13.3986, 52.5547]' }	
	then { the SUT sends a valid Response for the operations: Create Entity Selecting Content Type with Response Status Code set to 201 and Create Entity Selecting Content Type with Response Status Code set to 201 and Query Entities with Response Status Code set to 200 and Query Entities with Response Body containing a list containing Entity Elements and Query Parameter: expectation_filename set to 'building- geoproperty-query-expectation.jsonld' and Query Parameter: entities_ids set to 'entities_ids_to_be_compared' and Query Parameter: ignore_core_context_version set to 'False' }	SUT -> Client

TP Id	TP/NGSI-LD/CI/Cons/E/019_07
	If the count parameter is set to true the special HTTP header NGSILD-Results-Count is set in the response and it has to contain the total number of matching results.
Reference	ETSI GS CIM 009 V1.5.1 [1], clauses 5.7.2, 6.3.13
Config Id	CF_01
PICS Selection	PICS_QueEnt_AND_QueEntPos and PICS_HTTP_COUNT_RESULTS
Initial conditions	with { the SUT containing an initial state }

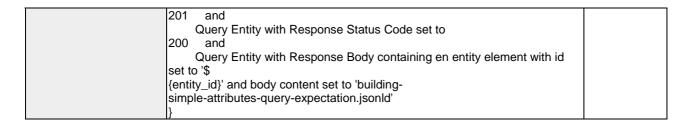
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/E/019_07_01	
Name	019_07_01 Check that the total number of matching results is returned if the	ne count
D	parameter is set to true and only the entity type is provided	
Parent Release	V1.3.1	
Description	If the count parameter is set to true the special HTTP header NGSILD-Results-Count is in the response and it has to contain the total number of matching results.	
Expected behaviour	Test events	Direction
Expedica bellavioui	when {	SUT <- Client
	the SUT receives a Request from the client containing:	OOT < Onem
	URL set to /	
	ngsi-ld/v1/entities/	
	method set to 'GET'	
	Get Entities Request: and	
	Query Parameter: entity_types set to 'Building' and	
	Query Parameter: q set to '\$	
	{q}' and	
	Query Parameter: count set to '\$	
	{True}' and	
	Query Parameter: Link set to '<	
	\$https://forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/	
	resources/jsonld-contexts/ngsi-ld-test-suite-	
	compound.jsonld>; rel="http://www.w3.org/ns/json-	
	ld#context";type="application/ld+json"	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Query Entities with Response Status Code set to	
	\${expected_status_code} and	
	Query Entities with Response Header: NGSILD- Results-Count equals to 2	
	Results-Court equals to 2	
TP Permutation id	TP/NGSI-LD/CI/Cons/E/019_07_02	
Name	019_07_02 Check that the total number of matching results is returned if the	he count
	parameter is set to true and a q parameter is provided	
Parent Release	v1.3.1	
Description	If the count parameter is set to true the special HTTP header NGSILD-Res	sults-Count is set
	in the response and it has to contain the total number of matching results.	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/'	
	method set to 'GET' Get Entities Request: and	
	Query Parameter: entity_types set to 'Building' and	
	Query Parameter: q set to '\$	
	(q)' and	
	Query Parameter: count set to '\$	
	{True}' and	
	Query Parameter: Link set to '<	
	\$https://forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/	
	resources/jsonId-contexts/ngsi-Id-test-suite-	
	compound.jsonld>; rel="http://www.w3.org/ns/json-	
	ld#context";type="application/ld+json"	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Query Entities with Response Status Code set to	
	\${expected_status_code} and	
	Query Entities with Response Header: NGSILD-	
	Results-Count equals to 1	
	}	

TP ld	TP/NGSI-LD/CI/Cons/E/019_05
Test objective	Check that the queried entities by id can be returned in GeoJSON format
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 6.3.7

Config Id	CF_01	
PICS Selection	PICS_HTTP_SIMP_REPR	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/E/019_05_01	
Name	019_05_01 Get an entity by id that can be returned in GeoJSON format	
Parent Release	v1.3.1	
Description	Check that the queried entities by id can be returned in GeoJSON format	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/' method set to 'GET' Get Entities Request: and Query Parameter: entity_ids set to '\$ {entities_ids_to_be_retrieved}' and Query Parameter: entity_types set to '\$ {entity_types_to_be_retrieved}' and Query Parameter: accept set to 'application/ geo+json' and Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/ resources/jsonld-contexts/ngsi-ld-test-suite- compound.jsonld>; rel="http://www.w3.org/ns/json- ld#context";type="application/ld+json"' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Query Entities with Response Status Code set to 200 and Query Entities with Check Response Body Content and Query Parameter: expectation_filename set to 'two-buildings-location-attribute-normalized.geojson' and Query Parameter: response_body set to 'response.json()' and Query Parameter: additional_ignored_path set to '\$ {EMPTY}' }	SUT -> Client

4.4.1.3.2 RetrieveEntity

TP Id	TP/NGSI-LD/CI/Cons/E/018_01_02	
Test objective	Check that one can query some attributes from an entity	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.1	
Config Id	CF_01	
PICS Selection	PICS EntRet	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/E/018_01_02	
Name	018_01_02 Query some attributes from an entity	
Parent Release	v1.3.1	
Description	Check that one can query some attributes from an entity	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\${entity_id}' and Query Parameter: accept set to 'application/ld+json' and Query Parameter: attrs set to '\${attributes_to_be_retrieved}' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Create Entity Selecting Content Type with Response Status Code set	SUT -> Client



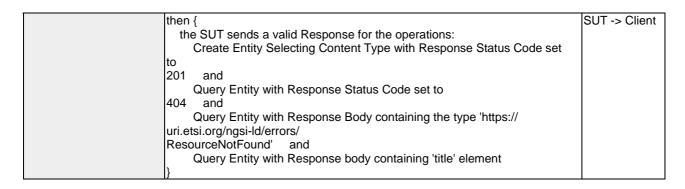
TP Id	TP/NGSI-LD/CI/Cons/E/018_01_01	
Test objective	Check that one can get an entity by id	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.1	
Config Id	CF 01	
PICS Selection	PICS_EntRet	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/E/018_01_01	
Name	018_01_01 Get an entity by id	
Parent Release	v1.3.1	
Description	Check that one can get an entity by id	
Expected behaviour	Test events	Direction SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\${entity_id}' and Query Parameter: accept set to 'application/ld+json' } then { the SUT sends a valid Response for the operations: Create Entity Selecting Content Type with Response Status Code set to 201 and Query Entity with Response Status Code set to 200 and Query Entity with Response Body containing en entity element with id set to '\$	SUT -> Client

TP ld	TP/NGSI-LD/CI/Cons/E/018 01 03	
Test objective	Check that one can query the geometry property from an entity	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.1	
Config Id	CF 01	
PICS Selection	PICS EntRet	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/E/018_01_03	
Name	018_01_03 Query the geometry property from an entity	
Parent Release	v1.3.1	
Description	Check that one can query the geometry property from an entity	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ngsi-ld/v1/entities/{id}'	
	method set to 'GET'	
	Get Entity Request: and	
	Query Parameter: id set to '\${entity_id}' and	
	Query Parameter: accept set to 'application/ld+json' and	
	Query Parameter: geoproperty set to 'location'	
	In the second se	

then { the SUT sends a valid Response for the operations: Create Entity Selecting Content Type with Response Status Code set to 201 and Charle Entity with Response Status Code set to	SUT -> Client
Query Entity with Response Status Code set to 200 and Query Entity with Response Body containing en entity element with id set to '\$ {entity_id}' and body content set to 'building- geoproperty-query-expectation.jsonId' }	

TP ld	TP/NGSI-LD/CI/Cons/E/018_02	
Test objective	Check that one cannot get an entity with invalid id	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.1	
Config Id	CF_01	
PICS Selection	PICS_EntRet	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/E/018_02_01	
Name	018_02_01 Get an entity if the Entity Id is not a valid URI	
Parent Release	v1.3.1	
Description	Check that one cannot get an entity with invalid/missing id	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ngsi-ld/v1/entities/{id}'	
	method set to 'GET'	
	Get Entity Request: and	
	Query Parameter: id set to 'thisisaninvaliduri' and	
	Query Parameter: accept set to 'application/ld+json'	
]	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Query Entity with Response Status Code set to	
	400 and	
	Query Entity with Response Body containing the type 'https://	
	uri.etsi.org/ngsi-ld/errors/	
	BadRequestData' and	
	Query Entity with Response body containing 'title' element	
	}	

T		
TP ld	TP/NGSI-LD/CI/Cons/E/018_03_02	
Test objective	Check that one cannot get an entity if an attribute is not known to the system	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.1	
Config Id	CF_01	
PICS Selection	PICS_EntRet	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/E/018_03_02	
Name	018_03_02 Get an entity if an attribute is not known to the system	
Parent Release	v1.3.1	
Description	Check that one cannot get an entity if an attribute is not known to the system	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ngsi-ld/v1/entities/{id}'	
	method set to 'GET'	
	Get Entity Request: and	
	Query Parameter: id set to '\${entity_id}' and	
	Query Parameter: accept set to 'application/ld+json' and	
	Query Parameter: attrs set to '\${attributes_to_be_retrieved}'	



TP Id	TP/NGSI-LD/CI/Cons/E/018_04	
Test objective	Check that the queried entity by Id can be returned in a simplified representati	on
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 6.3.7	
Config Id	CF_01	
PICS Selection	PICS_HTTP_SIMP_REPR	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/E/018_04_01	
Name	018_04_01 Get an entity in a simplified representation	
Parent Release	v1.3.1	
Description	Check that the queried entity by Id can be returned in a simplified representati	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\${entity_id}' and Query Parameter: accept set to 'application/ld+json' and Query Parameter: options set to 'keyValues' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Create Entity Selecting Content Type with Response Status Code set to 201 and Query Entity with Response Status Code set to 200 and Query Entity with Response Body containing en entity element with id set to '\$ {entity_id}' and body content set to 'building- simple-attributes-simplified-expectation.jsonld'}	SUT -> Client

TP Id	TP/NGSI-LD/CI/Cons/E/018_06
Test objective	Check that the JSON-LD @context is obtained from a Link header if present and that the
	default JSON-LD @context is used if not present
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 6.3.5
Config Id	CF_01
PICS Selection	PICS_HTTP_CONTEXT_RESOLV
Initial conditions	with { the SUT being in the "initial state" and the SUT containing an initial Entity \${entity} with
	an id set to \${entityId} }
	Permutations
TP Permutation id	TP/NGSI-LD/CI/Cons/E/018_06_01
Name	018_06_01 EmptyJsonLdContext
Parent Release	v1.3.1
Description	Check that the JSON-LD @context is obtained from a Link header if present and that the
	default JSON-LD @context is used if not present

Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/{id}'	
	method set to 'GET'	
	Get Entity Request: and	
	Query Parameter: id set to '\$	
	{entity_id}' and	
	Query Parameter: accept set to 'application/	
	json' and	
	Query Parameter: Link set to '<	
	\$\${EMPTY}>; rel="http://www.w3.org/ns/json-	
	ld#context";type="application/ld+json"	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Query Entity with Response Status Code set to	
	200 and	
	Query Entity with Response Body containing en entity element with id	
	set to '\$	
	{entity_id}' and body content set to 'building-	
	simple-attributes-sample-expanded-expectation.json'	
	}	
TP Permutation id	TP/NGSI-LD/CI/Cons/E/018_06_02	
Name	018_06_02 CreationTimeJsonLdContext	
Parent Release	v1.3.1	
Description	Check that the JSON-LD @context is obtained from a Link header if present a	and that the
	default JSON-LD @context is used if not present	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/{id}'	
	method set to 'GET'	
	Get Entity Request: and	
	Query Parameter: id set to '\$	
	{entity_id}' and	
	Query Parameter: accept set to 'application/	
	json' and	
	Query Parameter: Link set to '<	
	\$https://forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/	
	resources/jsonld-contexts/ngsi-ld-test-suite-	
	compound.jsonld>; rel="http://www.w3.org/ns/json-	
	ld#context";type="application/ld+json"	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Query Entity with Response Status Code set to	
	200 and	
	Query Entity with Response Body containing en entity element with id	
	set to '\$	
	{entity_id}' and body content set to 'building-	
	simple-attributes-sample-compacted-expectation.json'	
]}	

TP Id	TP/NGSI-LD/CI/Cons/E/018_03_01
Test objective	Check that one cannot get an entity if the entity id is not known to the system
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.7.1
Config Id	CF_01
PICS Selection	PICS_EntRet
Initial conditions	with { the SUT containing an initial state }
Permutations Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/E/018_03_01
Name	018_03_01 Get an entity if the Entity Id is not known to the system
Parent Release	v1.3.1

Description	Check that one cannot get an entity if the entity id or attributes are not known to the system	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ngsi-ld/v1/entities/{id}'	
	method set to 'GET'	
	Get Entity Request: and	
	Query Parameter: id set to '\${entity_id}' and	
	Query Parameter: accept set to 'application/ld+json'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Query Entity with Response Status Code set to	
	404 and	
	Query Entity with Response Body containing the type 'https://	
	uri.etsi.org/ngsi-ld/errors/	
	ResourceNotFound' and	
	Query Entity with Response body containing 'title' element	
	}	

TP Id	TP/NGSI-LD/CI/Cons/E/018_07	
Test objective	Check that one can retrieve an entity using Language Filter	
Reference	ETSI GS CIM 009 V1.4.1 [1], clauses 4.15 and 5.7.1	
Config Id	CF_01	
PICS Selection	PICS_LANGFILTER and PICS_EntRet	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/E/018_07_01	
Name	018_07_01 Retrieve an entity using a specific natural language	
Parent Release	v1.4.1	
Description	Check that one can retrieve an entity using Language Filter	
Expected behaviour	Test events	Direction
•	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/{id}'	
	method set to 'GET'	
	Get Entity Request: and	
	Query Parameter: id set to '\$	
	{entity_id}' and	
	Query Parameter: accept set to 'application/	
	ld+json' and	
	Query Parameter: Link set to '<	
	\$https://forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/	
	resources/jsonld-contexts/ngsi-ld-test-suite-	
	compound.jsonld>; rel="http://www.w3.org/ns/json-	
	ld#context";type="application/ld+json" and	
	Query Parameter: options set to '\$	
	{EMPTY}' and	
	Query Parameter: lang set to 'fr'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Query Entity with Response Status Code set to	
	200 and	
	Query Entity with Response Body containing en entity element with id	
	set to '\$ (notify id)' and back content act to 'building	
	{entity_id}' and body content set to 'building- language-property-fr-filter.jsonId'	
	nanguage-property-ir-iliter.jsoniu Ir	
TP Permutation id	TP/NGSI-LD/CI/Cons/E/018_07_02	l
Name	018_07_02 Retrieve an entity using multiple natural languages with no ranked	proforonco
Parent Release		preference
	V1.4.1	
Description	Check that one can retrieve an entity using Language Filter	

Expected behaviour	Test events	Direction
Expedica Bellavious	when {	SUT <- Client
	the SUT receives a Request from the client containing:	COT CHOIL
	URL set to '/	
	ngsi-ld/v1/entities/{id}'	
	method set to 'GET'	
	Get Entity Request: and	
	Query Parameter: id set to '\$	
	{entity_id}' and	
	Query Parameter: accept set to 'application/	
	ld+json' and	
	Query Parameter: Link set to '<	
	\$https://forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/	
	resources/jsonId-contexts/ngsi-Id-test-suite-	
	compound.jsonld>; rel="http://www.w3.org/ns/json-	
	Id#context";type="application/Id+json" and	
	Query Parameter: options set to '\$	
	{EMPTY}' and	
	Query Parameter: lang set to 'fr-	
	CH,fr'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Query Entity with Response Status Code set to	
	200 and	
	Query Entity with Response Body containing en entity element with id	
	set to '\$	
	{entity_id}' and body content set to 'building-	
	language-property-fr-filter.jsonld'	
	}	
TP Permutation id	TP/NGSI-LD/CI/Cons/E/018_07_03	
Name	018_07_03 Retrieve an entity with any supported language	
Parent Release	v1.4.1	
Description	Check that one can retrieve an entity using Language Filter	
Expected behaviour	Test events	Direction
-	Test events when {	Direction SUT <- Client
-	Test events when { the SUT receives a Request from the client containing:	
-	Test events when { the SUT receives a Request from the client containing: URL set to '/	
-	Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/{id}'	
-	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entities/{id}' method set to 'GET'	
-	Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entities/{id}' method set to 'GET' Get Entity Request: and	
-	Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$	
-	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and	
-	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/	
-	Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ Id+json' and	
-	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ Id+json' and Query Parameter: Link set to '<	
-	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ Id+json' and Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/	
-	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ Id+json' and Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/ resources/jsonId-contexts/ngsi-Id-test-suite-	
-	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ Id+json' and Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/ resources/jsonId-contexts/ngsi-Id-test-suite- compound.jsonId>; rel="http://www.w3.org/ns/json-	
-	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ Id+json' and Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/ resources/jsonId-contexts/ngsi-Id-test-suite- compound.jsonId>; rel="http://www.w3.org/ns/json- Id#context";type="application/Id+json" and	
-	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ Id+json' and Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/ resources/jsonId-contexts/ngsi-Id-test-suite- compound.jsonId>; rel="http://www.w3.org/ns/json- Id#context";type="application/Id+json" and Query Parameter: options set to '\$	
-	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ Id+json' and Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/ resources/jsonld-contexts/ngsi-ld-test-suite- compound.jsonld>; rel="http://www.w3.org/ns/json- Id#context";type="application/Id+json" and Query Parameter: options set to '\$ {EMPTY}' and	
-	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ Id+json' and Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/ resources/jsonId-contexts/ngsi-Id-test-suite- compound.jsonId>; rel="http://www.w3.org/ns/json- Id#context";type="application/Id+json" and Query Parameter: options set to '\$	
-	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ Id+json' and Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/ resources/jsonId-contexts/ngsi-Id-test-suite- compound.jsonId>; rel="http://www.w3.org/ns/json- Id#context";type="application/Id+json" and Query Parameter: options set to '\$ {EMPTY}' and Query Parameter: lang set to '*' }	SUT <- Client
-	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ Id+json' and Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/ resources/jsonId-contexts/ngsi-Id-test-suite- compound.jsonId>; rel="http://www.w3.org/ns/json- Id#context";type="application/Id+json"' and Query Parameter: options set to '\$ {EMPTY}' and Query Parameter: lang set to '*' } then {	
-	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ Id+json' and Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/ resources/jsonId-contexts/ngsi-Id-test-suite- compound.jsonId>; rel="http://www.w3.org/ns/json- Id#context";type="application/Id+json"' and Query Parameter: options set to '\$ {EMPTY}' and Query Parameter: lang set to '*' } then { the SUT sends a valid Response for the operations:	SUT <- Client
-	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ Id+json' and Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/ resources/jsonId-contexts/ngsi-Id-test-suite- compound.jsonId>; rel="http://www.w3.org/ns/json- Id#context";type="application/Id+json"' and Query Parameter: options set to '\$ {EMPTY}' and Query Parameter: lang set to '*' } then { the SUT sends a valid Response for the operations: Query Entity with Response Status Code set to	SUT <- Client
-	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ Id+json' and Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/ resources/jsonId-contexts/ngsi-Id-test-suite- compound.jsonId>; rel="http://www.w3.org/ns/json- Id#context";type="application/Id+json" and	SUT <- Client
-	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ Id+json' and Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/ resources/jsonId-contexts/ngsi-Id-test-suite- compound.jsonId>; rel="http://www.w3.org/ns/json- Id#context";type="application/Id+json" and	SUT <- Client
-	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ Id+json' and Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/ resources/jsonld-contexts/ngsi-ld-test-suite- compound.jsonld>; rel="http://www.w3.org/ns/json- Id#context";type="application/Id+json" and Query Parameter: options set to '\$ {EMPTY}' and Query Parameter: lang set to '*' } then { the SUT sends a valid Response for the operations: Query Entity with Response Body containing en entity element with id set to '\$	SUT <- Client
· -	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ Id+json' and Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/ resources/jsonld-contexts/ngsi-ld-test-suite- compound.jsonld>; rel="http://www.w3.org/ns/json- Id#context";type="application/Id+json" and Query Parameter: options set to '\$ {EMPTY}' and Query Parameter: lang set to '*' } then { the SUT sends a valid Response for the operations:	SUT <- Client
· -	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ Id+json' and Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/ resources/jsonld-contexts/ngsi-ld-test-suite- compound.jsonld>; rel="http://www.w3.org/ns/json- Id#context";type="application/Id+json" and Query Parameter: options set to '\$ {EMPTY}' and Query Parameter: lang set to '*' } then { the SUT sends a valid Response for the operations: Query Entity with Response Body containing en entity element with id set to '\$	SUT <- Client
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/ resources/jsonld-contexts/ngsi-ld-test-suite- compound.jsonld>; rel="http://www.w3.org/ns/json- ld#context";type="application/ld+json" and Query Parameter: options set to '\$ {EMPTY}' and Query Parameter: lang set to '*' } then { the SUT sends a valid Response for the operations: Query Entity with Response Status Code set to 200 and Query Entity with Response Body containing en entity element with id set to '\$ {entity_id}' and body content set to 'building- language-property-any-language-filter.jsonld' }	SUT <- Client
TP Permutation id	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ Id+json' and Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/ resources/jsonId-contexts/ngsi-Id-test-suite- compound.jsonId>; rel="http://www.w3.org/ns/json- Id#context";type="application/Id+json" and Query Parameter: options set to '\$ {EMPTY}' and Query Parameter: lang set to '*' } then { the SUT sends a valid Response for the operations: Query Entity with Response Status Code set to 200 and Query Entity with Response Body containing en entity element with id set to '\$ {entity_id}' and body content set to 'building- language-property-any-language-filter.jsonId' } TP/NGSI-LD/CI/Cons/E/018_07_04	SUT -> Client
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ Id+json' and Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/ resources/jsonId-contexts/ngsi-Id-test-suite- compound.jsonId>; rel="http://www.w3.org/ns/json- Id#context";type="application/Id+json" and Query Parameter: options set to '\$ {EMPTY}' and Query Parameter: lang set to '*' } then { the SUT sends a valid Response for the operations: Query Entity with Response Status Code set to 200 and Query Entity with Response Body containing en entity element with id set to '\$ {entity_id}' and body content set to 'building- language-property-any-language-filter.jsonId' } TP/NGSI-LD/CI/Cons/E/018_07_04 018_07_04 Retrieve an entity using multiple natural languages with ranked price of the property of the proper	SUT -> Client
TP Permutation id Name	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ Id+json' and Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/ resources/jsonId-contexts/ngsi-Id-test-suite- compound.jsonId>; rel="http://www.w3.org/ns/json- Id#context";type="application/Id+json" and Query Parameter: options set to '\$ {EMPTY}' and Query Parameter: lang set to '*' } then { the SUT sends a valid Response for the operations: Query Entity with Response Status Code set to 200 and Query Entity with Response Body containing en entity element with id set to '\$ {entity_id}' and body content set to 'building- language-property-any-language-filter.jsonId' } TP/NGSI-LD/CI/Cons/E/018_07_04	SUT -> Client

Expected behaviour	Test events	Direction
Exposica benavious	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/{id}'	
	method set to 'GET'	
	Get Entity Request: and	
	Query Parameter: id set to '\$	
	{entity_id}' and	
	Query Parameter: accept set to 'application/	
	Id+json' and	
	Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/	
	resources/jsonId-contexts/ngsi-Id-test-suite-	
	compound.jsonld>; rel="http://www.w3.org/ns/json-	
	Id#context";type="application/ld+json" and	
	Query Parameter: options set to '\$	
	{EMPTY}' and	
	Query Parameter: lang set to 'fr-	
	CH,fr;q=0.9,en;q=0.8,*;q=0.5'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Query Entity with Response Status Code set to	
	200 and	
	Query Entity with Response Body containing en entity element with id	
	set to '\$ {entity_id}' and body content set to 'building-	
	language-property-fr-filter.jsonld	
	language-property-n-inter.jsoniu	
TP Permutation id	TP/NGSI-LD/CI/Cons/E/018_07_05	1
Name	018_07_05 Retrieve an entity using a specific natural language with simplified	<u> </u>
	representation	
Parent Release	v1.4.1	
Description	Check that one can retrieve an entity using Language Filter	T
	Check that one can retrieve an entity using Language Filter Test events	Direction
Description	Check that one can retrieve an entity using Language Filter Test events when {	Direction SUT <- Client
Description	Check that one can retrieve an entity using Language Filter Test events when { the SUT receives a Request from the client containing:	
Description	Check that one can retrieve an entity using Language Filter Test events when { the SUT receives a Request from the client containing: URL set to '/	
Description	Check that one can retrieve an entity using Language Filter Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/{id}'	
Description	Check that one can retrieve an entity using Language Filter Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/{id}' method set to 'GET'	
Description	Check that one can retrieve an entity using Language Filter Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entities/{id}' method set to 'GET' Get Entity Request: and	
Description	Check that one can retrieve an entity using Language Filter Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$	
Description	Check that one can retrieve an entity using Language Filter Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entities/{id}' method set to 'GET' Get Entity Request: and	
Description	Check that one can retrieve an entity using Language Filter Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and	
Description	Check that one can retrieve an entity using Language Filter Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ Id+json' and Query Parameter: Link set to '<	
Description	Check that one can retrieve an entity using Language Filter Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ Id+json' and Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/	
Description	Check that one can retrieve an entity using Language Filter Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ Id+json' and Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/ resources/jsonld-contexts/ngsi-ld-test-suite-	
Description	Check that one can retrieve an entity using Language Filter Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ Id+json' and Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/ resources/jsonld-contexts/ngsi-ld-test-suite- compound.jsonld>; rel="http://www.w3.org/ns/json-	
Description	Check that one can retrieve an entity using Language Filter Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ Id+json' and Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/ resources/jsonld-contexts/ngsi-ld-test-suite- compound.jsonld>; rel="http://www.w3.org/ns/json- Id#context";type="application/ld+json" and	
Description	Check that one can retrieve an entity using Language Filter Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ Id+json' and Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/ resources/jsonld-contexts/ngsi-ld-test-suite- compound.jsonld>; rel="http://www.w3.org/ns/json- Id#context";type="application/ld+json" and Query Parameter: options set to 'keyValues' and	
Description	Check that one can retrieve an entity using Language Filter Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ Id+json' and Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/ resources/jsonld-contexts/ngsi-ld-test-suite- compound.jsonld>; rel="http://www.w3.org/ns/json- Id#context";type="application/ld+json" and	
Description	Check that one can retrieve an entity using Language Filter Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ Id+json' and Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/ resources/jsonld-contexts/ngsi-ld-test-suite- compound.jsonld>; rel="http://www.w3.org/ns/json- Id#context";type="application/Id+json" and Query Parameter: options set to 'keyValues' and Query Parameter: lang set to 'fr' }	SUT <- Client
Description	Check that one can retrieve an entity using Language Filter Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ ld+json' and Query Parameter: Link set to '<	
Description	Check that one can retrieve an entity using Language Filter Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ Id+json' and Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/ resources/jsonld-contexts/ngsi-ld-test-suite- compound.jsonld>; rel="http://www.w3.org/ns/json- Id#context";type="application/Id+json" and	SUT <- Client
Description	Check that one can retrieve an entity using Language Filter Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ ld+json' and Query Parameter: Link set to '<	SUT <- Client
Description	Check that one can retrieve an entity using Language Filter Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ Id+json' and Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/ resources/jsonId-contexts/ngsi-Id-test-suite- compound.jsonId>; rel="http://www.w3.org/ns/json- Id#context";type="application/Id+json"' and Query Parameter: options set to 'keyValues' and Query Parameter: lang set to 'fr' } then { the SUT sends a valid Response for the operations: Query Entity with Response Status Code set to 200 and	SUT <- Client
Description	Check that one can retrieve an entity using Language Filter Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ Id+json' and Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/ resources/jsonId-contexts/ngsi-Id-test-suite- compound.jsonId>; rel="http://www.w3.org/ns/json- Id#context";type="application/Id+json"' and Query Parameter: options set to 'keyValues' and Query Parameter: lang set to 'fr' } then { the SUT sends a valid Response for the operations: Query Entity with Response Status Code set to	SUT <- Client
Description	Check that one can retrieve an entity using Language Filter Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ Id+json' and Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/ resources/jsonId-contexts/ngsi-Id-test-suite- compound.jsonId>; rel="http://www.w3.org/ns/json- Id#context";type="application/Id+json" and Query Parameter: options set to 'keyValues' and Query Parameter: lang set to 'fr' } then { the SUT sends a valid Response for the operations: Query Entity with Response Status Code set to and Query Entity with Response Body containing en entity element with id set to '\$ {entity_id}' and body content set to 'building-	SUT <- Client
Description	Check that one can retrieve an entity using Language Filter Test events When { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ Id+json' and Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/ resources/jsonId-contexts/ngsi-Id-test-suite-/raw/develop/ resources/jsonId-conte	SUT <- Client
Description Expected behaviour	Check that one can retrieve an entity using Language Filter Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ Id+json' and Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/ resources/jsonId-contexts/ngsi-Id-test-suite/-/raw/develop/ resources/jsonId-contexts/ngsi-Id-test-suite/-/raw/	SUT <- Client
Expected behaviour TP Permutation id	Check that one can retrieve an entity using Language Filter Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ Id+json' and Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/ resources/jsonId-contexts/ngsi-Id-test-suite- compound.jsonId>; rel="http://www.w3.org/ns/json- Id#context";type="application/Id+json"' and Query Parameter: options set to 'keyValues' and Query Parameter: lang set to 'fr' } then { the SUT sends a valid Response for the operations: Query Entity with Response Status Code set to 200 and Query Entity with Response Body containing en entity element with id set to '\$ {entity_id}' and body content set to 'building- language-property-fr-filter-simplified.jsonId' } TP/NGSI-LD/CI/Cons/E/018_07_06	SUT -> Client
Expected behaviour TP Permutation id Name	Check that one can retrieve an entity using Language Filter Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ Id+json' and Query Parameter: Link set to '< *https://forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/ resources/jsonld-contexts/ngsi-ld-test-suite- compound.jsonld>; rel="http://www.w3.org/ns/json- Id#context";type="application/ld+json"' and	SUT -> Client
Expected behaviour TP Permutation id	Check that one can retrieve an entity using Language Filter Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ Id+json' and Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/ resources/jsonId-contexts/ngsi-Id-test-suite- compound.jsonId>; rel="http://www.w3.org/ns/json- Id#context";type="application/Id+json"' and Query Parameter: options set to 'keyValues' and Query Parameter: lang set to 'fr' } then { the SUT sends a valid Response for the operations: Query Entity with Response Status Code set to 200 and Query Entity with Response Body containing en entity element with id set to '\$ {entity_id}' and body content set to 'building- language-property-fr-filter-simplified.jsonId' } TP/NGSI-LD/CI/Cons/E/018_07_06	SUT -> Client

Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/{id}'	
	method set to 'GET'	
	Get Entity Request: and	
	Query Parameter: id set to '\$	
	{entity_id}' and	
	Query Parameter: accept set to 'application/	
	ld+json' and	
	Query Parameter: Link set to '<	
	\$https://forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/	
	resources/jsonld-contexts/ngsi-ld-test-suite-	
	compound.jsonld>; rel="http://www.w3.org/ns/json-	
	ld#context";type="application/ld+json" and	
	Query Parameter: options set to 'keyValues' and	
	Query Parameter: lang set to '*'	
	lth and C	OUT Officer
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Query Entity with Response Status Code set to	
	200 and	
	Query Entity with Response Body containing en entity element with id	
	set to '\$ (ontity, id)' and hady content set to 'building.	
	{entity_id}' and body content set to 'building- language-property-any-language-filter-simplified.jsonId'	
	hanguage-property-any-language-inter-simplineu.jsoniu	
]}	<u> </u>

TP ld	TP/NGSI-LD/CI/Cons/E/018_05	
Test objective	Check that the queried entity by id can be returned in a GeoJSON format	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 6.3.7	
Config Id	CF_01	
PICS Selection	PICS_HTTP_SIMP_REPR	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Cons/E/018_05_01	
Name	018_05_01 Simplified	
Parent Release	v1.3.1	
Description	Check that the queried entity by id can be returned in a GeoJSON format	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/{id}'	
	method set to 'GET'	
	Get Entity Request: and	
	Query Parameter: id set to '\$	
	{entity_id}' and	
	Query Parameter: accept set to 'application/	
	geo+json' and	
	Query Parameter: options set to 'keyValues' and	
	Query Parameter: Link set to '<	
	\$https://forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/	
	resources/jsonld-contexts/ngsi-ld-test-suite-	
	compound.jsonId>; rel="http://www.w3.org/ns/json- Id#context";type="application/Id+json"	
	lu#context ,type= application/lu+json	
	than (SUT -> Client
	then { the SUT sends a valid Response for the operations:	301 -> Client
	Query Entity with Response Status Code set to	
	200 and	
	Query Entity with Response Body containing en entity element with id	
	set to '\$	
	{entity_id}' and body content set to 'building-	
	location-attribute-simplified.geojson'	
	}	
	U	

TP Permutation id	TP/NGSI-LD/CI/Cons/E/018_05_02	
Name	018_05_02 Normalized	
Parent Release	v1.3.1	
Description	Check that the queried entity by id can be returned in a GeoJSON format	
Expected behaviour	Test events	Direction
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entities/{id}' method set to 'GET' Get Entity Request: and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ geo+json' and Query Parameter: options set to '\$ {EMPTY}' and Query Parameter: Link set to '< \$https://forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/ resources/jsonId-contexts/ngsi-Id-test-suite- compound.jsonId>; rel="http://www.w3.org/ns/json- Id#context";type="application/Id+json"' } then { the SUT sends a valid Response for the operations: Query Entity with Response Status Code set to 200 and Query Entity with Response Body containing en entity element with id set to '\$ {entity_id}' and body content set to 'building- location-attribute-normalized.geojson'	SUT -> Client

4.4.2 Subscription

4.4.2.1 UpdateSubscription

TP Id	TD/NCCL LD/CUCLD/020_04	
	TP/NGSI-LD/CI/SUB/029_01	
Test objective	Check that one cannot update a subscription: If the Subscription id is not pres	ent or it is not
	a valid URI, then an error of type BadRequestData shall be raised	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.8.2	
Config Id	CF_01	
PICS Selection	PICS_SubUpd	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/SUB/029_01_01	
Name	029_01_01 NotPresentId	
Parent Release	v1.3.1	
Description	Check that one cannot update a subscription: If the Subscription id is not pres	ent or it is not
-	a valid URI, then an error of type BadRequestData shall be raised	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ngsi-ld/v1/subscriptions/{subscription_id}'	
	method set to 'PATCH'	
	Request Update Subscription and	
	Header['Content-Type'] set to 'application/json' and	
	Payload defined in file 'subscriptions/fragments/subscription-update-	
	sample.json'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Update Subscription with Response Status Code set to	
	405 and	
	Update Subscription with Response Body containing the type '\$	

	{EMPTY}' and	
	Update Subscription with Response body containing 'title' element	
]	
TP Permutation id	TP/NGSI-LD/CI/SUB/029_01_02	
Name	029_01_02 InvalidId	
Parent Release	v1.3.1	
Description	Check that one cannot update a subscription: If the Subscription id is not prese a valid URI, then an error of type BadRequestData shall be raised	ent or it is not
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ngsi-ld/v1/subscriptions/{subscription_id}'	
	method set to 'PATCH'	
	Request Update Subscription and	
	Header['Content-Type'] set to 'application/json' and	
	Payload defined in file 'subscriptions/fragments/subscription-update-	
	sample.json'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Update Subscription with Response Status Code set to	
	400 and	
	Update Subscription with Response Body containing the type 'https://	
	uri.etsi.org/ngsi-ld/errors/	
	BadRequestData' and	
	Update Subscription with Response body containing 'title' element	
	}	

TP ld	TP/NGSI-LD/CI/SUB/029 03	
Test objective	Check that one cannot update a subscription: If the data types and restriction	are not met by
rest objective	the Subscription Fragment, then an error of type BadRequestData shall be rai	
Reference	ETSI GS CIM 009 V1.5.1 [1]. clause 5.8.2	360
Config Id	CF 01	
PICS Selection	PICS_SubUpd	
Initial conditions	with { the SUT containing an initial state }	
initial conditions	Permutations	
TP Permutation id	TP/NGSI-LD/CI/SUB/029_03_01	
Name	029_03_01 Update Subscription With Invalid Fragment	
Parent Release	v1.3.1	
Description	Check that one cannot update a subscription: If the data types and restriction	are not met by
2000mpo	the Subscription Fragment, then an error of type BadRequestData shall be rai	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ngsi-ld/v1/subscriptions/{subscription_id}'	
	method set to 'PATCH'	
	Request Update Subscription and	
	Header['Content-Type'] set to 'application/json' and	
	Payload defined in file 'subscriptions/fragments/subscription-invalid-update-	
	sample.json'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Update Subscription with Response Status Code set to	
	400 and	
	Update Subscription with Response Body containing the type 'https://	
	uri.etsi.org/ngsi-ld/errors/	
	BadRequestData' and	
	BadRequestData' and Update Subscription with Response body containing 'title' element	

TP Id	TP/NGSI-LD/CI/SUB/029_07
Test objective	Check that one can update a subscription: If isActive is equal to true and expiresAt is not
	present, then status shall be updated to "active", if and only if, the previous value of status
	lwas different than "expired"

Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.8.2	
Config Id	CF_01	
PICS Selection	PICS_SubUpd	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/SUB/029_07_01	
Name	029_07_01 ActiveTrue	
Parent Release	v1.3.1	
Description	Check that one can update a subscription: If isActive is equal to true and expi	
	present, then status shall be updated to "active", if and only if, the previous va	alue of status
	was different than "expired"	_
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ngsi-ld/v1/subscriptions/{subscription_id}'	
	method set to 'PATCH'	
	Request Update Subscription and	
	Header['Content-Type'] set to 'application/json' and Payload defined in file 'subscriptions/fragments/subscription-isActive-true-	
	update-sample.json'	
	apuate-sample.json	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	COT > CHOIN
	Update Subscription with Response Status Code set to	
	204 and	
	Retrieve Subscription with Check Response Body containing an	
	Attribute set to and	
	Query Parameter: expected_attribute_name set to 'status' and	
	Query Parameter: response_body set to 'response1.json()' and	
	Query Parameter: expected_attribute_value set to 'active'	
	}	

TP ld	TP/NGSI-LD/CI/SUB/029_05	
Test objective	Check that one can update a subcription: Term to URI expansion of Attribute r	names shall he
T C St OD JC Stive	observed	larries sriair be
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.8.2	
Config Id	CF_01	
PICS Selection	PICS_SubUpd	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/SUB/029_05_01	
Name	029_05_01 Update Subscription With Term to Uri Expansion with Context	
Parent Release	v1.3.1	
Description	Check that one can update a subcription: Term to URI expansion of Attribute r	names shall be
	observed	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/subscriptions/{subscription_id}'	
	method set to 'PATCH'	
	Request Update Subscription and	
	Header['Link'] contain the context 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	jsonId-contexts/ngsi-Id-test-suite-compound.jsonId' and Header['Content-Type'] set to 'application/json' and	
	Payload defined in file 'subscriptions/fragments/	
	subscription-vehicle-entities-sample.ison'	
	Subscription-verificie-entities-sample.json	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	COT > CHOIR
	Update Subscription with Response Status Code set to	
	204 and	
	Retrieve Subscription with Response Body containing the same content	
	defined in file 'subscriptions/	
	expectations/subscription-vehicle-	

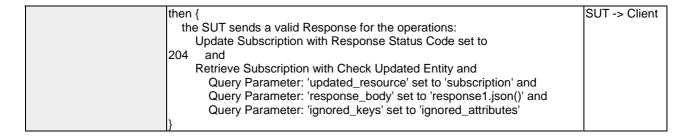
	sample.jsonId' with subscription id '\$	
	{subscription_id}'	
)	
TP Permutation id	TP/NGSI-LD/CI/SUB/029_05_02	
Name	029_05_02 Update Subscription With Term to Uri Expansion without Context	
Parent Release	v1.3.1	
Description	Check that one can update a subcription: Term to URI expansion of Attribute robserved	ames shall be
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/subscriptions/{subscription_id}'	
	method set to 'PATCH'	
	Request Update Subscription and	
	Header['Link'] contain the context 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and	
	Header['Content-Type'] set to 'application/json' and	
	Payload defined in file 'subscriptions/fragments/	
	subscription-vehicle-entities-sample.json'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Update Subscription with Response Status Code set to	
	204 and	
	Retrieve Subscription with Response Body containing the same content	
	defined in file 'subscriptions/	
	expectations/subscription-vehicle-expanded-types-	
	sample.jsonId' with subscription id '\$	
	{subscription_id}'	
]}	

TP ld	TP/NGSI-LD/CI/SUB/029_02	
Test objective	Check that one cannot update a subscription: If the NGSI-LD System does no	
	the target Subscription, because there is no existing Subscription whose id (U	RI) is
	equivalent, an error of type ResourceNotFound shall be raised	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.8.2	
Config Id	CF_01	
PICS Selection	PICS_SubUpd	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/SUB/029_02_01	
Name	029_02_01 Update Unknown Subscription	
Parent Release	v1.3.1	
Description	Check that one cannot update a subscription: If the NGSI-LD System does no	
	the target Subscription, because there is no existing Subscription whose id (U	RI) is
	equivalent, an error of type ResourceNotFound shall be raised	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ngsi-ld/v1/subscriptions/{subscription_id}'	
	method set to 'PATCH'	
	Request Update Subscription and	
	Header['Content-Type'] set to 'application/json' and	
	Payload defined in file 'subscriptions/fragments/subscription-update-	
	sample.json'	
	}	01.17
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Update Subscription with Response Status Code set to	
	404 and	
	Update Subscription with Response Body containing the type 'https://	
	uri.etsi.org/ngsi-ld/errors/	
	ResourceNotFound' and	

Update Subscription with Response body containing 'title' element	
}	

TP Id	TP/NGSI-LD/CI/SUB/029_04	
Test objective	Check that one cannot update a subscription: Any attempt to remove (by settir	ng them to null
	in the Fragment) mandatory properties of a Subscription (clause 5.2.12) shall I	esult in an
	error of type BadRequestData	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.8.2	
Config Id	CF_01	
PICS Selection	PICS_SubUpd	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/SUB/029_04_01	
Name	029_04_01 Update Subscription With Null Mandatory Properties	
Parent Release	v1.3.1	
Description	Check that one cannot update a subscription: Any attempt to remove (by settir	
	in the Fragment) mandatory properties of a Subscription (clause 5.2.12) shall I	result in an
	error of type BadRequestData	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ngsi-ld/v1/subscriptions/{subscription_id}'	
	method set to 'PATCH'	
	Request Update Subscription and	
	Header['Content-Type'] set to 'application/json' and	
	Payload defined in file 'subscriptions/fragments/subscription-null-properties-	
	update-sample.json'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Update Subscription with Response Status Code set to	
	400 and	
	Update Subscription with Response Body containing the type 'https://	
	uri.etsi.org/ngsi-ld/errors/	
	BadRequestData' and	
	Update Subscription with Response body containing 'title' element	
	}	
	17	

TP ld	TP/NGSI-LD/CI/SUB/029_06	
Test objective	Check that one can update a subscription: The implementation shall modify the	e target
	Subscription	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.8.2	
Config Id	CF_01	
PICS Selection	PICS_SubUpd	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/SUB/029_06_01	
Name	029_06_01 Update Subscription	
Parent Release	v1.3.1	
Description	Check that one can update a subscription: The implementation shall modify the	e target
	Subscription	
Expected behaviour	Test events	Direction
Expected behaviour	when {	SUT <- Client
Expected behaviour	when { the SUT receives a Request from the client containing:	
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/	
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/subscriptions/{subscription_id}'	
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/subscriptions/{subscription_id}' method set to 'PATCH'	
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/subscriptions/{subscription_id}' method set to 'PATCH' Request Update Subscription and	
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/subscriptions/{subscription_id}' method set to 'PATCH' Request Update Subscription and Header['Link'] contain the context 'https://	
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/subscriptions/{subscription_id}' method set to 'PATCH' Request Update Subscription and Header['Link'] contain the context 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/	
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/subscriptions/{subscription_id}' method set to 'PATCH' Request Update Subscription and Header['Link'] contain the context 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and	
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/subscriptions/{subscription_id}' method set to 'PATCH' Request Update Subscription and Header['Link'] contain the context 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Header['Content-Type'] set to 'application/json' and	
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/subscriptions/{subscription_id}' method set to 'PATCH' Request Update Subscription and Header['Link'] contain the context 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and	
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/subscriptions/{subscription_id}' method set to 'PATCH' Request Update Subscription and Header['Link'] contain the context 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Header['Content-Type'] set to 'application/json' and	



TP Id	TP/NGSI-LD/CI/SUB/029_08	
Test objective	Check that one can update a subscription: If isActive is equal to true and expir	esAt
	corresponds to a DateTime in the future, then status shall be updated to "active	e"
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.8.2	
Config Id	CF_01	
PICS Selection	PICS_SubUpd	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/SUB/029_08_01	
Name	029_08_01 ActiveTrueExpiresAt	
Parent Release	v1.3.1	
Description	Check that one can update a subscription: If isActive is equal to true and expir	
	corresponds to a DateTime in the future, then status shall be updated to "active	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ngsi-ld/v1/subscriptions/{subscription_id}'	
	method set to 'PATCH'	
	Request Update Subscription and	
	Header['Content-Type'] set to 'application/json' and	
	Payload defined in file 'subscriptions/fragments/subscription-isActive-	
	expiresAt-update-sample.json'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Update Subscription with Response Status Code set to	
	204 and	
	Retrieve Subscription with Check Response Body containing an	
	Attribute set to and	
	Query Parameter: expected_attribute_name set to 'status' and	
	Query Parameter: response_body set to 'response1.json()' and	
	Query Parameter: expected_attribute_value set to 'active'	
	}	

TP ld	TP/NGSI-LD/CI/SUB/029 11	
Test objective	Check that one cannot update a subscription: If expiresAt is included but refe DateTime in the past, then a BadRequestData error shall be raised	rring to a
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.8.2	
Config Id	CF_01	
PICS Selection	PICS_SubUpd	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/SUB/029_11_01	
Name	029_11_01 Update Subscription With ExpiresAt In The Past	
Parent Release	v1.3.1	
Description	Check that one cannot update a subscription: If expiresAt is included but refe DateTime in the past, then a BadRequestData error shall be raised	rring to a
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-Id/v1/subscriptions/{subscription_id}' method set to 'PATCH' Request Update Subscription and Header['Content-Type'] set to 'application/json' and Payload defined in file 'subscriptions/fragments/subscription-expiresAt-	SUT <- Client

update-sample.json'	
then { the SUT sends a valid Response for the operations: Update Subscription with Response Status Code set to 400 and Update Subscription with Response Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/ BadRequestData' and Update Subscription with Response body containing 'title' element }	SUT -> Client

TP ld	TP/NGSI-LD/CI/SUB/029_09	
Test objective	Check that one can update a subscription: If isActive is equal to false and exp	iresAt is not
	present, then status shall be updated to "paused", if and only if, the previous v	alue of status
	was different than "expired"	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.8.2	
Config Id	CF_01	
PICS Selection	PICS_SubUpd	
Initial conditions	with { the SUT containing an initial state }	
	Permutations Permutations	
TP Permutation id	TP/NGSI-LD/CI/SUB/029_09_01	
Name	029_09_01 Update Subscription Status To Paused	
Parent Release	v1.3.1	
Description	Check that one can update a subscription: If isActive is equal to false and exp	
	present, then status shall be updated to "paused", if and only if, the previous	/alue of status
	was different than "expired"	1 5: ()
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ngsi-ld/v1/subscriptions/{subscription_id}'	
	method set to 'PATCH'	
	Request Update Subscription and	
	Header['Content-Type'] set to 'application/json' and	
	Payload defined in file 'subscriptions/fragments/subscription-isActive-false-	
	update-sample.json'	
	}	OUT OF C
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Update Subscription with Response Status Code set to 204 and	
	Retrieve Subscription with Check Response Body containing an Attribute set to and	
	Query Parameter: expected_attribute_name set to 'status' and	
	Query Parameter: response_body set to 'response1.json()' and	
	Query Parameter: expected_attribute_value set to 'paused'	
	Query i arameter. expected_attribute_value set to pauseu	
	IJ	1

TD L	TD(NOO) D(O)(O) D(O)
TP ld	TP/NGSI-LD/CI/SUB/029_10
Test objective	Check that one can update a subscription: If only expiresAt is included and refers to a
	DateTime in the future, then status shall be updated to "active", if and only if the previous value of status was "expired"
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.8.2
Config Id	CF_01
PICS Selection	PICS_SubUpd
Initial conditions	with { the SUT containing an initial state }
	Permutations Permutations
TP Permutation id	TP/NGSI-LD/CI/SUB/029_10_01
Name	029_10_01 Activate Expired Subscription
Parent Release	v1.3.1
Description	Check that one can update a subscription: If only expiresAt is included and refers to a
-	DateTime in the future, then status shall be updated to "active", if and only if the previous
	value of status was "expired"

Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ngsi-ld/v1/subscriptions/{subscription_id}'	
	method set to 'PATCH'	
	Request Update Subscription and	
	Header['Content-Type'] set to 'application/json' and	
	Payload defined in file '\${subscription_update_fragment_file_path}'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Update Subscription with Response Status Code set to	
	204 and	
	Retrieve Subscription with Check Response Body containing an	
	Attribute set to and	
	Query Parameter: expected_attribute_name set to 'status' and	
	Query Parameter: response_body set to 'response1.json()' and	
	Query Parameter: expected_attribute_value set to 'active'	
	}	

4.4.2.2 RetrieveSubscription

TD L-I	TD/NOOL LD/OVOLD/000 00	
TP Id	TP/NGSI-LD/CI/SUB/030_02	
Test objective	Check that one cannot retrieve a subscription: If the identifier provided does not correspond	
	to any existing subscription in the system then an error of type ResourceNotF	ound shall be
_	raised	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.8.3	
Config Id	CF_01	
PICS Selection	PICS_SubRet	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/SUB/030_02_01	
Name	030_02_01 Retrieve Unknown Subscription	
Parent Release	v1.3.1	
Description	Check that one cannot retrieve a subscription: If the identifier provided does n	
	to any existing subscription in the system then an error of type ResourceNotF	ound shall be
	raised	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ngsi-ld/v1/subscriptions/{id}'	
	method set to 'GET'	
	Subscription Retrieve with the following data: and	
	Query Parameter: id set to 'urn:ngsi-ld:Subscription:unknowSubscription'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Retrieve Subscription with Response Status Code set to	
	404 and	
	Retrieve Subscription with Response Body containing the type 'https://	
	uri.etsi.org/ngsi-ld/errors/	
	ResourceNotFound' and	
	Retrieve Subscription with Response body containing 'title' element	
	}	

TP Id	TP/NGSI-LD/CI/SUB/030_01	
Test objective	Check that one cannot retrieve a subscription if the subscription Id is not a valid URI, then	
	an error of type BadRequestData shall be raised	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.8.3	
Config Id	CF_01	
PICS Selection	PICS_SubRet	
Initial conditions	with { the SUT containing an initial state }	
Permutations Permutations		
TP Permutation id	TP/NGSI-LD/CI/SUB/030_01_01	

Name	030_01_01 InvalidId	
Parent Release	v1.3.1	
Description	Check that one cannot retrieve a subscription: If the subscription Id is not presavalid URI, then an error of type BadRequestData shall be raised	sent or it is not
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/subscriptions/{id}' method set to 'GET' Subscription Retrieve with the following data: and Query Parameter: id set to 'InvalidUri' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Retrieve Subscription with Response Status Code set to 400 and Retrieve Subscription with Response Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/ BadRequestData' and Retrieve Subscription with Response body containing 'title' element }	SUT -> Client

TP Id	TP/NGSI-LD/CI/SUB/030_03	
Test objective	Check that one can retrieve a subscription	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.8.3	
Config Id	CF_01	
PICS Selection	PICS_SubRet	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/SUB/030_03_01	
Name	030_03_01 Retrieve Subscription	
Parent Release	v1.3.1	
Description	Check that one can retrieve a subscription	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/subscriptions/{id}'	
	method set to 'GET'	
	Subscription Retrieve with the following data: and	
	Query Parameter: id set to '\$	
	{subscription_id}' and	
	Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	jsonld-contexts/ngsi-ld-test-suite-compound.jsonld'	
	}	OLIT OII (
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Retrieve Subscription with Response Status Code set to 200 and	
	Retrieve Subscription with Response Body containing the same content defined in file 'subscriptions/	
	expectations/subscriptions-030-03-	
	expectations/subscriptions-050-05-	
	{subscription_id}'	
	{>ubscription_iu}	
	I/	

4.4.2.3 SubscriptionNotificationBehaviour

TP ld	TP/NGSI-LD/CI/SUB/046_16_02
Test objective	If a subscription defines an entity type selection query, a notification shall be sent whenever
	an entity matches the query.
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.8.6
Config Id	CF_01

PICS Selection	PICS_NOTIFICATIONS_SUB		
Initial conditions	with { the SUT containing an initial state }		
	Permutations		
TP Permutation id	TP/NGSI-LD/CI/SUB/046_16_02		
Name	046_16_02 Check that a notification is not sent if the entity type does not mat type selection	tch the entity	
Parent Release	V1.5.1		
Description	If a subscription defines an entity type selection query, a notification shall not entity type does not match the query	If a subscription defines an entity type selection query, a notification shall not be sent if the	
Expected behaviour	Test events	Direction	
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entities/' method set to 'POST' Request Header['Content- Type'] set to 'application/Id+json' and payload defined in file: 'vehicle-simple- attributes-sample.jsonId' }	SUT <- Client	
	then { the client at '\$ {endpoint}' receives a valid Notification containing: Notification received Waiting for no Notification data and Query Parameter: timeout set to '5' }	SUT -> Client	

TP Id	TP/NGSI-LD/CI/SUB/046_14	
Test objective	The Notification content shall be JSON-LD when endpoint accept is set to JS	SON-LD
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.8.6	3011 23
Config Id	CF_01	
PICS Selection	PICS_NOTIFICATIONS_SUB	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/SUB/046_14_01	
Name	046_14_01 Check that a notification is sent as JSON-LD	
Parent Release	v1.3.1	
Description	The Notification content shall be JSON-LD when endpoint.accept is set to	
	application/ld+json	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Update Entity Attributes and Query Parameter: id set to '\${entity_id}' and Query Parameter: fragment_filename set to '\${fragment_filename}' and Query Parameter: content_type set to 'application/ld+json' }	SUT <- Client
	then { the SUT sends a valid Response for the operation: Notification with The dictionary `\$ {notification}[data][0] ' should contain the key '@context' }	SUT -> Client

TP Id	TP/NGSI-LD/CI/SUB/046_10
Test objective	The Notification content shall be JSON by default.
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.8.6
Config Id	CF_01
PICS Selection	PICS_NOTIFICATIONS_SUB
Initial conditions	with { the SUT containing an initial state }
Permutations	
TP Permutation id	TP/NGSI-LD/CI/SUB/046_10_01
Name	046_10_01 Check that the notification is sent as JSON

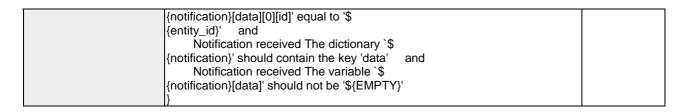
Parent Release	v1.3.1	
Description	The Notification shall be sent as JSON	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Update Entity Attributes and Query Parameter: id set to '\${entity_id}' and Query Parameter: fragment_filename set to '\${fragment_filename}' and Query Parameter: content_type set to 'application/ld+json' }	SUT <- Client
	then { the client at '\$ {endpoint}' receives a valid Notification containing: Notification received After waiting '10' seconds }	SUT -> Client

TP/NGSI-LD/CI/SUB/046_09	
A Notification shall be sent (as mandated by each concrete binding and inclu	iding any
optional endpoint.receiverInfo defined by clause 5.2.22) to the endpoint spec	ified by the
endpoint.uri member of the notification structure defined by clause 5.2.14	
ETSI GS CIM 009 V1.5.1 [1], clause 5.8.6	
PICS_NOTIFICATIONS_SUB	
with { the SUT containing an initial state }	
Permutations	
TP/NGSI-LD/CI/SUB/046_09_01	
	ified by the
	T
10010101	Direction
	SUT <- Client
Query Parameter: content_type set to application/ld+json	
than	SUT -> Client
	301 -> Client
the state of the s	
Notification received The dictionary `\$	
	A Notification shall be sent (as mandated by each concrete binding and incluoptional endpoint.receiverInfo defined by clause 5.2.22) to the endpoint speciendpoint.uri member of the notification structure defined by clause 5.2.14 ETSI GS CIM 009 V1.5.1 [1], clause 5.8.6 CF_01 PICS_NOTIFICATIONS_SUB with { the SUT containing an initial state } Permutations

TP ld	TP/NGSI-LD/CI/SUB/046_12	
	A Notification shall be sent (as mandated by each concrete binding and including any optional endpoint.info defined by clause 5.2.22) to the endpoint specified by the endpoint.uri	
	member of the notification structure defined by clause 5.2.14	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.8.6	
Config Id	CF_01	
PICS Selection	PICS_NOTIFICATIONS_SUB	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/SUB/046_12_01	
Name	046_12_01 Check that lastNotification is updated	
Parent Release	v1.3.1	

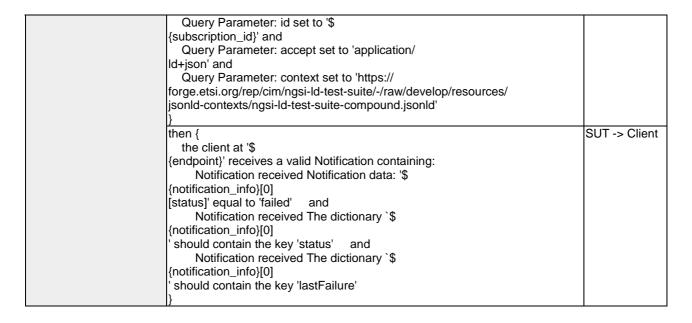
Description	The status, lastNotification and lastSuccess members shall be updated and dates. This test will check these formats.	with expected value
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/subscriptions/{id}'	
	method set to 'GET'	
	Subscription Retrieve with the following data: and	
	Query Parameter: id set to '\$	
	{subscription_id}' and	
	Query Parameter: accept set to 'application/	
	ld+json' and	
	Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	jsonld-contexts/ngsi-ld-test-suite-compound.jsonld'	
	}	
	then {	SUT -> Client
	the client at '\$	
	{endpoint}' receives a valid Notification containing:	
	Notification received After waiting '10' seconds and	
	Notification received Notification data: '\$	
	{notification_info}[0]	
	[status]' equal to 'ok' and	
	Notification received The dictionary `\$	
	{notification_info}[0]	
	' should contain the key 'status' and	
	Notification received The dictionary `\$	
	{notification_info}[0]	
	should contain the key 'lastNotification' and	
	Notification received The dictionary `\$	
	{notification_info}[0]	
	' should contain the key 'lastSuccess'	
)	

TP ld	TP/NGSI-LD/CI/SUB/046_16_01		
Test objective	If a subscription defines an entity type selection query, a notification shall	be sent whenever	
	an entity matches the query.		
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.8.6		
Config Id	CF_01		
PICS Selection	PICS_NOTIFICATIONS_SUB		
Initial conditions	with { the SUT containing an initial state }		
Permutations			
TP Permutation id	TP/NGSI-LD/CI/SUB/046_16_01		
Name	046_16_01 Check that a notification is sent with entity matching the entity	type selection	
Parent Release	v1.5.1		
Description	If a subscription defines an entity type selection query, a notification shall	be sent whenever	
	an entity matches the query.		
Expected behaviour	Test events	Direction	
	when {	SUT <- Client	
	the SUT receives a Request from the client containing:		
	URL set to '/		
	ngsi-ld/v1/entities/		
	method set to 'POST'		
	Request Header['Content-		
	Type'] set to 'application/ld+json' and		
	payload defined in file: 'building-simple-		
	attributes-sample.jsonld'		
	}		
	then {	SUT -> Client	
	the client at '\$		
	{endpoint}' receives a valid Notification containing:		
	Notification received After waiting '5' seconds and		
	Notification received Notification data: '\$		
	{notification}[subscriptionId]' equal to '\$		
	{subscription_id}' and		
	Notification received Notification data: '\$		



TP Id	TP/NGSI-LD/CI/SUB/046_15	
Test objective	If a Subscription defines a throttling member, a Notification shall not be se	nt if the throttling
rest objective		nt ii the throttling
Reference	specified (in seconds) has not elapsed yet. ETSI GS CIM 009 V1.5.1 [1], clause 5.8.6	
Config Id	CF_01	
PICS Selection	PICS_NOTIFICATIONS_SUB	
Initial conditions	with { the SUT containing an initial state }	
TD D 1.11	Permutations TRANSPORTER OF A STATE OF THE PROPERTY OF THE PRO	
TP Permutation id	TP/NGSI-LD/CI/SUB/046_15_01	
Name	046_15_01 Check that a notification is not sent if the throttling has not ela	psed yet
Parent Release	v1.3.1	
Description	If a Subscription defines a throttling member, a Notification shall not be sent if the	
	specified (in seconds) has not elapsed yet.	D'
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to /	
	ngsi-ld/v1/entities/{entityld}/attrs/{attributeId}'	
	method set to 'PATCH'	
	Update Entity Attributes and	
	Query Parameter: id set to '\$	
	{entity_id}' and	
	Query Parameter: fragment_filename set to 'airQualityLevel-	
	fragment.jsonld' and	
	Query Parameter: content_type set to 'application/	
	ld+json'	
	}	
	then {	SUT -> Client
	the client at '\$	
	{endpoint}' receives a valid Notification containing:	
	Notification received After waiting '10' seconds and	
	Notification received Waiting for no Notification data and	
	Query Parameter: timeout set to '5'	
	}	

TP ld	TP/NGSI-LD/CI/SUB/046 13	
Test objective	If the response to the notification request is different than 200 OK then implem	entations
rest objective	shall: Update notification.lastFailure with a timestamp representing the current	
		uate and
	time, update notification.status to "failed"	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.8.6	
Config Id	CF_01	
PICS Selection	PICS_NOTIFICATIONS_SUB	
Initial conditions	with { the SUT containing an initial state }	
Permutations		
TP Permutation id	TP/NGSI-LD/CI/SUB/046_13_01	
Name	046_13_01 Check that lastFailure and status are updated if a notification could	d not be sent
Parent Release	v1.3.1	
Description	If the response to the notification request is different than 200 OK then implem	entations
	shall: Update notification.lastFailure with a timestamp representing the current	date and
	time, update notification.status to "failed"	
Expected behaviour	Test events	Direction
-	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to /	
	ngsi-Id/v1/subscriptions/{id}'	
	method set to 'GET'	
	Subscription Retrieve with the following data: and	
	Subscription retrieve with the following data, and	



TP ld	TP/NGSI-LD/CI/SUB/046_11			
Test objective	A Notification shall be sent (as mandated by each concrete binding and including any			
	optional endpoint.info defined by clause 5.2.22) to the endpoint specified by th	e endpoint.uri		
	member of the notification structure defined by clause 5.2.14			
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.8.6			
Config Id	CF_01			
PICS Selection	PICS_NOTIFICATIONS_SUB			
Initial conditions	with { the SUT containing an initial state }			
Permutations				
TP Permutation id	TP/NGSI-LD/CI/SUB/046_11_01			
Name	046_11_01 Check that timesSent is increased by one			
Parent Release	v1.3.1			
Description	The notification.timesSent member shall be incremented by one.			
Expected behaviour	Test events	Direction		
	when {	SUT <- Client		
	the SUT receives a Request from the client containing:			
	URL set to '/			
	ngsi-ld/v1/subscriptions/{id}'			
	method set to 'GET'			
	Subscription Retrieve with the following data: and			
	Query Parameter: id set to '\$			
	{subscription_id}' and			
	Query Parameter: accept set to 'application/			
	ld+json' and			
	Query Parameter: context set to 'https://			
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/			
	jsonld-contexts/ngsi-ld-test-suite-compound.jsonld'			
	}			
	then {	SUT -> Client		
	the client at '\$			
	{endpoint}' receives a valid Notification containing:			
	Notification received After waiting '10' seconds and			
	Notification received Notification data: '\$			
	{response.json()}[notification]			
	[timesSent]' equal to '\${1}			
	' and			
	Retrieve Subscription received The dictionary `\$			
	{response.json()}			
	[notification]' should contain the key 'timesSent'			
]}			

TP Id	TP/NGSI-LD/CI/SUB/046_08
Test objective	Check that a notification is only sent if and only if the status is active
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.8.6

Config Id	CF_01	
PICS Selection	PICS_NOTIFICATIONS_SUB	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing an initial Sub	scription
	\${subscription} with an id set to \${subscription_id} }	
TD Dammartation id	Permutations Provided to an extension Provided to the control of t	
TP Permutation id	TP/NGSI-LD/CI/SUB/046_08_01	
Name Parent Release	046_08_01 Check that a notification is sent with all attributes v1.3.1	
Description	The structure of the notification message shall be as mandated by clause 5.3	1 The
Description	absence of the notification.attributes member of a Subscription means that all Attributes shall be included. All attributes are included	
Expected behaviour	Test events	Direction
·	when { the SUT receives a Request from the client containing: URL set to '/	SUT <- Client
	ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}'	
	method set to 'PATCH' Update Entity Attributes and	
	Query Parameter: id set to '\$ {entity_id}' and	
	Query Parameter: fragment_filename set to 'airQualityLevel- fragment.jsonId' and	
	Query Parameter: content_type set to 'application/ Id+json'	
	then {	SUT -> Client
	the client at '\$	SOT -> CHERT
	{endpoint}' receives a valid Notification containing: Notification received After waiting '5' seconds and	
	Notification received Notification data: '\$ {notification}[subscriptionId]' equal to '\$	
	{subscription_id}' and Notification received Notification data: '\$	
	{notification}[data][0][id]' equal to '\$	
	{entity_id}' and Notification received Notification data: '\$	
	{notification}[data][0][name] [value]' equal to 'Eiffel Tower' and	
	Notification received The dictionary `\$ {notification}' should contain the key 'data' and	
	Notification received The dictionary `\$	
	{notification}[data][0] ' should contain the key 'almostFull' and	
	Notification received The dictionary `\$ [{notification}[data][0]	
	should contain the key 'airQualityLevel' and Notification received The dictionary `\$	
	{notification}[data][0]	
	should contain the key 'subCategory' and Notification received The variable `\$	
	{notification}[data]' should not be '\${EMPTY}'	
TP Permutation id	TP/NGSI-LD/CI/SUB/046_08_02	l
Name	046_08_02 Check that a notification is sent with all attributes in simplified form	nat
Parent Release	v1.3.1	4 =-
Description	The structure of the notification message shall be as mandated by clause 5.3 absence of the notification.attributes member of a Subscription means that all	Entity
	Attributes shall be included If the notification format member value is "keyValue implified representation of the notification format member value is "keyValue implified representation of the notification format member value is "keyValue".	
Expected behaviour	simplified representation of the entities (as mandated by clause 4.5.3) shall be Test events	Direction
Expected beliaviour	when {	SUT <- Client
	the SUT receives a Request from the client containing:	SOT S- CHEFT
	LIPL set to '/	
	URL set to '/ ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}'	
	ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH'	
	ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}'	

Query Parameter: fragment_filename set to 'airQualityLevel-	
fragment.jsonId' and	
Query Parameter: content_type set to 'application/	
ld+json'	
}	
then {	SUT -> Client
the client at '\$	
{endpoint}' receives a valid Notification containing:	
Notification received After waiting '5' seconds and	
Notification received Notification data: '\$	
[notification][subscriptionId]' equal to '\$	
{subscription_id}' and	
Notification received Notification data: '\$	
[notification][data][0][id]' equal to '\$	
{entity_id}' and	
Notification received Notification data: '\$	
[notification][data][0]	
[name]' equal to 'Eiffel Tower' and	
Notification received The dictionary `\$	
{notification}' should contain the key 'data' and	
Notification received The variable `\$	
{notification}[data]' should not be '\${EMPTY}'	
}	

TDIA	TD/NCCLLD/CL/DID/O4C O4	
TP Id	TP/NGSI-LD/CI/SUB/046_04	
Test objective	If a Subscription does not define a timeInterval member, the notification shall be	
	whenever there is a change in the watched Attributes. The notification message	
	include all the subscribed Entities that changed and that match (as mandated	by clauses 4.9
Defenses	and 4.10) the query and geoquery conditions	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.8.6	
Config Id	CF_01	
PICS Selection	PICS_NOTIFICATIONS_SUB	
Initial conditions	with { the SUT containing an initial state }	
TD D	Permutations Provide the second secon	
TP Permutation id	TP/NGSI-LD/CI/SUB/046_04_01	
Name	046_04_01 Check that a notification is sent with all entities	
Parent Release	v1.3.1	
Description	The notification message shall include all the subscribed Entities that changed	
	match (as mandated by clauses 4.9 and 4.10) the query and geoquery condition	
Expected behaviour	Test events	Direction
	The client at \$ {endpoint} receives a valid Notification containing a	SUT <- Client
	subscription with id set to \${subscriptionId} and status equals 'active'	
	and timeInterval is set to '\$ {timeInterval}' and watchedAttributes is	
	'Empty' and q is 'Empty' and geoQ is 'Empty' and with	
	subscription.entity with type 'Building' and notification.endpoint.accept	
	is 'application/ json' When the timeinterval is reached at	
	\${timeInterval} seconds the SUT needs to send out a notification to	
	the client sends a notification to the client every \${timeInterval}	
	seconds	
	then {	SUT -> Client
	the client at '\$	
	{endpoint}' receives a valid Notification containing:	
	Notification received After waiting '5' seconds and	
	Notification received Notification data: '\$	
	{notification}[subscriptionId]' equal to '\$	
	{subscription_id}' and	
	Notification received Notification data: '\$	
	(notification)[data][0][id]' equal to '\$	
	{entity_id}' and	
	Notification received Notification data: '\$	
	[notification][data][0][airQualityLevel]	
	[value]' equal to '\${5}' and	
	Notification received Notification data: '\$	
	[notification][data][0][name]	
	[value]' equal to 'Eiffel Tower' and	
	Notification received The dictionary `\$	
	{notification}' should contain the key 'data' and	

Notification received The variable `\$	
{notification}[data]' should not be '\${EMPTY}'	
}	

TP ld	TD/NCSLLD/CL/SLID/046 06	
Test objective	TP/NGSI-LD/CI/SUB/046_06 If a Subscription does not define a timeInterval member, the notification shall be	o cont
rest objective		
	whenever an entity matches the query defined in the subscription. The notifical	
	shall include all the subscribed Entities that changed and that match (as mand	ated by
Reference	clauses 4.9 and 4.10) the query and geoquery conditions	
	ETSI GS CIM 009 V1.5.1 [1], clause 5.8.6	
Config Id	CF_01	
PICS Selection	PICS_NOTIFICATIONS_SUB	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/SUB/046_06_01	
Name	046_06_01 Check that a notification is sent with all matching entities	
Parent Release	v1.3.1	
Description	only the subscribed Entities matching the query and watched attributes shall b	e included.
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/'	
	method set to 'POST'	
	Request Header['Content-	
	Type'] set to 'application/ld+json' and	
	payload defined in file: 'building-location-	
	attribute.jsonld'	
	}	
	then {	SUT -> Client
	the client at '\$	
	{endpoint}' receives a valid Notification containing:	
	Notification received After waiting '10' seconds and	
	Notification received Notification data: '\$	
	{notification}[subscriptionId]' equal to '\$	
	{subscription_id}' and	
	Notification received Notification data: '\$	
	{notification}[data][0][id]' equal to '\${entity_id}'	
	 }	

TP Id	TP/NGSI-LD/CI/SUB/046 02	
Test objective	If a Subscription defines a timeInterval member, a Notification shall be sent pe when the time interval (in seconds) specified in such value field is reached, reg Attribute changes.	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.8.6	
Config Id	CF_01	
PICS Selection	PICS_NOTIFICATIONS_SUB	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/SUB/046_02_01	
Name	046_02_01 Check that a notification is sent on the timeInterval	
Parent Release	v1.3.1	
Description	If a Subscription defines a timeInterval member, a Notification shall be sent pe when the time interval (in seconds) specified in such value field is reached, regattribute changes.	
Expected behaviour	Test events	Direction
	The client at \$ {endpoint} receives a valid Notification containing a subscription with id set to \${subscriptionId} and status equals 'active' and timeInterval is set to '\$ {timeInterval}' and watchedAttributes is 'Empty' and q is 'Empty' and geoQ is 'Empty' and with subscription.entity with type 'Building' and notification.endpoint.accept is 'application/ json' When the timeinterval is reached at \${timeInterval} seconds the SUT needs to send out a notification to the client sends a notification to the client every \${timeInterval} seconds	SUT <- Client

then {	SUT -> Client
the client at '\$	
{endpoint}' receives a valid Notification containing:	
Notification received After waiting '15' seconds and	
Notification received After waiting '15' seconds and	
Notification received Notification data: '\$	
{notification}[subscriptionId]' equal to '\$	
{subscription_id}' and	
Notification received Notification data: '\$	
{notification}[data][0][id]' equal to '\$	
{entity_id}' and	
Notification received Notification data: '\$	
{notification}[data][0][name]	
[value]' equal to 'Eiffel Tower' and	
Notification received The dictionary `\$	
{notification}' should contain the key 'data' and	
Notification received The variable `\$	
{notification}[data]' should not be '\$	
{EMPTY}' and	
Notification received The expression `'\$	
{notification}[data][0][airQualityLevel][value]'=='4.0' or '\$	
{notification}[data][0][airQualityLevel]	
[value]'=='4" should be True	
}	

I		
TP ld	TP/NGSI-LD/CI/SUB/046_07	
Test objective	Check that a notification is only sent if and only if the status is active	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.8.6	
Config Id	CF_01	
PICS Selection	PICS_NOTIFICATIONS_SUB	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing an initial Subs	scription
	\${subscription} with an id set to \${subscription_id} }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/SUB/046_07_01	
Name	046_07_01 Check notification structure	
Parent Release	v1.3.1	
Description	The structure of the notification message shall be as mandated by clause 5.3.	Valid
	notification with attributes as stated above	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}'	
	method set to 'PATCH'	
	Update Entity Attributes and	
	Query Parameter: id set to '\${entity_id}' and	
	Query Parameter: fragment_filename set to '\${fragment_filename}' and	
	Query Parameter: content_type set to 'application/ld+json'	
	}	
	then {	SUT -> Client
	the client at '\$	
	{endpoint}' receives a valid Notification containing:	
	Notification received After waiting '5' seconds	
	}	
TP Permutation id	TP/NGSI-LD/CI/SUB/046_07_02	
Name	046_07_02 Check correct attributes are included	
Parent Release	v1.3.1	
Description	The structure of the notification message shall be as mandated by clause 5.3.	The Entity
	Attributes included (Properties or Relationships) shall be those specified by the	е
	notification.attributes member in the Subscription data type (clause 5.2.12).	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}'	
	method set to 'PATCH'	
	Update Entity Attributes and	
	Query Parameter: id set to '\${entity_id}' and	

	Query Parameter: fragment_filename set to '\$\{\text{fragment_filename}\}' \text{ and Query Parameter: content_type set to 'application/ld+json'}\} then { the client at '\$ {\text{endpoint}\}' receives a valid Notification containing: Notification received After waiting '5' seconds and Notification received The dictionary `\$ {\text{notification}\}[data][0] ' should contain the key 'airQualityLevel'}	SUT -> Client
TP Permutation id	TP/NGSI-LD/CI/SUB/046_07_03	_
Name	046_07_03 Check URI expansion is observed	
Parent Release	v1.3.1	
Description	The structure of the notification message shall be as mandated by clause 5.3	. URI
	expansion shall be observed (clause 5.5.7).	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Update Entity Attributes and Query Parameter: id set to '\${entity_id}' and Query Parameter: fragment_filename set to '\${fragment_filename}' and	SUT <- Client
	Query Parameter: content_type set to 'application/ld+json' } then {	SUT -> Client

TP ld	TP/NGSI-LD/CI/SUB/046_05	
Test objective	If a Subscription does not define a timeInterval member, the notification shall be whenever there is a change in the watched Attributes and the entity matches the parameter. The notification message shall include all the subscribed Entities the and that match (as mandated by clauses 4.9 and 4.10) the query and geoquer	ne q nat changed
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.8.6	
Config Id	CF_01	
PICS Selection	PICS_NOTIFICATIONS_SUB	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/SUB/046_05_01	
Name	046_05_01 Check that a notification is sent with all entities	
Parent Release	v1.3.1	
Description	The notification message shall include all the subscribed Entities that changed match (as mandated by clauses 4.9 and 4.10) the query and geoquery condition	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Update Entity Attributes and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: fragment_filename set to 'airQualityLevel-fragment.jsonId' and Query Parameter: content_type set to 'application/ Id+json' }	SUT <- Client

then {	SUT -> Client
the client at '\$	
{endpoint}' receives a valid Notification containing:	
Notification received After waiting '5' seconds and	
Notification received Notification data: '\$	
{notification}[subscriptionId]' equal to '\$	
{subscription id}' and	
Notification received Notification data: '\$	
{notification}[data][0][id]' equal to '\$	
{entity id}' and	
Notification received Notification data: '\$	
{notification}[data][0][name]	
[value] equal to 'Eiffel Tower' and	
Notification received The dictionary `\$	
{notification}' should contain the key 'data' and	
Notification received The variable `\$	
{notification}[data]' should not be '\$	
{EMPTY}' and	
Notification received The expression `\\$ Instification Idata IO Idata IO	
{notification}[data][0][airQualityLevel][value]'=='5.0' or '\$	
{notification}[data][0][airQualityLevel]	
[value]'=='5" should be True	
}	

TP ld	TP/NGSI-LD/CI/SUB/046_01	
Test objective	Check that a notification is only sent if and only if the status is active	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.8.6	
Config Id	CF_01	
PICS Selection	PICS_NOTIFICATIONS_SUB	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/SUB/046_01_01	
Name	046_01_01 Check that a notification is only sent if status is active	
Parent Release	v1.3.1	
Description	Check that a notification is only sent if and only if the status is active	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}'	
	method set to 'PATCH'	
	Update Entity Attributes and	
	Query Parameter: id set to '\$	
	{entity_id}' and	
	Query Parameter: fragment_filename set to 'airQualityLevel-	
	fragment.jsonld' and	
	Query Parameter: content_type set to 'application/	
	ld+json'	
	}	0
	then {	SUT -> Client
	the client at '\$	
	{endpoint}' receives a valid Notification containing:	
	Notification received After waiting '10' seconds and	
	Notification received Notification data: '\$	
	[notification][subscriptionId] equal to '\$	
	{subscription_id}' and	
	Notification received Notification data: '\$	
	{notification}[data][0][id]' equal to '\$ {entity id}' and	
	Notification received Notification data: '\$ {notification}[data][0][airQualityLevel]	
	[value] equal to '\${5}'	
	13	

TP Id	TP/NGSI-LD/CI/SUB/046_03	
Test objective	A notification with all subscribed Entities will be included if query or geoquery are not	
,	defined.	2.0
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.8.6	
Config Id	CF_01	
PICS Selection	PICS_NOTIFICATIONS_SUB	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/SUB/046_03_01	
Name	046_03_01 Check that a notification is sent with all entities	
Parent Release	v1.3.1	
Description	A notification with all subscribed Entities will be included if query or geoquery defined.	are not
Expected behaviour	Test events	Direction
	The client at \$ {endpoint} receives a valid Notification containing a subscription with id set to \$ {subscriptionId} and status equals 'active' and timeInterval is set to '\$ {timeInterval}' and watchedAttributes is 'Empty' and q is 'Empty' and geoQ is 'Empty' and with subscription.entity with type 'Building' and notification.endpoint.accept is 'application/ json' When the timeinterval is reached at \$ {timeInterval} seconds the SUT needs to send out a notification to the client sends a notification to the client every \$ {timeInterval} seconds then { the client at '\$ {endpoint}' receives a valid Notification containing: Notification received After waiting '5' seconds and Notification received Notification data: '\$ {notification}{subscriptionId}' equal to '\$ {subscription_id}' and Notification received Notification data: '\$ {notification}{data}[0][id]' equal to '\$ {entity_id}' and Notification received Notification data: '\$ {notification}{catan}[data][0][name] [value]' equal to 'Eiffel Tower' and Notification received The dictionary `\$ {notification} chata]' should not be '\$ {entity_id} and Notification received The variable `\$ {notification}[data]' should not be '\$ {entitication}[data][0][airQualityLevel][value]'=='5.0' or '\$ {notification}[data][0][airQualityLevel][value]'=='5.0' or '\$ {notification}[data][0][airQualityLevel][value]'=-5.0' or '\$ {notification}[data][0][airQualityLevel][ontificat	SUT -> Client

4.4.2.4 QuerySubscriptions

TP ld	TP/NGSI-LD/CI/SUB/031_02
Test objective	Check that one can query a list of subscriptions: Pagination logic shall be in place
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.8.4
Config Id	CF_01
PICS Selection	PICS_RetLisSub
Initial conditions	with { the SUT being in the "initial state" and the SUT containing an initial Subscription
	\${subscription} with an id set to \${subscription_id} }
	Permutations
TP Permutation id	TP/NGSI-LD/CI/SUB/031_02_01
Name	031_02_01 Query Second Subscription
Parent Release	v1.3.1
Description	Check that one can query a list of subscriptions: Pagination logic shall be in place

Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/subscriptions/'	
	method set to 'GET'	
	Query Subscription Request with data: and Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	isonId-contexts/ngsi-Id-test-suite-compound.jsonId and	
	Query Parameter: limit set to '\${1}	
	'and	
	Query Parameter: offset set to '\${1}	
	'and	
	Query Parameter: accept set to 'application/	
	ld+json'	
	than (SUT -> Client
	then { the SUT sends a valid Response for the operations:	301 -> Client
	Query Subscriptions with Response Status Code set to	
	200 and	
	Query Subscriptions with Response Body containing a list of entities	
	equal to '\$	
	{expectation_subscription_number}' of type 'Subscription' with response set	
	to '\$	
	{response.json()}' and	
	Query Subscriptions with Response header with 'Prev' header equal to	
	' </th <th></th>	
	ngsi-ld/v1/subscriptions? limit=1&offset=2>;rel="next";type="application/	
	Id+json" and with 'Next' header equal to '\$	
	{response.headers}'	
	}	
TP Permutation id	TP/NGSI-LD/CI/SUB/031_02_02	•
Name	031_02_02 Query Last Subscription	
Name Parent Release	031_02_02 Query Last Subscription v1.3.1	
Name Parent Release Description	031_02_02 Query Last Subscription v1.3.1 Check that one can query a list of subscriptions: Pagination logic shall be in p	
Name Parent Release	031_02_02 Query Last Subscription v1.3.1 Check that one can query a list of subscriptions: Pagination logic shall be in p Test events	Direction
Name Parent Release Description	031_02_02 Query Last Subscription v1.3.1 Check that one can query a list of subscriptions: Pagination logic shall be in p Test events when {	
Name Parent Release Description	031_02_02 Query Last Subscription v1.3.1 Check that one can query a list of subscriptions: Pagination logic shall be in p Test events when { the SUT receives a Request from the client containing:	Direction
Name Parent Release Description	031_02_02 Query Last Subscription v1.3.1 Check that one can query a list of subscriptions: Pagination logic shall be in p Test events when { the SUT receives a Request from the client containing: URL set to '/	Direction
Name Parent Release Description	031_02_02 Query Last Subscription v1.3.1 Check that one can query a list of subscriptions: Pagination logic shall be in p Test events when { the SUT receives a Request from the client containing:	Direction
Name Parent Release Description	031_02_02 Query Last Subscription v1.3.1 Check that one can query a list of subscriptions: Pagination logic shall be in p Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/subscriptions/' method set to 'GET' Query Subscription Request with data: and	Direction
Name Parent Release Description	031_02_02 Query Last Subscription v1.3.1 Check that one can query a list of subscriptions: Pagination logic shall be in p Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/subscriptions/' method set to 'GET' Query Subscription Request with data: and Query Parameter: context set to 'https://	Direction
Name Parent Release Description	031_02_02 Query Last Subscription v1.3.1 Check that one can query a list of subscriptions: Pagination logic shall be in p Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/subscriptions/' method set to 'GET' Query Subscription Request with data: and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	Direction
Name Parent Release Description	031_02_02 Query Last Subscription v1.3.1 Check that one can query a list of subscriptions: Pagination logic shall be in p Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/subscriptions/' method set to 'GET' Query Subscription Request with data: and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and	Direction
Name Parent Release Description	031_02_02 Query Last Subscription v1.3.1 Check that one can query a list of subscriptions: Pagination logic shall be in p Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/subscriptions/' method set to 'GET' Query Subscription Request with data: and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: limit set to '\${1}	Direction
Name Parent Release Description	031_02_02 Query Last Subscription v1.3.1 Check that one can query a list of subscriptions: Pagination logic shall be in p Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/subscriptions/' method set to 'GET' Query Subscription Request with data: and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: limit set to '\${1} ' and	Direction
Name Parent Release Description	031_02_02 Query Last Subscription v1.3.1 Check that one can query a list of subscriptions: Pagination logic shall be in p Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/subscriptions/' method set to 'GET' Query Subscription Request with data: and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: limit set to '\${1} ' and Query Parameter: offset set to '\${2}	Direction
Name Parent Release Description	v1.3.1 Check that one can query a list of subscriptions: Pagination logic shall be in p Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/subscriptions/' method set to 'GET' Query Subscription Request with data: and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: limit set to '\${1} ' and Query Parameter: offset set to '\${2} ' and	Direction
Name Parent Release Description	031_02_02 Query Last Subscription v1.3.1 Check that one can query a list of subscriptions: Pagination logic shall be in p Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/subscriptions/' method set to 'GET' Query Subscription Request with data: and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: limit set to '\${1} ' and Query Parameter: offset set to '\${2}	Direction
Name Parent Release Description	v1.3.1 Check that one can query a list of subscriptions: Pagination logic shall be in p Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/subscriptions/' method set to 'GET' Query Subscription Request with data: and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: limit set to '\${1} ' and Query Parameter: offset set to '\${2} ' and Query Parameter: accept set to 'application/	Direction
Name Parent Release Description	v1.3.1 Check that one can query a list of subscriptions: Pagination logic shall be in p Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/subscriptions/' method set to 'GET' Query Subscription Request with data: and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: limit set to '\${1} ' and Query Parameter: offset set to 'application/ ld+json' } then {	Direction
Name Parent Release Description	v1.3.1 Check that one can query a list of subscriptions: Pagination logic shall be in p Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/subscriptions/' method set to 'GET' Query Subscription Request with data: and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: limit set to '\${1} ' and Query Parameter: offset set to 'application/ ld+json' } then { the SUT sends a valid Response for the operations:	Direction SUT <- Client
Name Parent Release Description	v1.3.1 Check that one can query a list of subscriptions: Pagination logic shall be in p Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/subscriptions/' method set to 'GET' Query Subscription Request with data: and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: limit set to '\${1} ' and Query Parameter: offset set to '\${2} ' and Query Parameter: accept set to 'application/ ld+json' } then { the SUT sends a valid Response for the operations: Query Subscriptions with Response Status Code set to	Direction SUT <- Client
Name Parent Release Description	031_02_02 Query Last Subscription v1.3.1 Check that one can query a list of subscriptions: Pagination logic shall be in p Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/subscriptions/' method set to 'GET' Query Subscription Request with data: and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: limit set to '\${1} ' and Query Parameter: offset set to '\${2} ' and Query Parameter: accept set to 'application/ ld+json' } then { the SUT sends a valid Response for the operations: Query Subscriptions with Response Status Code set to	Direction SUT <- Client
Name Parent Release Description	031_02_02 Query Last Subscription v1.3.1 Check that one can query a list of subscriptions: Pagination logic shall be in p Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/subscriptions/' method set to 'GET' Query Subscription Request with data: and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: limit set to '\${1} ' and Query Parameter: offset set to '\${2} ' and Query Parameter: accept set to 'application/ Id+json' } then { the SUT sends a valid Response for the operations: Query Subscriptions with Response Status Code set to 200 and Query Subscriptions with Response Body containing a list of entities	Direction SUT <- Client
Name Parent Release Description	v1.3.1 Check that one can query a list of subscriptions: Pagination logic shall be in p Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/subscriptions/' method set to 'GET' Query Subscription Request with data: and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ isonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: limit set to '\${1} 'and Query Parameter: offset set to '\${2} 'and Query Parameter: accept set to 'application/ Id+json' } then { the SUT sends a valid Response for the operations: Query Subscriptions with Response Status Code set to 200 and Query Subscriptions with Response Body containing a list of entities equal to '\$	Direction SUT <- Client
Name Parent Release Description	031_02_02 Query Last Subscription v1.3.1 Check that one can query a list of subscriptions: Pagination logic shall be in p Test events When {	Direction SUT <- Client
Name Parent Release Description	031_02_02 Query Last Subscription v1.3.1 Check that one can query a list of subscriptions: Pagination logic shall be in p Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/subscriptions/' method set to 'GET' Query Subscription Request with data: and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: limit set to '\${1} ' and Query Parameter: offset set to '\${2} ' and Query Parameter: accept set to 'application/ ld+json' } then { the SUT sends a valid Response for the operations: Query Subscriptions with Response Status Code set to 200 and Query Subscriptions with Response Body containing a list of entities equal to '\$ {expectation_subscription_number}' of type 'Subscription' with response set to '\$	Direction SUT <- Client
Name Parent Release Description	031_02_02 Query Last Subscription v1.3.1 Check that one can query a list of subscriptions: Pagination logic shall be in p Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/subscriptions/' method set to 'GET' Query Subscription Request with data: and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: limit set to '\${1} ' and Query Parameter: offset set to '\${2} 'and Query Parameter: accept set to 'application/ Id+json' } then { the SUT sends a valid Response for the operations: Query Subscriptions with Response Status Code set to 200 and Query Subscriptions with Response Body containing a list of entities equal to '\$ {expectation_subscription_number}' of type 'Subscription' with response set to '\$ {response.json()}' and	Direction SUT <- Client
Name Parent Release Description	031_02_02 Query Last Subscription v1.3.1 Check that one can query a list of subscriptions: Pagination logic shall be in p Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/subscriptions/' method set to 'GET' Query Subscription Request with data: and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: limit set to '\${1} ' and Query Parameter: offset set to '\${2} ' and Query Parameter: accept set to 'application/ ld+json' } then { the SUT sends a valid Response for the operations: Query Subscriptions with Response Status Code set to 200 and Query Subscriptions with Response Body containing a list of entities equal to '\$ {expectation_subscription_number}' of type 'Subscription' with response set to '\$	Direction SUT <- Client
Name Parent Release Description	O31_02_02 Query Last Subscription V1.3.1 Check that one can query a list of subscriptions: Pagination logic shall be in p Test events When {	Direction SUT <- Client
Name Parent Release Description	031_02_02 Query Last Subscription v1.3.1 Check that one can query a list of subscriptions: Pagination logic shall be in p Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/subscriptions/' method set to 'GET' Query Subscription Request with data: and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/ jsonId-contexts/ngsi-Id-test-suite-compound.jsonId' and Query Parameter: limit set to '\${1} 'and Query Parameter: offset set to '\${2} 'and Query Parameter: accept set to 'application/ Id+json' } then { the SUT sends a valid Response for the operations: Query Subscriptions with Response Status Code set to 200 and Query Subscriptions with Response Body containing a list of entities equal to '\$ {expectation_subscription_number}' of type 'Subscription' with response set to '\$ response.json()}' and Query Subscriptions with Response header with 'Prev' header equal to '\$	Direction SUT <- Client

TP Permutation id	TP/NGSI-LD/CI/SUB/031_02_03	
Name	031_02_03 Query All Subscriptions	
Parent Release	v1.3.1	
Description	Check that one can query a list of subscriptions: Pagination logic shall be in place	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/subscriptions/' method set to 'GET' Query Subscription Request with data: and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' and Query Parameter: limit set to '\${15} ' and Query Parameter: offset set to '\${0} ' and Query Parameter: accept set to 'application/ ld+json' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Query Subscriptions with Response Status Code set to 200 and Query Subscriptions with Response Body containing a list of entities equal to '\$ {expectation_subscription_number}' of type 'Subscription' with response set to '\$ {response.json()}' and Query Subscriptions with Response header with 'Prev' header equal to '\$ {EMPTY}' and with 'Next' header equal to '\$ {response.headers}' }	SUT -> Client

TP ld	TP/NGSI-LD/CI/SUB/031_01	
	_	
Test objective	Check that one can query a list of subscriptions	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.8.4	
Config Id	CF_01	
PICS Selection	PICS_RetLisSub	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/SUB/031_01_01	
Name	031_01_01 Query Subscriptions	
Parent Release	v1.3.1	
Description	Check that one can query a list of subscriptions	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/subscriptions/' method set to 'GET' Query Subscription Request with data: and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Query Subscriptions with Response Status Code set to 200 and Query Subscriptions with Response containing: * file set to 'subscriptions/ expectations/subscriptions-031-01-expectation.json' * id set to '\$	SUT -> Client



4.4.2.5 DeleteSubscription

TP ld	TP/NGSI-LD/CI/SUB/032_02	
Test objective	Check that one cannot delete a subscription: If the subscription id provided do	es not
	correspond to any existing subscription in the system then an error of type	
	ResourceNotFound shall be raised	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.8.5	
Config Id	CF_01	
PICS Selection	PICS_SubDel	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/SUB/032_02_01	
Name	032_02_01 Delete Unknown Subscription	
Parent Release	v1.3.1	
Description	Check that one cannot delete a subscription: If the subscription id provided do	es not
	correspond to any existing subscription in the system then an error of type	
	ResourceNotFound shall be raised	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/subscriptions/{subscription_id}'	
	method set to 'DELETE'	
	Delete Subscription with id set to 'urn:ngsi-	
	ld:Subscription:unknowSubscription'	
	then {	SUT -> Client
		00. 7 0
	the SUT sends a valid Response for the operations:	
	the SUT sends a valid Response for the operations: Delete Subscription with Response Status Code set to	
	the SUT sends a valid Response for the operations: Delete Subscription with Response Status Code set to 404 and	
	the SUT sends a valid Response for the operations: Delete Subscription with Response Status Code set to 404 and Delete Subscription with Response Body containing the type 'https://	
	the SUT sends a valid Response for the operations: Delete Subscription with Response Status Code set to 404 and	
	the SUT sends a valid Response for the operations: Delete Subscription with Response Status Code set to 404 and Delete Subscription with Response Body containing the type 'https://uri.etsi.org/ngsi-ld/errors/	

TP ld	TP/NGSI-LD/CI/SUB/032 03	
	_	
Test objective	Check that one can delete a subscription	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.8.5	
Config Id	CF_01	
PICS Selection	PICS_SubDel	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/SUB/032_03_01	
Name	032_03_01 Delete Subscription	
Parent Release	v1.3.1	
Description	Check that one can delete a subscription	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/subscriptions/{subscription_id}'	
	method set to 'DELETE'	
	Delete Subscription with id set to '\$	
	{subscription_id}'	
]}	

then {	SUT -> Client
the SUT sends a valid Response for the operations:	
Delete Subscription with Response Status Code set to	
204 and	
Retrieve Subscription with Response Status Code set to	
\${response1.status_code}	
]	

TP/NGSI-LD/CI/SUB/032 01	
Check that one cannot delete a subscription: If the subscription Id is not present or it is not a	
ETSI GS CIM 009 V1.5.1 [1], clause 5.8.5	
CF_01	
PICS SubDel	
Permutations	
TP/NGSI-LD/CI/SUB/032_01_01	
032_01_01 NotPresentId	
v1.3.1	
Check that one cannot delete a subscription: If the subscription Id is not presvalid URI, then an error shall be raised	ent or it is not a
Test events	Direction
when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/subscriptions/{subscription_id}' method set to 'DELETE' Delete Subscription with id set to '\${EMPTY}' }	SUT <- Client
then { the SUT sends a valid Response for the operations: Delete Subscription with Response Status Code set to 405 and Delete Subscription with Response Body containing the type '\$ {EMPTY}' and Delete Subscription with Response body containing 'title' element }	SUT -> Client
TP/NGSI-LD/CI/SUB/032_01_02	•
032_01_02 InvalidId	
v1.3.1	
Check that one cannot delete a subscription: If the subscription Id is not presvalid URI, then an error shall be raised	
Test events	Direction
when { the SUT receives a Request from the client containing: URL set to '/ngsi-Id/v1/subscriptions/{subscription_id}' method set to 'DELETE' Delete Subscription with id set to 'InvalidUri' }	SUT <- Client
then { the SUT sends a valid Response for the operations: Delete Subscription with Response Status Code set to 400 and Delete Subscription with Response Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/ BadRequestData' and Delete Subscription with Response body containing 'title' element }	SUT -> Client
	valid URI, then an error shall be raised ETSI GS CIM 009 V1.5.1 [1], clause 5.8.5 CF_01 PICS_SubDel with { the SUT containing an initial state }

4.4.2.6 CreateSubscription

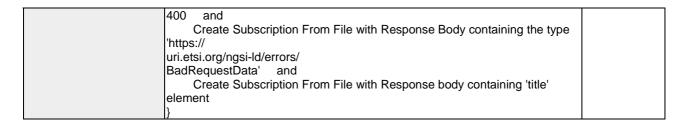
TP Id	TP/NGSI-LD/CI/SUB/028_01
Test objective	Check that one can create a subscription
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.8.1
Config Id	CF_01

PICS Selection	PICS_CreSub	
Initial conditions	with { the SUT containing an initial state }	
Permutations		
TP Permutation id	TP/NGSI-LD/CI/SUB/028_01_01	
Name	028_01_01 Create Subscription	
Parent Release	v1.3.1	
Description	Check that one can create a subscription	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/subscriptions/' method set to 'POST' Request Header['Content- Type'] set to 'application/ld+json' and payload defined in file: 'subscriptions/ subscription-sample.jsonId' }	SUT <- Client
	then { the SUT sends a valid Response for the operation: Retrieve Subscription with Check Created Resource Set To and Query Parameter: 'ignored_keys' set to '\$ {None}' list of keys }	SUT -> Client

TP Id	TD/NCCL L D/CL/CLID/000, 00	1
	TP/NGSI-LD/CI/SUB/028_03	
Test objective	Check that one cannot create a subscription with an invalid/empty id	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.8.1	
Config Id	CF_01	
PICS Selection	PICS_CreSub	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/SUB/028_03_01	
Name	028_03_01 InvalidId	
Parent Release	v1.3.1	
Description	Check that one cannot create a subscription with an invalid/empty id	,
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/subscriptions/	
	method set to 'POST'	
	Request Header['Content-	
	Type'] set to 'application/ld+json' and	
	payload defined in file: 'subscriptions/	
	subscription-sample.jsonld'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Create Subscription with Response Status Code set to	
	400 and	
	Create Subscription with Response Body containing the type 'https://	
	uri.etsi.org/ngsi-ld/errors/	
	BadRequestData' and	
	Create Subscription with Response body containing 'title' element	
TP Permutation id	TP/NGSI-LD/CI/SUB/028_03_02	
Name	028_03_02 Emptyld	
Parent Release	v1.3.1	
Description	Check that one cannot create a subscription with an invalid/empty id	
Expected behaviour	Test events	Direction
•	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/subscriptions/	
	method set to 'POST'	

Request Header['Content- Type'] set to 'application/ld+json' and payload defined in file: 'subscriptions/ subscription-sample.jsonId' }	
then { the SUT sends a valid Response for the operations: Create Subscription with Response Status Code set to 400 and Create Subscription with Response Body containing the type 'https:// uri.etsi.org/ngsi-Id/errors/ BadRequestData' and Create Subscription with Response body containing 'title' element }	SUT -> Client

TP ld	TP/NGSI-LD/CI/SUB/028_05	
Test objective	Check that one cannot create a subscription with invalid throttling	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.8.1	
Config Id	CF_01	
PICS Selection	PICS_CreSub	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/SUB/028_05_01	
Name	028_05_01 ThrottlingAndTimeIntervalConjunction	
Parent Release	v1.3.1	
Description	Check that one cannot create a subscription with invalid throttling	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/subscriptions/' method set to 'POST' Create Subscription Request with Header['Content-Type'] set to 'application/ld+json' and payload defined in file: 'subscription- invalid-throttling-timeInterval-sample.jsonId' } then { the SUT sends a valid Response for the operations: Create Subscription From File with Response Status Code set to 400 and Create Subscription From File with Response Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/ BadRequestData' and Create Subscription From File with Response body containing 'title' element	SUT <- Client SUT -> Client
TP Permutation id	TP/NGSI-LD/CI/SUB/028_05_02	
Name	028_05_02 NegativeThrottling	
Parent Release	v1.3.1	
Description	Check that one cannot create a subscription with invalid throttling	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/subscriptions/' method set to 'POST' Create Subscription Request with Header['Content- Type'] set to 'application/ld+json' and payload defined in file: 'subscription- invalid-negative-throttling-sample.jsonld' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Create Subscription From File with Response Status Code set to	SUT -> Client



TP/NGSI-LD/CI/SUB/028_02	
	Direction
when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/subscriptions/' method set to 'POST' Create Subscription Request with Header['Content- Type'] set to 'application/ld+json' and payload defined in file: 'subscription- invalid-json-sample.jsonId' } then { the SUT sends a valid Response for the operations: Create Subscription From File with Response Status Code set to 400 and Create Subscription From File with Response Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/ InvalidRequest' and Create Subscription From File with Response body containing 'title' element	SUT <- Client SUT -> Client
}	
	1
Test events	Direction
when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/subscriptions/' method set to 'POST' Create Subscription Request with Header['Content- Type'] set to 'application/ld+json' and payload defined in file: 'subscription-empty- sample.jsonld' }	SUT <- Client
	Check that one cannot create a subscription with an invalid request ETSI GS CIM 009 V1.5.1 [1], clause 5.8.1 CF_01 PICS_CreSub with { the SUT containing an initial state } Permutations TP:NGSI-LD/CI/SUB/028_02_01 028_02_01 InvalidJson v1.3.1 Check that one cannot create a subscription with an invalid request Test events when { the SUT receives a Request from the client containing:

then {	SUT -> Client
the SUT sends a valid Response for the operations:	
Create Subscription From File with Response Status Code set to	
400 and	
Create Subscription From File with Response Body containing the type	
'https://	
uri.etsi.org/ngsi-ld/errors/	
BadRequestData' and	
Create Subscription From File with Response body containing 'title'	
element	
}	

TP ld	TP/NGSI-LD/CI/SUB/028_04	
Test objective	Check that one cannot create a subscription with an existing id	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.8.1	
Config Id	CF_01	
PICS Selection	PICS_CreSub	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/SUB/028_04_01	
Name	028_04_01 Create a subscription with an id known to the system	
Parent Release	v1.3.1	
Description	Check that one cannot create a subscription with an existing id	
Expected behaviour	Test events	Direction SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/subscriptions/' method set to 'POST' Request Header['Content- Type'] set to 'application/ld+json' and payload defined in file: 'subscriptions/ subscription-sample.jsonId' }	
	then { the SUT sends a valid Response for the operations: Create Subscription with Response Status Code set to 409 and Create Subscription with Response Body containing the type 'https:// uri.etsi.org/ngsi-Id/errors/ AlreadyExists' and Create Subscription with Response body containing 'title' element }	SUT -> Client

4.4.3 Provision

4.4.3.1 TemporalEntity

4.4.3.1.1 UpdateTemporalRepresentationOfEntity

TP Id	TP/NGSI-LD/CI/Prov/TE/008_01
Test objective	Check that one can update a temporal representation of an entity with simple temporal
	properties
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.11
Config Id	CF_01
PICS Selection	PICS_TemEvoEntCre
Initial conditions	with { the SUT containing an initial state }
	Permutations
TP Permutation id	TP/NGSI-LD/CI/Prov/TE/008_01_01
Name	008_01_01 Update a temporal representation of an entity with simple temporal properties
Parent Release	v1.3.1
Description	Check that one can update a temporal representation of an entity with simple temporal
	properties

Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/temporal/entities'	
	method set to 'POST'	
	Create or Update Temporal Representation of Entity Selecting Content	
	Type: and	
	Query Parameter: temporal_entity_representation_id set to '\$	
	{temporal_entity_representation_id}' and	
	Query Parameter: filename set to 'vehicle-	
	create-temporal-representation-sample.jsonld' and	
	Query Parameter: content_type set to 'application/	
	ld+json' and	
	Query Parameter: accept set to '\$	
	{EMPTY}'	
	<u>}</u>	OLIT OF C
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Create Or Update Temporal Representation Of Entity Selecting Content	
	Type with Response Status Code set to 201 and	
	Create Or Update Temporal Representation Of Entity Selecting Content Type with Response Status Code set to	
	204 and	
	Retrieve Temporal Representation Of Entity with Check Updated Entity	
	land	
	Query Parameter: 'updated_resource' set to	
	'temporal_entity_expectation_payload' and	
	Query Parameter: 'response_body' set to 'response2.json()' and	
	Query Parameter: 'ignored_keys' set to 'ignored_attributes'	
	\ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \	
	D.	

4.4.3.1.2 DeleteTemporalRepresentationOfEntity

TP Id	TP/NGSI-LD/CI/Prov/TE/009_01	
Test objective	Check that one can delete a temporal representation of an entity with simple to	emporal
	properties	•
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.16	
Config Id	CF_01	
PICS Selection	PICS_TemEvoEntDel	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing an initial Tem	poral Entity
	\${entity} with an id set to \${temporal_entity_representation_id} }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/TE/009_01_01	
Name	009_01_01 Delete a temporal representation of an entity with simple temporal	properties
Parent Release	v1.3.1	
Description	Check that one can delete a temporal representation of an entity with simple to	emporal
	properties	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}'	
	method set to 'DELETE'	
	Delete Temporal Representation Of Entity With Returning Response	
	with id set to '\$	
	{temporal_entity_representation_id}'	
	}	

then {	SUT -> Client
the SUT sends a valid Response for the operations:	
Delete Temporal Representation Of Entity With Returning Response	
with Response Status Code set to	
204 and	
Retrieve Temporal Representation Of Entity with Response Status	
Code set to	
\${response1.status_code}	
)	

TP Id	TP/NGSI-LD/CI/Prov/TE/009_03	
Test objective	Check that an error is raised if one deletes a temporal entity with a non-existin	g/invalid
	EntityId	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.16	
Config Id	CF_01	
PICS Selection	PICS_TemEvoEntDel	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/TE/009_03_01	
Name	009_03_01 Delete a temporal representation of an entity with an unknown ent	ity id
Parent Release	v1.3.1	
Description	Check that an error is raised if one deletes a temporal entity with a non-existing	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}'	
	method set to 'DELETE'	
	Delete Temporal Representation Of Entity With Returning Response	
	with id set to '\$	
	{temporal_entity_id}'	
	}	OUT OF C
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Delete Temporal Representation Of Entity With Returning Response	
	with Response Status Code set to 404 and	
	1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2 1 2	
	Delete Temporal Representation Of Entity With Returning Response with Response Body containing the type 'https://	
	uri.etsi.org/ngsi-ld/errors/	
	ResourceNotFound' and	
	Delete Temporal Representation Of Entity With Returning Response	
	with Response body containing 'title' element	
	IJ	

T		
TP Id	TP/NGSI-LD/CI/Prov/TE/009_02	
Test objective	Check that an error is raised if one deletes a temporal entity with an empty/inv	alid EntityId
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.16	
Config Id	CF_01	
PICS Selection	PICS_TemEvoEntDel	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/TE/009_02_01	
Name	009_02_01 Delete a temporal representation of an entity with an empty entity	id
Parent Release	v1.3.1	
Description	Check that an error is raised if one deletes a temporal entity with an empty/inv	alid EntityId
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}'	
	method set to 'DELETE'	
	Delete Temporal Representation Of Entity With Returning Response	
	with id set to '\$	

	(EMPTYO)	
	(EMPTY)'	
	than (SUT -> Client
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Delete Temporal Representation Of Entity With Returning Response	
	with Response Status Code set to	
	405 and	
	Delete Temporal Representation Of Entity With Returning Response	
	with Response Body containing the type '\$	
	{EMPTY}' and	
	Delete Temporal Representation Of Entity With Returning Response	
	with Response body containing 'title' element	
	}	
TP Permutation id	TP/NGSI-LD/CI/Prov/TE/009_02_02	
Name	009_02_02 Delete a temporal representation of an entity with an invalid entity	∕ id
Parent Release	v1.3.1	
Description	Check that an error is raised if one deletes a temporal entity with an empty/in	valid EntityId
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	· ·	
	the SUT receives a Request from the client containing: URL set to '/	
	the SUT receives a Request from the client containing: URL set to '/	
	the SUT receives a Request from the client containing:	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'DELETE'	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'DELETE' Delete Temporal Representation Of Entity With Returning Response	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'DELETE'	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'DELETE' Delete Temporal Representation Of Entity With Returning Response with id set to 'invalidId' }	SUT -> Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'DELETE' Delete Temporal Representation Of Entity With Returning Response with id set to 'invalidId' } then {	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'DELETE' Delete Temporal Representation Of Entity With Returning Response with id set to 'invalidId' } then { the SUT sends a valid Response for the operations:	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'DELETE' Delete Temporal Representation Of Entity With Returning Response with id set to 'invalidId' } then { the SUT sends a valid Response for the operations: Delete Temporal Representation Of Entity With Returning Response	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'DELETE' Delete Temporal Representation Of Entity With Returning Response with id set to 'invalidId' } then { the SUT sends a valid Response for the operations: Delete Temporal Representation Of Entity With Returning Response with Response Status Code set to	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'DELETE' Delete Temporal Representation Of Entity With Returning Response with id set to 'invalidId' } then { the SUT sends a valid Response for the operations: Delete Temporal Representation Of Entity With Returning Response with Response Status Code set to 400 and	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'DELETE' Delete Temporal Representation Of Entity With Returning Response with id set to 'invalidId' } then { the SUT sends a valid Response for the operations: Delete Temporal Representation Of Entity With Returning Response with Response Status Code set to 400 and Delete Temporal Representation Of Entity With Returning Response	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'DELETE' Delete Temporal Representation Of Entity With Returning Response with id set to 'invalidId' } then { the SUT sends a valid Response for the operations: Delete Temporal Representation Of Entity With Returning Response with Response Status Code set to 400 and Delete Temporal Representation Of Entity With Returning Response with Response Body containing the type 'https://	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'DELETE' Delete Temporal Representation Of Entity With Returning Response with id set to 'invalidId' } then { the SUT sends a valid Response for the operations: Delete Temporal Representation Of Entity With Returning Response with Response Status Code set to 400 and Delete Temporal Representation Of Entity With Returning Response with Response Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'DELETE' Delete Temporal Representation Of Entity With Returning Response with id set to 'invalidId' } then { the SUT sends a valid Response for the operations: Delete Temporal Representation Of Entity With Returning Response with Response Status Code set to 400 and Delete Temporal Representation Of Entity With Returning Response with Response Body containing the type 'https:// uri.etsi.org/ngsi-Id/errors/ BadRequestData' and	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'DELETE' Delete Temporal Representation Of Entity With Returning Response with id set to 'invalidId' } then { the SUT sends a valid Response for the operations: Delete Temporal Representation Of Entity With Returning Response with Response Status Code set to 400 and Delete Temporal Representation Of Entity With Returning Response with Response Body containing the type 'https:// uri.etsi.org/ngsi-Id/errors/ BadRequestData' and Delete Temporal Representation Of Entity With Returning Response	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities/{temporal_entity_representation_id}' method set to 'DELETE' Delete Temporal Representation Of Entity With Returning Response with id set to 'invalidId' } then { the SUT sends a valid Response for the operations: Delete Temporal Representation Of Entity With Returning Response with Response Status Code set to 400 and Delete Temporal Representation Of Entity With Returning Response with Response Body containing the type 'https:// uri.etsi.org/ngsi-Id/errors/ BadRequestData' and	

4.4.3.1.3 CreateTemporalRepresentationOfEntity

TP Id	TP/NGSI-LD/CI/Prov/TE/007 02	
Test objective	Check that one cannot create a temporal entity with an empty/invalid json/id	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.11	
Config Id	CF_01	
PICS Selection	PICS_TemEvoEntCre	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/TE/007_02_01	
Name	007_02_01 Create a temporal entity with an invalid json	
Parent Release	v1.3.1	
Description	Check that one cannot create a temporal entity with an empty/invalid json/id	
From a set and the allowed according	T (
Expected behaviour	Test events	Direction
Expected benaviour	when {	SUT <- Client
Expected benaviour	when {	
Expected benaviour	when { the SUT receives a Request from the client containing:	
Expected benaviour	when { the SUT receives a Request from the client containing: URL set to '/	
Expected benaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities'	
Expected benaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'POST'	
Expected benaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'POST' Create Temporal Representation of Entity Selecting Content Type: and	
Expected benaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'POST'	
Expected benaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'POST' Create Temporal Representation of Entity Selecting Content Type: and	
Expected benaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'POST' Create Temporal Representation of Entity Selecting Content Type: and Query Parameter: filename set to 'vehicle-temporal-representation-invalid-json-sample.jsonId' and	
Expected benaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'POST' Create Temporal Representation of Entity Selecting Content Type: and Query Parameter: filename set to 'vehicle-temporal-representation-invalid-json-sample.jsonId' and Query Parameter: content_type set to 'application/	
Expected benaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'POST' Create Temporal Representation of Entity Selecting Content Type: and Query Parameter: filename set to 'vehicle-temporal-representation-invalid-json-sample.jsonId' and	

	then { the SUT sends a valid Response for the operations: Create Temporal Representation Of Entity Selecting Content Type with Response Status Code set to 400 and Create Temporal Representation Of Entity Selecting Content Type with Response Body containing the type 'https:// uri.etsi.org/ngsi-Id/errors/ InvalidRequest' and Create Temporal Representation Of Entity Selecting Content Type with Response body containing 'title' element	SUT -> Client
TP Permutation id	TD/NCCLLD/CU/Droy/TE/007, 02, 02	
Name	TP/NGSI-LD/CI/Prov/TE/007_02_02 007_02_02 Create a temporal entity with an empty json	
Parent Release	v1.3.1	
Description	Check that one cannot create a temporal entity with an empty/invalid json/id	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'POST' Create Temporal Representation of Entity Selecting Content Type: and Query Parameter: filename set to 'vehicle- temporal-representation-empty-json-sample.jsonId' and Query Parameter: content_type set to 'application/ Id+json' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Create Temporal Representation Of Entity Selecting Content Type with Response Status Code set to 400 and Create Temporal Representation Of Entity Selecting Content Type with Response Body containing the type 'https:// uri.etsi.org/ngsi-Id/errors/ InvalidRequest' and Create Temporal Representation Of Entity Selecting Content Type with Response body containing 'title' element }	SUT -> Client

Check that one can create a temporal representation of an entity	
ETSI GS CIM 009 V1.5.1 [1], clause 5.6.11	
CF_01	
PICS_TemEvoEntCre	
with { the SUT containing an initial state }	
Permutations	
TP/NGSI-LD/CI/Prov/TE/007_01_01	
007_01_01 Create a temporal representation of an entity	
v1.3.1	
Check that one can create a temporal representation of an entity	
Test events	Direction
when {	SUT <- Client
the SUT receives a Request from the client containing:	
URL set to '/	
ngsi-ld/v1/temporal/entities'	
method set to 'POST'	
Create or Update Temporal Representation of Entity Selecting Content	
Type: and	
Query Parameter: temporal_entity_representation_id set to '\$	
{temporal_entity_representation_id}' and	
Id+json' and	
	CF_01 PICS_TemEvoEntCre with { the SUT containing an initial state } Permutations TP/NGSI-LD/CI/Prov/TE/007_01_01 007_01_01 Create a temporal representation of an entity v1.3.1 Check that one can create a temporal representation of an entity Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'POST' Create or Update Temporal Representation of Entity Selecting Content Type: and Query Parameter: temporal_entity_representation_id set to '\$ {temporal_entity_representation_id}' and Query Parameter: filename set to 'vehicle- create-temporal-representation-sample.jsonId' and Query Parameter: content_type set to 'application/

		1
	Query Parameter: accept set to '\$	
	[EMPTY]'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Create Or Update Temporal Representation Of Entity Selecting Content	
	Type with Response Status Code set to	
	201 and	
	Retrieve Temporal Representation Of Entity with Check Created	
	Resource Set To and	
	Query Parameter: 'ignored_keys' set to '\$	
	{None}' list of keys	
	(Notic) list of keys	
TP Permutation id	TP/NGSI-LD/CI/Prov/TE/007_01_02	
Name	007_01_02 Create a temporal entity with no context	
Parent Release	v1.3.1	
Description		
	Check that one can create a temporal representation of an entity	Dinastian
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/temporal/entities'	
	method set to 'POST'	
	Create or Update Temporal Representation of Entity Selecting Content	
	Type: and	
	Query Parameter: temporal_entity_representation_id set to '\$	
	{temporal_entity_representation_id}' and	
	Query Parameter: filename set to 'vehicle-	
	create-temporal-representation-without-context-sample.jsonld' and	
	Query Parameter: content_type set to 'application/	
	ison' and	
	Query Parameter: accept set to '\$	
	{EMPTY}'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Create Or Update Temporal Representation Of Entity Selecting Content	
	Type with Response Status Code set to	
	201 and	
	Retrieve Temporal Representation Of Entity with Check Created	
	Resource Set To and	
	Query Parameter: 'ignored_keys' set to '\$	
	{None}' list of keys	
)	
	17	

TP ld	TP/NGSI-LD/CI/Prov/TE/007_03	
Test objective	Check that one cannot create a temporal entity with an empty/invalid json/id	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.11	
Config Id	CF_01	
PICS Selection	PICS_TemEvoEntCre	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/TE/007_03_01	
Name	007_03_01 Create a temporal entity with missing id	
Parent Release	v1.3.1	
Description	Check that one cannot create a temporal entity with an invalid @context	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/temporal/entities'	
	method set to 'POST'	
	Create or Update Temporal Representation of Entity Selecting Content	
	Type: and	
	Query Parameter: temporal_entity_representation_id set to '\$	
	{EMPTY}' and	

T		1
	Query Parameter: filename set to 'vehicle-	
	temporal-representation-without-id-sample.jsonld' and	
	Query Parameter: content_type set to 'application/	
	ld+json' and	
	Query Parameter: accept set to '\$	
	{EMPTY}'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operation:	
	Create Or Update Temporal Representation Of Entity Selecting Content	
	Type with Response Status Code set to	
	400	
	\ \	
TP Permutation id	TP/NGSI-LD/CI/Prov/TE/007_03_02	
Name	007_03_02 Create a temporal invalid URI	
Parent Release	v1.3.1	
Description	Check that one cannot create a temporal entity with an invalid @context	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/temporal/entities'	
	method set to 'POST'	
	Create or Update Temporal Representation of Entity Selecting Content	
	Type: and	
	Query Parameter: temporal_entity_representation_id set to 'invalidId' and	
	Query Parameter: filename set to 'vehicle-	
	temporal-representation-sample.jsonld' and	
	Query Parameter: content_type set to 'application/	
	Id+ison' and	
	Query Parameter: accept set to '\$	
	{EMPTY}'	
	I have (SUT -> Client
	then {	SUT -> Client
	the SUT sends a valid Response for the operation:	
	Create Or Update Temporal Representation Of Entity Selecting Content	
	Type with Response Status Code set to	

4.4.3.2 BatchEntities

4.4.3.2.1 CreateBatchOfEntities

TP Id	TP/NGSI-LD/CI/Prov/BE/003_08
Test objective	Check that an HTTP error response of type BadRequestData is raised if the Content-Type header is "application/ld+json" and the request payload body does not contain a @context term
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 6.3.5
Config Id	CF_01
PICS Selection	PICS_HTTP_CONTEXT_RESOLV
Initial conditions	with { the SUT containing an initial state }
	Permutations
TP Permutation id	TP/NGSI-LD/CI/Prov/BE/003_08_01
Name	003_08_01 Create a batch of one entity not containing a JSON-LD @context with a JSON-LD content type
Parent Release	v1.3.1
Description	Check that an HTTP error response of type BadRequestData is raised if the Content-Type header is "application/ld+json" and the request payload body does not contain a @context term

Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entityOperations/create'	
	method set to 'POST'	
	Request Header['Content-	
	Type'] set to 'application/ld+json' and	
	payload set to a list of entities to be created	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Batch Create Entities with Response Status Code set to	
	400 and	
	Batch Create Entities with Response Body containing the type 'https://	
	uri.etsi.org/ngsi-ld/errors/	
	BadRequestData' and	
	Batch Create Entities with Response body containing 'title' element	
	}	

TP ld	TP/NGSI-LD/CI/Prov/BE/003_10	
Test objective	Check that one can create a batch of entities where two have the same id	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.7	
Config Id	CF_01	
PICS Selection	PICS_BatEntCre	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/BE/003_10_01	
Name	003_10_01 Create a batch of three valid entities where two have the same id	
Parent Release	v1.5.1	
Description	Check that one can create a batch of entities where two have the same id	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/entityOperations/create' method set to 'POST' Request Header['Content-Type'] set to " and payload set to a list of entities to be created }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Batch Create Entities with Response Status Code set to 207 and Batch Create Entities with Response body containing batch operation result set to '\$ {expected_batch_operation_result}' }	SUT -> Client

TP ld	TP/NGSI-LD/CI/Prov/BE/003_09
Test objective	Check that an HTTP error response of type BadRequestData is raised if the Content-Type header is "application/ld+json" and a JSON-LD Link header is present in the incoming HTTP request
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 6.3.5
Config Id	CF_01
PICS Selection	PICS_HTTP_CONTEXT_RESOLV
Initial conditions	with { the SUT containing an initial state }
	Permutations
TP Permutation id	TP/NGSI-LD/CI/Prov/BE/003_09_01
Name	003_09_01 Create a batch of one entity with a Link header and a JSON-LD content type
Parent Release	v1.3.1
Description	Check that an HTTP error response of type BadRequestData is raised if the Content-Type header is "application/ld+json" and a JSON-LD Link header is present in the incoming HTTP request

Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entityOperations/create'	
	method set to 'POST'	
	Request Header['Content-	
	Type'] set to 'application/ld+json' and	
	payload set to a list of entities to be created	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Batch Create Entities with Response Status Code set to	
	400 and	
	Batch Create Entities with Response Body containing the type 'https://	
	uri.etsi.org/ngsi-ld/errors/	
	BadRequestData' and	
	Batch Create Entities with Response body containing 'title' element	
	}	

TP ld	TP/NGSI-LD/CI/Prov/BE/003_03	
Test objective	Check that one cannot create a batch of entities with an invalid request	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.7	
Config Id	CF_01	
PICS Selection	PICS_BatEntCre	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/BE/003_03_01	
Name	003_03_01 InvalidJson	
Parent Release	v1.3.1	
Description	Check that one cannot create a batch of entities with an invalid request	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entityOperations/create' method set to 'POST' Batch Entity Delete Request with operation set to 'create', Content- Type set to 'application/ Id+json', and body set to 'batch/invalid-json- sample.jsonId }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Batch Request Entities From File with Response Status Code set to 400 and Batch Request Entities From File with Response Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/ InvalidRequest' and Batch Request Entities From File with Response body containing 'title' element }	SUT -> Client
TP Permutation id	TP/NGSI-LD/CI/Prov/BE/003_03_02	
Name	003_03_02 EmptyJson	
Parent Release	v1.3.1	
Description	Check that one cannot create a batch of entities with an invalid request	

Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entityOperations/create'	
	method set to 'POST'	
	Batch Entity Delete Request with operation set to 'create', Content-	
	Type set to 'application/	
	ld+json', and body set to 'batch/empty-	
	sample.jsonId	
	<u>}</u>	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Batch Request Entities From File with Response Status Code set to	
	400 and	
	Batch Request Entities From File with Response Body containing the	
	type 'https://	
	uri.etsi.org/ngsi-ld/errors/	
	BadRequestData' and	
	Batch Request Entities From File with Response body containing 'title'	
	element	
]}	

TDII	TTD/N G G L TD (G L TD C L TD	1
TP Id	TP/NGSI-LD/CI/Prov/BE/003_01	
Test objective	Check that one can create a batch of entities	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.7	
Config Id	CF_01	
PICS Selection	PICS_BatEntCre	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/BE/003_01_01	
Name	003_01_01 MinimalEntity	
Parent Release	v1.3.1	
Description	Check that one can create a batch of entities	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/entityOperations/create' method set to 'POST' Request Header['Content-Type'] set to " and payload set to a list of entities to be created } then { the SUT sends a valid Response for the operations: Batch Create Entities with Response Status Code set to 201 and Batch Create Entities with Response Body set to an array of created entities ids and Query Entities with Check Created Resource Set To and Query Parameter: 'expected_resources' set to 'entities_to_be_created' and Query Parameter: 'response_body' set to 'response1.json()' and Query Parameter: 'ignored_keys' set to '\$ {None}'	SUT <- Client SUT -> Client
TP Permutation id	TP/NGSI-LD/CI/Prov/BE/003_01_02	
Name	003_01_02 EntityWithSimpleProperties	
Parent Release	v1.3.1	
Description	Check that one can create a batch of entities	

Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ngsi-ld/v1/entityOperations/create'	
	method set to 'POST'	
	Request Header['Content-Type'] set to " and	
	payload set to a list of entities to be created	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	OOT -> Chefit
	Batch Create Entities with Response Status Code set to	
	201 and	
	Batch Create Entities with Response Body set to an array of created	
	entities ids and	
	Query Entities with Check Created Resource Set To and	
	Query Parameter: 'expected_resources' set to	
	'entities_to_be_created' and	
	Query Parameter: 'response_body' set to 'response1.json()' and Query Parameter: 'ignored_keys' set to '\$	
	(None)	
	}	
TP Permutation id	TP/NGSI-LD/CI/Prov/BE/003_01_03	
Name	003_01_03 EntityWithSimpleRelationships	
Parent Release	v1.3.1	
Description	Check that one can create a batch of entities	1
Expected behaviour	Test events	Direction SUT <- Client
	when { the SUT receives a Request from the client containing:	SUT <- Client
	URL set to '/ngsi-ld/v1/entityOperations/create'	
	method set to 'POST'	
	Request Header['Content-Type'] set to " and	
	payload set to a list of entities to be created	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Batch Create Entities with Response Status Code set to	
	201 and Batch Create Entities with Response Body set to an array of created	
	entities ids and	
	Query Entities with Check Created Resource Set To and	
	Query Parameter: 'expected_resources' set to	
	'entities_to_be_created' and	
	Query Parameter: 'response_body' set to 'response1.json()' and	
	Query Parameter: 'ignored_keys' set to '\$	
	{None}'	
TP Permutation id		
Name	TP/NGSI-LD/CI/Prov/BE/003_01_04 003_01_04 EntityWithRelationshipsProperties	
Parent Release	v1.3.1	
Description	Check that one can create a batch of entities	
Expected behaviour	Test events	Direction
•	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ngsi-ld/v1/entityOperations/create'	
	method set to 'POST'	
	Request Header['Content-Type'] set to " and	
	payload set to a list of entities to be created	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	301 -> Ciletit
	Batch Create Entities with Response Status Code set to	
	201 and	
	Batch Create Entities with Response Body set to an array of created	
	entities ids and	
	Query Entities with Check Created Resource Set To and	
	Query Parameter: 'expected_resources' set to	
	'entities_to_be_created' and Query Parameter: 'response_body' set to 'response1.json()' and	
		1

Query Parameter: 'ignored_keys' set to '\$	
{None}'	
}	

TDIA	TD/NOOLL D/OVD/DE/000 OF	
TP Id	TP/NGSI-LD/CI/Prov/BE/003_05	- 4: /: II I
Test objective	Check that the default @context is used if the Content-Type header is "applicated by the Link has deep deep and applicated by the Link has deep deep and applicated by the Link has deep deep and applicated by the Link has deep deep and the Link has deep deep deep deep deep deep deep dee	ation/json" and
D	the Link header does not contain a JSON-LD @context	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 6.3.5	
Config Id	CF_01	
PICS Selection	PICS_HTTP_CONTEXT_RESOLV	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/BE/003_05_01	
Name	003_05_01 Create a batch of one entity using the default context with JSON of	ontent type
Parent Release	v1.3.1	
Description	Check that the default @context is used if the Content-Type header is "applicate the Link header does not contain a JSON-LD @context and retrieve the entity context."	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entityOperations/create' method set to 'POST' Request Header['Content- Type'] set to 'application/json' and payload set to a list of entities to be created	SUT <- Client
	then { the SUT sends a valid Response for the operations: Batch Create Entities with Response Status Code set to 201 and Retrieve Entity by Id with Check Response Body containing an Attribute set to and	SUT -> Client
	}	
TP Permutation id	TP/NGSI-LD/CI/Prov/BE/003_05_02	
Name	003_05_02 Create a batch of one entity using the default context with JSON of	ontent type
Parent Release	v1.3.1	
Description	Check that the default @context is used if the Content-Type header is "applicate the Link header does not contain a JSON-LD @context and retrieve the entity	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entityOperations/create' method set to 'POST' Request Header['Content- Type'] set to 'application/json' and payload set to a list of entities to be created }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Batch Create Entities with Response Status Code set to 201 and Retrieve Entity by Id with Check Response Body containing an Attribute set to and Query Parameter: expected_attribute_name set to 'ngsi-Id:default-context/almostFull' and Query Parameter: response_body set to 'response1.json()'	SUT -> Client

TP ld	TP/NGSI-LD/CI/Prov/BE/003_07	
Test objective	Check that the @context is obtained from the request payload body itself if the	Content-
	Type header is "application/ld+json"	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 6.3.5	
Config Id	CF 01	
PICS Selection	PICS_HTTP_CONTEXT_RESOLV	
Initial conditions	with { the SUT containing an initial state }	
initial conditions	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/BE/003_07_01	
Name	003_07_01 Create a batch of one entity using a JSON-LD @context obtained	from the
Name	request payload	nom me
Parent Release	v1.3.1	
Description		Contont
Description	Check that the @context is obtained from the request payload body itself if the	
	Type header is "application/ld+json" and the response attribute should be com	pacted as one
Formanda di bahandana	used the same context as provided when creating the entity	Discotion
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entityOperations/create'	
	method set to 'POST'	
	Request Header['Content-	
	Type'] set to 'application/ld+json' and	
	payload set to a list of entities to be created	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Batch Create Entities with Response Status Code set to	
	201 and	
	Retrieve Entity by Id with Check Response Body containing an Attribute	
	set to and	
	Query Parameter: expected_attribute_name set to 'almostFull' and	
	Query Parameter: response_body set to 'response1.json()'	
	}	
TP Permutation id	TP/NGSI-LD/CI/Prov/BE/003_07_02	
Name	003_07_02 Create a batch of one entity using a JSON-LD @context obtained	from the
	request payload	
Parent Release	v1.3.1	
Description	Check that the @context is obtained from the request payload body itself if the	Content-
	Type header is "application/ld+json" and response attribute should not be com	pacted as one
	did not provide a context containing this term	
Expected behaviour	Test events	Direction
-	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entityOperations/create'	
	method set to 'POST'	
	Request Header['Content-	
	Type'] set to 'application/ld+json' and	
	payload set to a list of entities to be created	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Batch Create Entities with Response Status Code set to	
	201 and	
	Retrieve Entity by Id with Check Response Body containing an Attribute	
	set to and	
	Query Parameter: expected_attribute_name set to 'https://	
		l l
	Ingsi-Id-test-suite/context#almostFull' and	
	ngsi-ld-test-suite/context#almostFull' and Query Parameter: response body set to 'response1.ison()'	
	Ingsi-Id-test-suite/context#almostFull' and Query Parameter: response_body set to 'response1.json()' }	

TP ld	TP/NGSI-LD/CI/Prov/BE/003_02
Test objective	Check that one can create a batch of entities where some will succeed and others will fail
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.7
Config Id	CF 01

PICS Selection	PICS_BatEntCre	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/BE/003_02_01	
Name	003_02_01 Create a batch of two valid entities and one invalid entity	
Parent Release	v1.3.1	
Description	Check that one can create a batch of two valid entities and one invalid entity	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/entityOperations/create' method set to 'POST' Request Header['Content-Type'] set to " and payload set to a list of entities to be created } then {	SUT <- Client SUT -> Client
	the SUT sends a valid Response for the operations: Batch Create Entities with Response Status Code set to 207 and Batch Create Entities with Response body containing batch operation result set to '\$ {expected_batch_operation_result}' and Query Entities with Check Created Resource Set To and Query Parameter: 'ignored_keys' set to '\$ {None}' }	

TP ld	TP/NGSI-LD/CI/Prov/BE/003_06	
Test objective	Check that an HTTP error response of type BadRequestData is raised if the Content-Type header is "application/json" and the request payload body (as JSON) contains a "@context" term	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 6.3.5	
Config Id	CF_01	
PICS Selection	PICS_HTTP_CONTEXT_RESOLV	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/BE/003_06_01	
Name	003_06_01 Create a batch of one entity containing a JSON-LD @context with content type	a JSON
Parent Release	v1.3.1	
Description	Check that an HTTP error response of type BadRequestData is raised if the Content-Type header is "application/json" and the request payload body (as JSON) contains a "@context" term	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entityOperations/create' method set to 'POST' Request Header['Content- Type'] set to 'application/json' and payload set to a list of entities to be created }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Batch Create Entities with Response Status Code set to 400 and Batch Create Entities with Response Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/ BadRequestData' and Batch Create Entities with Response body containing 'title' element }	SUT -> Client

TP ld	TP/NGSI-LD/CI/Prov/BE/003_04		
Test objective	Check that the @context is obtained from a Link Header if the Content-Type h "application/json"	eader is	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 6.3.5		
Config Id	CF_01		
PICS Selection	PICS_HTTP_CONTEXT_RESOLV		
Initial conditions	with { the SUT containing an initial state }		
	Permutations		
TP Permutation id	TP/NGSI-LD/CI/Prov/BE/003_04_01		
Name	003_04_01 Create a batch of one entity using a provided Link header with JS0	ON content	
Parent Release	type and retrieve the entity with context detail v1.3.1		
Description	Check that the @context is obtained from a Link Header if the Content-Type h	eader is	
E	"application/json"	l Bi ii	
Expected behaviour	Test events	Direction	
	when { the SUT receives a Request from the client containing: URL set to '/	SUT <- Client	
	ngsi-ld/v1/entityOperations/create' method set to 'POST' Request Header['Content-		
	Type'] set to 'application/json' and		
	payload set to a list of entities to be created		
	then {	SUT -> Client	
	the SUT sends a valid Response for the operations:		
	Batch Create Entities with Response Status Code set to		
	201 and		
	Retrieve Entity by Id with Check Response Body containing an Attribute		
	set to and		
	Query Parameter: expected_attribute_name set to 'almostFull' and		
	Query Parameter: response_body set to 'response1.json()'		
	}		
TP Permutation id	TP/NGSI-LD/CI/Prov/BE/003_04_02		
Name	003_04_02 Create a batch of one entity using a provided Link header with JS0	ON content	
	type and retrieve the entity without context detail		
Parent Release	v1.3.1		
Description	Check that the @context is obtained from a Link Header if the Content-Type header is "application/json"		
Expected behaviour	Test events	Direction	
	when {	SUT <- Client	
	the SUT receives a Request from the client containing:		
	URL set to '/		
	ngsi-ld/v1/entityOperations/create'		
	method set to 'POST'		
	Request Header['Content-		
	Type'] set to 'application/json' and		
	payload set to a list of entities to be created		
	then {	SUT -> Client	
	the SUT sends a valid Response for the operations:		
	Batch Create Entities with Response Status Code set to		
	201 and		
	Retrieve Entity by Id with Check Response Body containing an Attribute		
	set to and		
	Query Parameter: expected_attribute_name set to 'https://		
	ngsi-ld-test-suite/context#almostFull' and		
	Query Parameter: response_body set to 'response1.json()'		
	1}	Ì	

4.4.3.2.2 DeleteBatchOfEntities

TP Id	TP/NGSI-LD/CI/Prov/BE/006_02
Test objective	Check that one can delete a batch of entities where some will succeed and others will fail
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.10
Config Id	CF_01

PICS Selection	PICS_BatEntDel	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing an initial Entit	y \${entity} with
	an id set to \${entityId} }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/BE/006_02_01	
Name	006_02_01 Delete a batch of non-existing and existing entities	
Parent Release	v1.3.1	
Description	Check that one can delete a batch of non-existing and existing entities	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entityOperations/delete'	
	method set to 'POST'	
	Batch Delete Entities: and	
	Query Parameter: entities_ids_to_be_deleted set to	
	['@{entities_ids_to_be_deleted}'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Batch Delete Entities with Response Status Code set to	
	207 and	
	\${response.json()}	
	with Response body containing batch operation result set to '\$	
	{response1}' and	
	Query Entities with Response body is empty	
	}	

TP ld	TP/NGSI-LD/CI/Prov/BE/006_04	
Test objective	Check that one can delete a batch of entities with the same id	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.10	
Config Id	CF_01	
PICS Selection	PICS_BatEntDel	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing an initial Entity an id set to \${entityId} }	y \${entity} with
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/BE/006_04_01	
Name	006_04_01 Delete a batch of existing entities with the same id	
Parent Release	v1.5.1	
Description	Check that one can delete a batch entities with the same id	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entityOperations/delete' method set to 'POST' Batch Delete Entities: and Query Parameter: entities_ids_to_be_deleted set to '@{entities_ids_to_be_deleted}' } then { the SUT sends a valid Response for the operations:	SUT <- Client
	Batch Delete Entities with Response Status Code set to 207 and \${response.json()} with Response body containing batch operation result set to '\$ {response1}' }	

TP Id	TP/NGSI-LD/CI/Prov/BE/006_01
Test objective	Check that one can delete a batch of entities
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.10
Config Id	CF_01
PICS Selection	PICS_BatEntDel
Initial conditions	with { the SUT being in the "initial state" and and containing a list of entities }

Permutations		
TP Permutation id	TP/NGSI-LD/CI/Prov/BE/006_01_01	
Name	006_01_01 Delete a batch of entities	
Parent Release	v1.3.1	
Description	Check that one can delete a batch of entities	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entityOperations/delete' method set to 'POST' Batch Delete Entities: and Query Parameter: entities_ids_to_be_deleted set to '@{entities_ids_to_be_deleted}' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Batch Delete Entities with Response Status Code set to 204 and Query Entities with Response body is empty }	SUT -> Client

TP ld	TP/NGSI-LD/CI/Prov/BE/006_03	
Test objective	Check that one cannot delete a batch of entities with an invalid request	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.10	
Config Id	CF_01	
PICS Selection	PICS BatEntDel	
Initial conditions	with { the SUT containing an initial state }	
ilitiai conditions	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/BE/006_03_01	
Name	006_03_01 InvalidJson	
Parent Release	v1.3.1	
Description	Check that one cannot delete a batch of entities with an invalid request	
Expected behaviour	Test events	Direction
Expected benaviour		SUT <- Client
	when { the SUT receives a Request from the client containing:	SUT <- CHELL
	URL set to '/	
	ngsi-ld/v1/entityOperations/delete'	
	method set to 'POST'	
	Batch Entity Delete Request with operation set to 'delete', Content-	
	Type set to 'application/	
	ld+json', and body set to 'batch/invalid-json-	
	sample.jsonId	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	00.70
	Batch Request Entities From File with Response Status Code set to	
	400 and	
	Batch Request Entities From File with Response Body containing the	
	type 'https://	
	uri.etsi.org/ngsi-ld/errors/	
	InvalidRequest' and	
	Batch Request Entities From File with Response body containing 'title'	
	element	
	}	
TP Permutation id	TP/NGSI-LD/CI/Prov/BE/006_03_02	
Name	006_03_02 EmptyJson	
Parent Release	v1.3.1	
Description	Check that one cannot delete a batch of entities with an invalid request	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entityOperations/delete'	
	method set to 'POST'	
	Batch Entity Delete Request with operation set to 'delete', Content-	

Type set to 'application/ Id+json', and body set to 'batch/empty- sample.jsonId	
then { the SUT sends a valid Response for the operations: Batch Request Entities From File with Response Status Code set to 400 and Batch Request Entities From File with Response Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/ BadRequestData' and Batch Request Entities From File with Response body containing 'title' element }	SUT -> Client

4.4.3.2.3 UpsertBatchOfEntities

TP Id	TP/NGSI-LD/CI/Prov/BE/004_06	
Test objective	Check that one cannot upsert a batch of entities with an invalid request	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.8	
Config Id	CF_01	
PICS Selection	PICS_BatEntCreUpdUps	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/BE/004_06_01	
Name	004_06_01 InvalidJson	
Parent Release	v1.3.1	
Description	Check that one cannot upsert a batch of entities with an invalid request	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entityOperations/upsert' method set to 'POST' Batch Entity Delete Request with operation set to 'upsert', Content-Type set to 'application/ Id+json', and body set to 'batch/invalid-json-sample.jsonId } then { the SUT sends a valid Response for the operations: Batch Request Entities From File with Response Status Code set to 400 and Batch Request Entities From File with Response Body containing the type 'https:// uri.etsi.org/ngsi-Id/errors/ InvalidRequest' and Batch Request Entities From File with Response body containing 'title' element	SUT <- Client SUT -> Client
TP Permutation id	TP/NGSI-LD/CI/Prov/BE/004_06_02	
Name	004_06_02 InvalidJsonLd	
Parent Release	V1.3.1	
Description	Check that one cannot upsert a batch of entities with an invalid request	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entityOperations/upsert' method set to 'POST' Batch Entity Delete Request with operation set to 'upsert', Content-Type set to 'application/ Id+json', and body set to 'batch/invalid-json- Id-sample.jsonId }	SUT <- Client

then {	SUT -> Client
the SUT sends a valid Response for the operations:	
Batch Request Entities From File with Response Status Code set to	
400 and	
Batch Request Entities From File with Response Body containing the	
type 'https://	
uri.etsi.org/ngsi-ld/errors/	
BadRequestData' and	
Batch Request Entities From File with Response body containing 'title'	
element	
}	

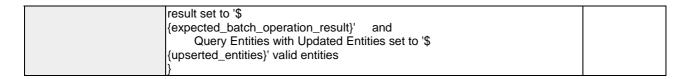
TP ld	TP/NGSI-LD/CI/Prov/BE/004_04		
Test objective	Check that one can upsert a batch of entities with update option		
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.8		
Config Id	CF 01		
PICS Selection			
Initial conditions	PICS_BatEntCreUpdUps with { the SUT being in the "initial state" and the SUT containing an initial Entit	v Clantity) with	
initial conditions		y \${entity} with	
	an id set to \${entityId} }		
TD Dames delice id	Permutations		
TP Permutation id	TP/NGSI-LD/CI/Prov/BE/004_04_01		
Name	004_04_01 EntityWithSimpleProperties		
Parent Release	v1.3.1		
Description	Check that one can upsert a batch of entities with update option	I 5: .:	
Expected behaviour	Test events	Direction	
	when {	SUT <- Client	
	the SUT receives a Request from the client containing:		
	URL set to '/		
	ngsi-ld/v1/entityOperations/upsert?options=\${update_option}' method set to 'POST'		
	Request batch upsert operation over entity from filename		
	'@{entities_to_be_upserted}' with update_option set to 'update' and Content-		
	Type set to 'application/ld+json'		
	Type set to application/lu+json		
	than (SUT -> Client	
	then { the SUT sends a valid Response for the operations:	301 -> Client	
	Batch Upsert Entities with Response Status Code set to		
	201 and		
	Batch Upsert Entities with Response Body set to an array of created		
	entities ids and		
	Query Entities with Updated Entities set to '\$		
	{updated_entities}' valid entities		
	}		
TP Permutation id	TP/NGSI-LD/CI/Prov/BE/004_04_02	l	
Name	004_04_02 EntityWithSimpleRelationships		
Parent Release	v1.3.1		
Description	Check that one can upsert a batch of entities with update option		
Expected behaviour	Test events	Direction	
zapostou zonatiou.	when {	SUT <- Client	
	the SUT receives a Request from the client containing:	OOT CHOIL	
	URL set to '/		
	ngsi-ld/v1/entityOperations/upsert?options=\${update_option}'		
	method set to 'POST'		
	Request batch upsert operation over entity from filename		
	'@{entities_to_be_upserted}' with update_option set to 'update' and Content-		
	Type set to 'application/ld+json'		
	}		
	then {	SUT -> Client	
	the SUT sends a valid Response for the operations:		
	Batch Upsert Entities with Response Status Code set to		
	201 and		
	Batch Upsert Entities with Response Body set to an array of created		
	entities ids and		
	Query Entities with Updated Entities set to '\$		
	{updated_entities}' valid entities		
	· · · · · · · · · · · · · · · · · · ·		
	- 17		

TP Permutation id	TP/NGSI-LD/CI/Prov/BE/004_04_03	
Name	004_04_03 EntityWithRelationshipsProperties	
Parent Release	v1.3.1	
Description	Check that one can upsert a batch of entities with update option	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entityOperations/upsert?options=\${update_option}' method set to 'POST' Request batch upsert operation over entity from filename '@{entities_to_be_upserted}' with update_option set to 'update' and Content- Type set to 'application/ld+json' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Batch Upsert Entities with Response Status Code set to 201 and Batch Upsert Entities with Response Body set to an array of created entities ids and Query Entities with Updated Entities set to '\$ {updated_entities}' valid entities }	SUT -> Client

TP Id	TP/NGSI-LD/CI/Prov/BE/004_02	
Test objective	Check that one can upsert a batch of non-existing and existing entities when	e non-existing
	will be created and existing will be replaced	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.8	
Config Id	CF_01	
PICS Selection	PICS_BatEntCreUpdUps	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/BE/004_02_01	
Name	004_02_01 EntityWithSimpleProperties	
Parent Release	v1.3.1	
Description	Check that one can upsert a batch of non-existing and existing entities	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entityOperations/upsert?options=\${update_option}'	
	method set to 'POST'	
	Request batch upsert operation over entity from filename	
	'@{entities_to_be_upserted}' with update_option set to 'replace' and	
	Content-	
	Type set to 'application/ld+json'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Batch Upsert Entities with Response Status Code set to	
	201 and	
	Batch Upsert Entities with Response Body set to an array of created	
	entities ids and	
	Query Entities with Updated Entities set to '\$	
	{entities_to_be_upserted}' valid entities	
TP Permutation id	TP/NGSI-LD/CI/Prov/BE/004_02_02	
Name	004_02_02 EntityWithSimpleRelationships	
Parent Release	v1.3.1	
Description	Check that one can upsert a batch of non-existing and existing entities	
	Test events	Direction
Expected behaviour	1303 3 13113	SUT <- Client
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ ngsi-ld/v1/entityOperations/upsert?options=\${update_option}'	
	method set to 'POST'	

	Request batch upsert operation over entity from filename	
	@{entities_to_be_upserted}' with update_option set to 'replace' and	
	Content- Type set to 'application/ld+json'	
	Type set to application/lu+json	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	COT FORCIN
	Batch Upsert Entities with Response Status Code set to	
	201 and	
	Batch Upsert Entities with Response Body set to an array of created	
	entities ids and	
	Query Entities with Updated Entities set to '\$	
	{entities_to_be_upserted}' valid entities	
	}	
TP Permutation id	TP/NGSI-LD/CI/Prov/BE/004_02_03	
Name	004_02_03 EntityWithRelationshipsProperties	
Parent Release	v1.3.1	
Description	Check that one can upsert a batch of non-existing and existing entities	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entityOperations/upsert?options=\${update_option}'	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entityOperations/upsert?options=\${update_option}' method set to 'POST'	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entityOperations/upsert?options=\${update_option}' method set to 'POST' Request batch upsert operation over entity from filename	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entityOperations/upsert?options=\${update_option}' method set to 'POST' Request batch upsert operation over entity from filename '@{entities_to_be_upserted}' with update_option set to 'replace' and	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entityOperations/upsert?options=\${update_option}' method set to 'POST' Request batch upsert operation over entity from filename '@{entities_to_be_upserted}' with update_option set to 'replace' and Content-	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entityOperations/upsert?options=\${update_option}' method set to 'POST' Request batch upsert operation over entity from filename '@{entities_to_be_upserted}' with update_option set to 'replace' and	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entityOperations/upsert?options=\${update_option}' method set to 'POST' Request batch upsert operation over entity from filename '@{entities_to_be_upserted}' with update_option set to 'replace' and Content- Type set to 'application/ld+json' }	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entityOperations/upsert?options=\${update_option}' method set to 'POST' Request batch upsert operation over entity from filename '@{entities_to_be_upserted}' with update_option set to 'replace' and Content-	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entityOperations/upsert?options=\${update_option}' method set to 'POST' Request batch upsert operation over entity from filename '@{entities_to_be_upserted}' with update_option set to 'replace' and Content- Type set to 'application/ld+json' } then {	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entityOperations/upsert?options=\${update_option}' method set to 'POST' Request batch upsert operation over entity from filename '@{entities_to_be_upserted}' with update_option set to 'replace' and Content- Type set to 'application/ld+json' } then { the SUT sends a valid Response for the operations: Batch Upsert Entities with Response Status Code set to 201 and	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entityOperations/upsert?options=\${update_option}' method set to 'POST' Request batch upsert operation over entity from filename '@{entities_to_be_upserted}' with update_option set to 'replace' and Content- Type set to 'application/ld+json' } then { the SUT sends a valid Response for the operations: Batch Upsert Entities with Response Status Code set to 201 and Batch Upsert Entities with Response Body set to an array of created	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entityOperations/upsert?options=\${update_option}' method set to 'POST' Request batch upsert operation over entity from filename '@{entities_to_be_upserted}' with update_option set to 'replace' and Content- Type set to 'application/ld+json' } then { the SUT sends a valid Response for the operations: Batch Upsert Entities with Response Status Code set to 201 and Batch Upsert Entities with Response Body set to an array of created entities ids and	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entityOperations/upsert?options=\${update_option}' method set to 'POST' Request batch upsert operation over entity from filename '@{entities_to_be_upserted}' with update_option set to 'replace' and Content- Type set to 'application/ld+json' } then { the SUT sends a valid Response for the operations: Batch Upsert Entities with Response Status Code set to 201	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entityOperations/upsert?options=\${update_option}' method set to 'POST' Request batch upsert operation over entity from filename '@{entities_to_be_upserted}' with update_option set to 'replace' and Content- Type set to 'application/ld+json' } then { the SUT sends a valid Response for the operations: Batch Upsert Entities with Response Status Code set to 201 and Batch Upsert Entities with Response Body set to an array of created entities ids and	

· ·		
TP Id	TP/NGSI-LD/CI/Prov/BE/004_05	
Test objective	Check that one can upsert a batch of entities where some will succeed and other	ners will fail
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.8	
Config Id	CF_01	
PICS Selection	PICS_BatEntCreUpdUps	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/BE/004_05_01	
Name	004_05_01 Upsert a batch of two valid entities and one invalid entity	
Parent Release	v1.3.1	
Description	Check that one can upsert a batch of two valid entities and one invalid entity	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entityOperations/upsert?options=\${update_option}'	
	method set to 'POST'	
	Request batch upsert operation over entity from filename	
	'@{entities_to_be_upserted}' with update_option set to 'replace' and	
	Content-	
	Type set to 'application/ld+json'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Batch Upsert Entities with Response Status Code set to	
	207 and	
	Batch Upsert Entities with Response body containing batch operation	



Υ			
TP ld	TP/NGSI-LD/CI/Prov/BE/004_07		
Test objective	Check that one can upsert a batch of entities where two have the same id		
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.8		
Config Id	CF_01		
PICS Selection	PICS_BatEntCreUpdUps		
Initial conditions	with { the SUT containing an initial state }		
Permutations			
TP Permutation id	TP/NGSI-LD/CI/Prov/BE/004_07_01		
Name	004_07_01 Upsert a batch of three valid entities where two have the same id		
Parent Release	v1.5.1		
Description	Check that one can upsert a batch of where two have the same id		
Expected behaviour	Test events	Direction	
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entityOperations/upsert?options=\${update_option}' method set to 'POST' Request batch upsert operation over entity from filename '@{entities_to_be_upserted}' with update_option set to 'replace' and Content- Type set to 'application/ld+json' }	SUT <- Client	
	then { the SUT sends a valid Response for the operations: Batch Upsert Entities with Response Status Code set to 201 and Batch Upsert Entities with Response Body set to an array of created entities ids }	SUT -> Client	

TP ld	TP/NGSI-LD/CI/Prov/BE/004_03		
Test objective	Check that one can upsert a batch of existing entities and they will be repla	aced	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.8		
Config Id	CF_01		
PICS Selection	PICS_BatEntCreUpdUps		
Initial conditions	with { the SUT being in the "initial state" and and containing a list of entities	; }	
Permutations			
TP Permutation id	TP/NGSI-LD/CI/Prov/BE/004_03_01		
Name	004_03_01 EntityWithSimpleProperties		
Parent Release	v1.3.1		
Description	Check that one can upsert a batch of existing entities		
Expected behaviour	Test events	Direction	
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entityOperations/upsert?options=\${update_option}' method set to 'POST' Request batch upsert operation over entity from filename '@{entities_to_be_upserted}' with update_option set to 'replace' and Content- Type set to 'application/ld+json' }	SUT <- Client	
	then { the SUT sends a valid Response for the operations: Batch Upsert Entities with Response Status Code set to 204 and Query Entities with Updated Entities set to '\$ {entities_to_be_upserted}' valid entities }	SUT -> Client	

TP Permutation id	TP/NGSI-LD/CI/Prov/BE/004_03_02	
Name	004_03_02 EntityWithSimpleRelationships	
Parent Release	v1.3.1	
Description	Check that one can upsert a batch of existing entities	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entityOperations/upsert?options=\${update_option}' method set to 'POST' Request batch upsert operation over entity from filename '@{entities_to_be_upserted}' with update_option set to 'replace' and Content- Type set to 'application/ld+json' } then { the SUT sends a valid Response for the operations: Batch Upsert Entities with Response Status Code set to 204 and	SUT <- Client SUT -> Client
TP Permutation id	Query Entities with Updated Entities set to '\$ {entities_to_be_upserted}' valid entities } TP/NGSI-LD/CI/Prov/BE/004_03_03	
Name	004_03_03 EntityWithRelationshipsProperties	
Parent Release	v1.3.1	
Description	Check that one can upsert a batch of existing entities	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entityOperations/upsert?options=\${update_option}' method set to 'POST' Request batch upsert operation over entity from filename '@{entities_to_be_upserted}' with update_option set to 'replace' and Content- Type set to 'application/Id+json' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Batch Upsert Entities with Response Status Code set to 204 and Query Entities with Updated Entities set to '\$ {entities_to_be_upserted}' valid entities }	SUT -> Client

TP Id	TP/NGSI-LD/CI/Prov/BE/004_01	
Test objective	Check that one can upsert a batch of non-existing entities and they will be created	ated
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.8	
Config Id	CF_01	
PICS Selection	PICS_BatEntCreUpdUps	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/BE/004_01_01	
Name	004_01_01 EntityWithSimpleProperties	
Parent Release	v1.3.1	
Description	Check that one can upsert a batch of non-existing entities	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entityOperations/upsert?options=\${update_option}'	
	method set to 'POST'	
	Request batch upsert operation over entity from filename	
	'@{entities_to_be_upserted}' with update_option set to 'replace' and	
	Content-	

the CLIT conde a valid Deanging for the appretional	> Client
the SUT sends a valid Response for the operations:	
Batch Upsert Entities with Response Status Code set to	
201 and	
Batch Upsert Entities with Response Body set to an array of created	
entities ids and	
Query Entities with Updated Entities set to '\$	
{entities_to_be_upserted}' valid entities	
}	
P Permutation id TP/NGSI-LD/CI/Prov/BE/004_01_02	
ame 004_01_02 EntityWithSimpleRelationships	
arent Release v1.3.1	
escription Check that one can upsert a batch of non-existing entities	
	ction
	 Client
the SUT receives a Request from the client containing:	
URL set to '/	
ngsi-ld/v1/entityOperations/upsert?options=\${update_option}'	
method set to 'POST'	
Request batch upsert operation over entity from filename	
'@{entities_to_be_upserted}' with update_option set to 'replace' and	
Content-	
Type set to 'application/ld+json'	
}	
	> Client
the SUT sends a valid Response for the operations:	
Batch Upsert Entities with Response Status Code set to	
201 and	
Batch Upsert Entities with Response Body set to an array of created	
Batch Upsert Entities with Response Body set to an array of created entities ids and	
Batch Upsert Entities with Response Body set to an array of created entities ids and Query Entities with Updated Entities set to '\$	
Batch Upsert Entities with Response Body set to an array of created entities ids and	
Batch Upsert Entities with Response Body set to an array of created entities ids and Query Entities with Updated Entities set to '\$ {entities_to_be_upserted}' valid entities }	
Batch Upsert Entities with Response Body set to an array of created entities ids and Query Entities with Updated Entities set to '\$ {entities_to_be_upserted}' valid entities } P Permutation id TP/NGSI-LD/CI/Prov/BE/004_01_03	
Batch Upsert Entities with Response Body set to an array of created entities ids and Query Entities with Updated Entities set to '\$ {entities_to_be_upserted}' valid entities } P Permutation id TP/NGSI-LD/CI/Prov/BE/004_01_03 ame 004_01_03 EntityWithRelationshipsProperties	
Batch Upsert Entities with Response Body set to an array of created entities ids and Query Entities with Updated Entities set to '\$ {entities_to_be_upserted}' valid entities } P Permutation id TP/NGSI-LD/CI/Prov/BE/004_01_03 ame 004_01_03 EntityWithRelationshipsProperties arent Release v1.3.1	
Batch Upsert Entities with Response Body set to an array of created entities ids and Query Entities with Updated Entities set to '\$ {entities_to_be_upserted}' valid entities } P Permutation id TP/NGSI-LD/CI/Prov/BE/004_01_03 ame 004_01_03 EntityWithRelationshipsProperties arent Release v1.3.1 escription Check that one can upsert a batch of non-existing entities	ction
Batch Upsert Entities with Response Body set to an array of created entities ids and Query Entities with Updated Entities set to '\$ {entities_to_be_upserted}' valid entities } P Permutation id TP/NGSI-LD/CI/Prov/BE/004_01_03 ame 004_01_03 EntityWithRelationshipsProperties arent Release v1.3.1 escription Check that one can upsert a batch of non-existing entities xpected behaviour Test events Dire	ction
Batch Upsert Entities with Response Body set to an array of created entities ids and Query Entities with Updated Entities set to '\$ {entities_to_be_upserted}' valid entities } P Permutation id TP/NGSI-LD/CI/Prov/BE/004_01_03 ame 004_01_03 EntityWithRelationshipsProperties arent Release v1.3.1 escription Check that one can upsert a batch of non-existing entities xpected behaviour Test events Dire when {	ction - Client
Batch Upsert Entities with Response Body set to an array of created entities ids and Query Entities with Updated Entities set to '\$ {entities_to_be_upserted}' valid entities } P Permutation id TP/NGSI-LD/CI/Prov/BE/004_01_03 ame 004_01_03 EntityWithRelationshipsProperties arent Release v1.3.1 escription Check that one can upsert a batch of non-existing entities xpected behaviour Test events Dire when { SUT <	
Batch Upsert Entities with Response Body set to an array of created entities ids and Query Entities with Updated Entities set to '\$ {entities_to_be_upserted}' valid entities } P Permutation id TP/NGSI-LD/CI/Prov/BE/004_01_03 ame 004_01_03 EntityWithRelationshipsProperties arent Release v1.3.1 escription Check that one can upsert a batch of non-existing entities xpected behaviour Test events Dire when { SUT receives a Request from the client containing: URL set to '/	
Batch Upsert Entities with Response Body set to an array of created entities ids and Query Entities with Updated Entities set to '\$ {entities_to_be_upserted}' valid entities } P Permutation id TP/NGSI-LD/CI/Prov/BE/004_01_03 ame 004_01_03 EntityWithRelationshipsProperties v1.3.1 escription Check that one can upsert a batch of non-existing entities xpected behaviour Test events Dire when { SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entityOperations/upsert?options=\${update_option}'	
Batch Upsert Entities with Response Body set to an array of created entities ids and Query Entities with Updated Entities set to '\$ {entities_to_be_upserted}' valid entities } P Permutation id TP/NGSI-LD/CI/Prov/BE/004_01_03 ame 004_01_03 EntityWithRelationshipsProperties arent Release v1.3.1 escription Check that one can upsert a batch of non-existing entities xpected behaviour Test events Dire when { SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entityOperations/upsert?options=\${update_option}' method set to 'POST'	
Batch Upsert Entities with Response Body set to an array of created entities ids and Query Entities with Updated Entities set to '\$ {entities_to_be_upserted}' valid entities } P Permutation id TP/NGSI-LD/CI/Prov/BE/004_01_03 ame 004_01_03 EntityWithRelationshipsProperties arent Release v1.3.1 escription Check that one can upsert a batch of non-existing entities xpected behaviour Test events Dire when { SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entityOperations/upsert?options=\${update_option}' method set to 'POST' Request batch upsert operation over entity from filename	
Batch Upsert Entities with Response Body set to an array of created entities ids and Query Entities with Updated Entities set to '\$ {entities_to_be_upserted}' valid entities } P Permutation id TP/NGSI-LD/CI/Prov/BE/004_01_03 ame 004_01_03 EntityWithRelationshipsProperties arent Release v1.3.1 escription Check that one can upsert a batch of non-existing entities xpected behaviour Test events Dire when { SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entityOperations/upsert?options=\${update_option}' method set to 'POST'	
Batch Upsert Entities with Response Body set to an array of created entities ids—and Query Entities with Updated Entities set to '\$ {entities_to_be_upserted}' valid entities } P Permutation id TP/NGSI-LD/CI/Prov/BE/004_01_03 ame 004_01_03 EntityWithRelationshipsProperties arent Release v1.3.1 escription Check that one can upsert a batch of non-existing entities xpected behaviour Test events Dire when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entityOperations/upsert?options=\${update_option}' method set to 'POST' Request batch upsert operation over entity from filename '@{entities_to_be_upserted}' with update_option set to 'replace' and Content-	
Batch Upsert Entities with Response Body set to an array of created entities ids and Query Entities with Updated Entities set to '\$ {entities_to_be_upserted}' valid entities } P Permutation id TP/NGSI-LD/CI/Prov/BE/004_01_03 ame 004_01_03 EntityWithRelationshipsProperties arent Release v1.3.1 escription Check that one can upsert a batch of non-existing entities xpected behaviour Test events Dire when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entityOperations/upsert?options=\${update_option}' method set to 'POST' Request batch upsert operation over entity from filename '@{entities_to_be_upserted}' with update_option set to 'replace' and	
Batch Upsert Entities with Response Body set to an array of created entities ids and Query Entities with Updated Entities set to '\$ {entities_to_be_upserted}' valid entities} P Permutation id TP/NGSI-LD/CI/Prov/BE/004_01_03 ame 004_01_03 EntityWithRelationshipsProperties arent Release v1.3.1 escription Check that one can upsert a batch of non-existing entities xpected behaviour Test events Dire when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entityOperations/upsert?options=\${update_option}' method set to 'POST' Request batch upsert operation over entity from filename '@ {entities_to_be_upserted}' with update_option set to 'replace' and Content- Type set to 'application/ld+json' }	
Batch Upsert Entities with Response Body set to an array of created entities ids and Query Entities with Updated Entities set to '\$ {entities_to_be_upserted}' valid entities } P Permutation id TP/NGSI-LD/CI/Prov/BE/004_01_03 ame 004_01_03 EntityWithRelationshipsProperties arent Release v1.3.1 escription Check that one can upsert a batch of non-existing entities xpected behaviour Test events Dire when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entityOperations/upsert?options=\${update_option}' method set to 'POST' Request batch upsert operation over entity from filename '@{entities_to_be_upserted}' with update_option set to 'replace' and Content-Type set to 'application/ld+json' }	- Client
Batch Upsert Entities with Response Body set to an array of created entities ids and Query Entities with Updated Entities set to '\$ {entities_to_be_upserted}' valid entities } P Permutation id TP/NGSI-LD/CI/Prov/BE/004_01_03 ame 004_01_03 EntityWithRelationshipsProperties arent Release v1.3.1 escription Check that one can upsert a batch of non-existing entities xpected behaviour Test events Dire when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entityOperations/upsert?options=\${update_option}' method set to 'POST' Request batch upsert operation over entity from filename '@{entities_to_be_upserted}' with update_option set to 'replace' and Content- Type set to 'application/ld+json' } then { SUT ->	- Client
Batch Upsert Entities with Response Body set to an array of created entities ids and Query Entities with Updated Entities set to '\$ {entities_to_be_upserted}' valid entities } P Permutation id TP/NGSI-LD/CI/Prov/BE/004_01_03 ame 004_01_03 EntityWithRelationshipsProperties arent Release v1.3.1 escription Check that one can upsert a batch of non-existing entities xpected behaviour Test events Dire when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entityOperations/upsert?options=\${update_option}' method set to 'POST' Request batch upsert operation over entity from filename '@ {entities_to_be_upserted}' with update_option set to 'replace' and Content- Type set to 'application/Id+json' } then { the SUT sends a valid Response for the operations:	- Client
Batch Upsert Entities with Response Body set to an array of created entities ids and Query Entities with Updated Entities set to '\$ {entities_to_be_upserted}' valid entities } P Permutation id TP/NGSI-LD/CI/Prov/BE/004_01_03 ame 004_01_03 EntityWithRelationshipsProperties arent Release v1.3.1 escription Check that one can upsert a batch of non-existing entities xpected behaviour Test events Dire when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entityOperations/upsert?options=\${update_option}' method set to 'POST' Request batch upsert operation over entity from filename '@{entities_to_be_upserted}' with update_option set to 'replace' and Content- Type set to 'application/ld+json' } then { the SUT sends a valid Response for the operations: Batch Upsert Entities with Response Status Code set to	- Client
Batch Upsert Entities with Response Body set to an array of created entities ids and Query Entities with Updated Entities set to '\$ {entities_to_be_upserted}' valid entities } } P Permutation id TP/NGSI-LD/CI/Prov/BE/004_01_03 ame 004_01_03 EntityWithRelationshipsProperties arent Release v1.3.1 escription Check that one can upsert a batch of non-existing entities xpected behaviour Test events Dire when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entityOperations/upsert?options=\${update_option}' method set to 'POST' Request batch upsert operation over entity from filename '@{entities_to_be_upserted}' with update_option set to 'replace' and Content- Type set to 'application/ld+json' } then { the SUT sends a valid Response for the operations: Batch Upsert Entities with Response Status Code set to 201 and	- Client
Batch Upsert Entities with Response Body set to an array of created entities ids and Query Entities with Updated Entities set to '\$ {entities_to_be_upserted}' valid entities } P Permutation id TP/NGSI-LD/CI/Prov/BE/004_01_03 ame 004_01_03 EntityWithRelationshipsProperties arent Release v1.3.1 escription Check that one can upsert a batch of non-existing entities xpected behaviour Test events Dire when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entityOperations/upsert?options=\${update_option}' method set to 'POST' Request batch upsert operation over entity from filename '@{entities_to_be_upserted}' with update_option set to 'replace' and Content- Type set to 'application/ld+json' } then { the SUT sends a valid Response for the operations: Batch Upsert Entities with Response Status Code set to 201 and Batch Upsert Entities with Response Body set to an array of created	- Client
Batch Upsert Entities with Response Body set to an array of created entities ids and Query Entities with Updated Entities set to '\$ {entities_to_be_upserted}' valid entities } P Permutation id TP/NGSI-LD/CI/Prov/BE/004_01_03 ame 004_01_03 EntityWithRelationshipsProperties arent Release v1.3.1 escription Check that one can upsert a batch of non-existing entities xpected behaviour Test events Dire when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entityOperations/upsert?options=\${update_option}' method set to 'POST' Request batch upsert operation over entity from filename '@{entities_to_be_upserted}' with update_option set to 'replace' and Content- Type set to 'application/ld+json' } then { the SUT sends a valid Response for the operations: Batch Upsert Entities with Response Body set to an array of created entities ids and	- Client

4.4.3.2.4 UpdateBatchOfEntities

TP Id	TP/NGSI-LD/CI/Prov/BE/005_04
Test objective	Check that one cannot update a batch of entities with an invalid request
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.9
Config Id	CF_01

PICS Selection	PICS_BatEntUpd	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/BE/005_04_01	
Name	005_04_01 InvalidJson	
Parent Release	v1.3.1	
Description	Check that one cannot update a batch of entities with an invalid request	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entityOperations/update' method set to 'POST' Batch Entity Delete Request with operation set to 'update', Content- Type set to 'application/ Id+json', and body set to 'batch/invalid-json- sample.jsonId }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Batch Request Entities From File with Response Status Code set to 400 and Batch Request Entities From File with Response Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/ InvalidRequest' and Batch Request Entities From File with Response body containing 'title' element }	SUT -> Client
TP Permutation id	TP/NGSI-LD/CI/Prov/BE/005_04_02	-1
Name	005_04_02 InvalidJsonLd	
Parent Release	v1.3.1	
Description	Check that one cannot update a batch of entities with an invalid request	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entityOperations/update' method set to 'POST' Batch Entity Delete Request with operation set to 'update', Content- Type set to 'application/ Id+json', and body set to 'batch/invalid-json- Id-sample.jsonId }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Batch Request Entities From File with Response Status Code set to 400 and Batch Request Entities From File with Response Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/ BadRequestData' and Batch Request Entities From File with Response body containing 'title' element	SUT -> Client

TP Id	TP/NGSI-LD/CI/Prov/BE/005_02
Test objective	Check that one can update a batch of entities with noOverwrite option
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.9
Config Id	CF_01
PICS Selection	PICS_BatEntUpd
Initial conditions	with { the SUT being in the "initial state" and and containing a list of entities }
	Permutations
TP Permutation id	TP/NGSI-LD/CI/Prov/BE/005_02_01
Name	005_02_01 EntityWithExistingAttributes
Parent Release	v1.3.1

Description	Check that one can update a batch of entities with noOverwrite option	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing:	SUT <- Client
	URL set to '/ ngsi-ld/v1/entityOperations/update?\${params_as_string}'	
	method set to 'POST'	
	Request batch update operation over entity from filename	
	'@{entities_to_be_updated}' with overwrite_option set to '\$ {EMPTY}' and Content-Type set to 'application/	
	Id+json'	
	}	
	then { the SUT sends a valid Response for the operations:	SUT -> Client
	Batch Update Entities with Response Status Code set to	
	204 and Query Entities with Updated Entities set to '\$	
	{updated_entities}' valid entities }	
TP Permutation id	TP/NGSI-LD/CI/Prov/BE/005_02_02	
Name	005_02_02 EntityWithNewAttribute	
Parent Release Description	v1.3.1 Check that one can update a batch of entities with noOverwrite option	
Expected behaviour	Test events	Direction
•	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ ngsi-ld/v1/entityOperations/update?\${params_as_string}'	
	method set to 'POST'	
	Request batch update operation over entity from filename	
	'@{entities_to_be_updated}' with overwrite_option set to '\$ {EMPTY}' and Content-Type set to 'application/	
	Id+json'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations: Batch Update Entities with Response Status Code set to	
	204 and	
	Query Entities with Updated Entities set to '\$	
	{updated_entities}' valid entities	
TP Permutation id	TP/NGSI-LD/CI/Prov/BE/005_02_03	
Name	005_02_03 EntityWithNewAndExistingAttribute	
Parent Release	v1.3.1	
Description	Check that one can update a batch of entities with noOverwrite option	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/	SUT <- Client
	ngsi-ld/v1/entityOperations/update?\${params_as_string}' method set to 'POST'	
	Request batch update operation over entity from filename	
	'@{entities_to_be_updated}' with overwrite_option set to '\$	
	{EMPTY}' and Content-Type set to 'application/	
	ld+json' }	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Batch Update Entities with Response Status Code set to 204 and	
	Query Entities with Updated Entities set to '\$	
	{updated_entities}' valid entities	
]	

TP ld	TP/NGSI-LD/CI/Prov/BE/005_03	
Test objective	Check that one can update a batch of entities where some will succeed and others will fail	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.9	
Config Id	CF_01	
PICS Selection	PICS_BatEntUpd	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/BE/005_03_01	
Name	005_03_01 Update a batch of non-existing and existing entities	
Parent Release	v1.3.1	
Description	Check that one can update a batch of non-existing and existing entities	
Expected behaviour	Test events	Direction
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entityOperations/update?\${params_as_string}' method set to 'POST' Request batch update operation over entity from filename '@{entities_to_be_updated}' with overwrite_option set to '\$ {EMPTY}' and Content-Type set to 'application/ Id+json' }	
	then { the SUT sends a valid Response for the operations: Batch Update Entities with Response Status Code set to 207 and Batch Update Entities with Response body containing batch operation result set to '\$ {expected_batch_operation_result}' and Query Entities with Updated Entities set to '\$ {updated_entities}' valid entities }	SUT -> Client

TP Id	TP/NGSI-LD/CI/Prov/BE/005_01	
Test objective	Check that one can update a batch of entities	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.9	
Config Id	CF_01	
PICS Selection	PICS_BatEntUpd	
Initial conditions	with { the SUT being in the "initial state" and and containing a list of entities }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/BE/005_01_01	
Name	005_01_01 EntityWithSimpleProperties	
Parent Release	v1.3.1	
Description	Check that one can update a batch of entities	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entityOperations/update?\${params_as_string}' method set to 'POST' Request batch update operation over entity from filename '@{entities_to_be_updated}' with overwrite_option set to '\$ {EMPTY}' and Content-Type set to 'application/ Id+json' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Batch Update Entities with Response Status Code set to 204 and Query Entities with Updated Entities set to '\$ {updated_entities}' valid entities }	SUT -> Client
TP Permutation id	TP/NGSI-LD/CI/Prov/BE/005_01_02	
Name	005_01_02 EntityWithSimpleRelationships	
Parent Release	v1.3.1	
Description	Check that one can update a batch of entities	

Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entityOperations/update?\${params_as_string}'	
	method set to 'POST'	
	Request batch update operation over entity from filename	
	'@{entities_to_be_updated}' with overwrite_option set to '\$ {EMPTY}' and Content-Type set to 'application/	
	10+15-011 1	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	OO1 > OHOIR
	Batch Update Entities with Response Status Code set to	
	204 and	
	Query Entities with Updated Entities set to '\$	
	{updated_entities}' valid entities	
)	
TP Permutation id	TP/NGSI-LD/CI/Prov/BE/005_01_03	
Name	005_01_03 EntityWithRelationshipsProperties	
Parent Release	v1.3.1	
Description	Check that one can update a batch of entities	1 5
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ ngsi-Id/v1/entityOperations/update?\${params_as_string}'	
	method set to 'POST'	
	Request batch update operation over entity from filename	
	'@{entities_to_be_updated}' with overwrite_option set to '\$	
	{EMPTY}' and Content-Type set to 'application/	
	lid+json'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Batch Update Entities with Response Status Code set to	
	204 and	
	Query Entities with Updated Entities set to '\$	
	{updated_entities}' valid entities	
	<u>}</u>	

4.4.3.3 EntityAttributes

4.4.3.3.1 DeleteEntityAttribute

TP Id	TP/NGSI-LD/CI/Prov/EA/013_02	
Test objective	Check that one cannot delete an attribute from an entity with invalid/missing id	S
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.5	
Config Id	CF_01	
PICS Selection	PICS_AttDel	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing an initial Entity	y \${entity} with
	an id set to \${entityId} }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/EA/013_02_01	
Name	013_02_01 Delete an attribute if the Entity Id is not present	
Parent Release	v1.3.1	
Description	Check that one cannot delete an attribute from an entity with invalid/missing id	S
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}?\${params_as_string}'	
	method set to 'DELETE'	
	Delete Entity Attributes: and	
	Query Parameter: entityId set to '\$	

•		
	{EMPTY}' and	
	Query Parameter: attributeld set to 'speed' and	
	Query Parameter: datasetId set to '\$	
	{EMPTY}' and	
	Query Parameter: deleteAll set to 'false'	
	}	
	then {	SUT -> Client
		301 -> Client
	the SUT sends a valid Response for the operation:	
	Delete Entity Attributes with Response Status Code set to	
	400	
TP Permutation id	TP/NGSI-LD/CI/Prov/EA/013_02_02	
Name	013_02_02 Delete an attribute if the Entity Id is not a valid URI	
Parent Release	v1.3.1	
Description	Check that one cannot delete an attribute from an entity with invalid/missin	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/{entityld}/attrs/{attributeId}?\${params_as_string}'	
	method set to 'DELETE'	
	Delete Entity Attributes: and	
	Query Parameter: entityld set to 'thisIsAnInvalidURI' and	
	Query Parameter: attributeld set to 'speed' and	
	Query Parameter: datasetId set to '\$	
	{EMPTY}' and	
	Query Parameter: deleteAll set to 'false'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operation:	00.70
	Delete Entity Attributes with Response Status Code set to	
	400	
	1	
TP Permutation id	TP/NGSI-LD/CI/Prov/EA/013_02_03	
Name	013_02_03 Delete an attribute if the Attribute Name is not present	
Parent Release	v1.3.1	
Description	Check that one cannot delete an attribute from an entity with invalid/missin	a ido
Expected behaviour	Test events	Direction
Expected benaviour		
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}?\${params_as_string}'	
	method set to 'DELETE'	
	Delete Entity Attributes: and	
	Query Parameter: entityId set to '\$	
	{valid_entity_id}' and	
	Query Parameter: attributeld set to '\$	
	{EMPTY}' and	
	Query Parameter: datasetId set to '\$	
	{EMPTY}' and	
	Query Parameter: deleteAll set to 'false'	
	Query Parameter, delete Ali Set to Taise	
	} th. res. f	OUT OF
	then {	SUT -> Client
	the SUT sends a valid Response for the operation:	
	Delete Entity Attributes with Response Status Code set to	
	405	
	}	

TP Id	TP/NGSI-LD/CI/Prov/EA/013_04	
Test objective	Check that one can delete a scope from an entity	
Reference	ETSI GS CIM 009 V1.5.1 [1], clauses 4.18 and 5.6.5	
Config Id	CF_01	
PICS Selection	PICS_SCOPES and PICS_AttDel	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing an initial Entity \${entity} with	
	an id set to \${entityId} }	
Permutations		

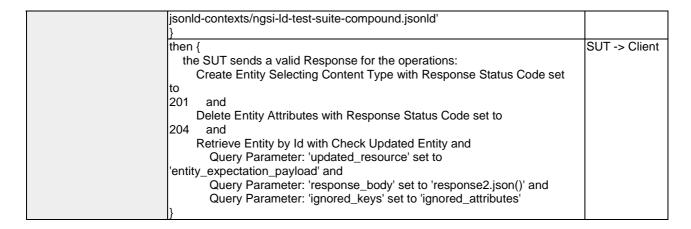
TP Permutation id	TP/NGSI-LD/CI/Prov/EA/013_04	
Name	013_04 Delete scope from an entity	
Parent Release	v1.5.1	
Description	Check that one can delete a scope from an entity	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/' method set to 'GET' Request Retrieve Entity by Id and Query Parameter: id set to '\${entity_id}' and Query Parameter: accept set to 'application/json' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Delete Entity Attributes with Response Status Code set to 204 and Delete Entity Attributes with Check Updated Entity and Query Parameter: 'ignored_keys' set to '\$ {None}' }	SUT -> Client

TP Id	TP/NGSI-LD/CI/Prov/EA/013_03	
Test objective	Check that one cannot delete an attribute from an entity with invalid/missing id	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.5	3
Config Id	CF_01	
PICS Selection		
	PICS_AttDel	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing an initial Entit	y \${entity} with
	an id set to \${entityId} } Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/EA/013_03_01	
Name	013_03_01 Delete an attribute when the Entity Id is not known to the system	
Parent Release	v1.3.1	
Description	Check that one cannot delete an attribute from an entity with invalid/missing id	S
Expected behaviour	Test events	Direction
•	when {	SUT <- Client
	the SUT receives a Request from the client containing:	OOT CONCIN
	URL set to '/	
	ngsi-ld/v1/entities/{entityld}/attrs/{attributeld}?\${params_as_string}'	
	method set to 'DELETE'	
	Delete Entity Attributes: and	
	Query Parameter: entityld set to '\$	
	{not_found_entity_id}' and	
	Query Parameter: attributeld set to 'speed' and	
	Query Parameter, detecated act to 'f	
	Query Parameter: datasetId set to '\$	
	{datasetId}' and	
	Query Parameter: deleteAll set to 'false'	
	then {	SUT -> Client
		SUT -> CITETIL
	the SUT sends a valid Response for the operation:	
	Delete Entity Attributes with Response Status Code set to 404	
	1	
TP Permutation id	TP/NGSI-LD/CI/Prov/EA/013_03_02	
Name	013_03_02 Delete an attribute when the Entity does not contain the target attr	ibute id
Parent Release	V1.3.1	ibate ia
Description	Check that one cannot delete an attribute from an entity with invalid/missing id	9
Expected behaviour	Test events	Direction
Expected Bellavious	when {	SUT <- Client
	the SUT receives a Request from the client containing:	OOT V OILCHE
	URL set to '/	
	Ingsi-Id/v1/entities/{entityId}/attrs/{attributeId}?\${params_as_string}'	
	method set to 'DELETE'	
	Delete Entity Attributes: and	
	Query Parameter: entityld set to '\$	
	{valid_entity_id}' and	

	Query Parameter: attributeld set to 'notFound' and	
	Query Parameter: datasetId set to '\$	
	{datasetId}' and	
	Query Parameter: deleteAll set to 'false'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operation:	OUT > OHORK
	Delete Entity Attributes with Response Status Code set to	
	404	
	\ \	
TP Permutation id	TP/NGSI-LD/CI/Prov/EA/013_03_03	
Name	013_03_03 Delete an attribute when the Entity does not contain the target attr	bute with
	same datasetId	
Parent Release	v1.3.1	
Description	Check that one cannot delete an attribute from an entity with invalid/missing id	S
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}?\${params_as_string}'	
	method set to 'DELETE'	
	Delete Entity Attributes: and	
	Query Parameter: entityId set to '\$	
	{valid_entity_id}' and	
	Query Parameter: attributeld set to 'speed' and	
	Query Parameter: datasetId set to '\$	
	{datasetId}' and	
	Query Parameter: deleteAll set to 'false'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operation:	
	Delete Entity Attributes with Response Status Code set to	
	404	

TP ld	TP/NGSI-LD/CI/Prov/EA/013_01	
Test objective	Check that one can delete an attribute from an entity	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.5	
Config Id	CF_01	
PICS Selection	PICS_AttDel	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/EA/013_01_01	
Name	013_01_01 Delete an attribute with the default instance	
Parent Release	v1.3.1	
Description	Check that one can delete an attribute from an entity	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/{entityld}/attrs/{attributeld}?\${params_as_string}'	
	method set to 'DELETE'	
	Delete Entity Attributes: and	
	Query Parameter: entityld set to '\$	
	{entity_id}' and	
	Query Parameter: attributeld set to 'speed' and	
	Query Parameter: datasetId set to '\$	
	{datasetId}' and Query Parameter: deleteAll set to '\$	
	{deleteAll}' and	
	Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	isonId-contexts/ngsi-Id-test-suite-compound.jsonId	
	301 u-001 ext3/11g3 -1u-test-3ulte-001 p0u u.j301 u }	
	I)	1

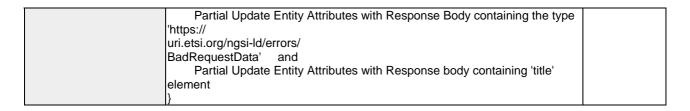
	T	Tarana
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Create Entity Selecting Content Type with Response Status Code set	
	to	
	201 and	
	Delete Entity Attributes with Response Status Code set to	
	204 and	
	Retrieve Entity by Id with Check Updated Entity and	
	Query Parameter: 'updated_resource' set to	
	'entity_expectation_payload' and	
	Query Parameter: 'response_body' set to 'response2.json()' and	
	Query Parameter: 'ignored_keys' set to 'ignored_attributes'	
]}	
TP Permutation id	TP/NGSI-LD/CI/Prov/EA/013_01_02	•
Name	013_01_02 Delete an attribute with the datasetId	
Parent Release	v1.3.1	
Description	Check that one can delete an attribute from an entity	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/{entityld}/attrs/{attributeId}?\${params_as_string}'	
	method set to 'DELETE'	
	Delete Entity Attributes: and	
	Query Parameter: entityId set to '\$	
	{entity_id}' and	
	Query Parameter: attributeld set to 'speed' and	
	Query Parameter: datasetId set to '\$	
	{datasetId}' and	
	Query Parameter: deleteAll set to '\$	
	{deleteAll}' and	
	Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	jsonld-contexts/ngsi-ld-test-suite-compound.jsonld'	
]	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	00.70
	Create Entity Selecting Content Type with Response Status Code set	
	to	
	201 and	
	Delete Entity Attributes with Response Status Code set to	
	204 and	
	Retrieve Entity by Id with Check Updated Entity and	
	Query Parameter: 'updated_resource' set to	
	'entity_expectation_payload' and	
	Query Parameter: 'response_body' set to 'response2.json()' and	
	Query Parameter: 'ignored_keys' set to 'ignored_attributes'	
	}	
TP Permutation id	TP/NGSI-LD/CI/Prov/EA/013_01_03	
Name	013_01_03 Delete all target attribute instances	
Parent Release	v1.3.1	
Description	Check that one can delete an attribute from an entity	
Expected behaviour	Test events	Direction
Expected beliavious		
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}?\${params_as_string}'	
	method set to 'DELETE'	
	Delete Entity Attributes: and	
	Query Parameter: entityld set to '\$	
	{entity_id}' and	
	Query Parameter: attributeld set to 'speed' and	
	Query Parameter: datasetId set to '\$	
	{datasetId}' and	
	Query Parameter: deleteAll set to '\$	
	{deleteAll}' and	
	Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	prorige.etai.org/1ep/cim/ngarid-teat-auite/-/raw/develop/feaouffea/	ĺ



4.4.3.3.2 PartialAttributeUpdate

	TP/NGSI-LD/CI/Prov/EA/012_02	
Test objective	Check that one cannot perform a partial update on an entity attribute with inval	lid/missing ids
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.4	_
Config Id	CF_01	
PICS Selection	PICS_ParAttUpd	
	with { the SUT being in the "initial state" and the SUT containing an initial Entity	v \${entitv} with
	an id set to \${entityId} }	, . (),
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/EA/012_02_01	
	012_02_01 Make a partial attribute update if the Entity Id is not present	
Parent Release	v1.3.1	
	Check that one cannot perform a partial update on an entity attribute with inval	lid/missing ids
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Request Partial Update Entity Attributes and Query Parameter: entityId set to '\$ {EMPTY}' and Query Parameter: AttributeId set to 'speed' and Query Parameter: fragment_filename set to 'vehicle- speed-equal-datasetid-fragment.jsonId' and Query Parameter: content_type set to 'application/ Id+json' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Partial Update Entity Attributes with Response Status Code set to 400 and Partial Update Entity Attributes with Response Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/ BadRequestData' and Partial Update Entity Attributes with Response body containing 'title' element }	SUT -> Client
	TP/NGSI-LD/CI/Prov/EA/012_02_02	
	012_02_02 Make a partial attribute update if the Entity Id is not a valid URI	
Parent Release	v1.3.1	
	Check that one cannot perform a partial update on an entity attribute with inval	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH'	SUT <- Client

	Request Partial Update Entity Attributes and	
	Query Parameter: entityId set to 'thisisaninvaliduri' and	
	Query Parameter: Attributeld set to 'speed' and	
	Query Parameter: fragment_filename set to 'vehicle-	
	speed-equal-datasetid-fragment.jsonld' and	
	Query Parameter: content_type set to 'application/	
	ld+json'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Partial Update Entity Attributes with Response Status Code set to	
	400 and	
	Partial Update Entity Attributes with Response Body containing the type	
	https://	
	uri.etsi.org/ngsi-ld/errors/	
	BadRequestData' and	
	Partial Update Entity Attributes with Response body containing 'title'	
	element	
	}	
TP Permutation id	TP/NGSI-LD/CI/Prov/EA/012_02_03	I.
Name	012_02_03 Make a partial attribute update if the Attribute type does not match	
Parent Release	V1.3.1	
		::-!/:- : : : : : : : : : : : : : : : : :
Description	Check that one cannot perform a partial update on an entity attribute with inva	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/{entityld}/attrs/{attributeld}'	
	method set to 'PATCH'	
	Request Partial Update Entity Attributes and	
	Query Parameter: entityId set to '\$	
	{valid_entity_id}' and	
	Query Parameter: Attributeld set to 'speed' and	
	Query Parameter: fragment_filename set to 'vehicle-	
	speed-equal-datasetid-different-type-fragment.jsonld' and	
	Query Parameter: content_type set to 'application/	
	ld+json'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Partial Update Entity Attributes with Response Status Code set to	
	400 and	
	Partial Update Entity Attributes with Response Body containing the type	
	https://	
	uri.etsi.org/ngsi-ld/errors/	
	BadRequestData' and	
	Partial Update Entity Attributes with Response body containing 'title'	
	element	
	}	
TP Permutation id	TP/NGSI-LD/CI/Prov/EA/012_02_04	•
Name	012_02_04 Make a partial attribute update if the entity fragment is empty	
Parent Release	v1.3.1	
4		الماسمام منح عاطح
Description	Check that one cannot perform a partial update on an entity attribute with inva	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}'	
	method set to 'PATCH'	
	Request Partial Update Entity Attributes and	
	Query Parameter: entityId set to '\${valid_entity_id}' and	
	Query Parameter: Attributeld set to 'speed' and	
	Query Parameter: fragment_filename set to 'empty-fragment.json' and	
	Query Parameter: content_type set to 'application/ld+json'	
	· · · · · · · · · · · · · · · · · · ·	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Partial Update Entity Attributes with Response Status Code set to	
	400 and	ļ



TP Id	TP/NGSI-LD/CI/Prov/EA/012_04	
Test objective	Check that one can perform a partial update on a LanguageProperty property	
Reference	ETSI GS CIM 009 V1.4.1 [1], clauses 4.5.18 and 5.6.4	
Config Id	CF_01	
PICS Selection		
Initial conditions	PICS_LANG and PICS_ParAttUpd	(† (= .= 1; 4, .) ; 4 =
initial conditions	with { the SUT being in the "initial state" and the SUT containing an initial Entit	y \${entity} with
	an id set to \${entityId} and an attribute with an id set to \${atrId} } Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/EA/012_04_01	
Name	012_04_01 Check that one can partially update a LanguageProperty property	
Parent Release	V1.4.1	
Description	Check that one can perform a partial update on a LanguageProperty property	
Expected behaviour	Test events	Direction
Exposica sonaviour	when {	SUT <- Client
	the SUT receives a Request from the client containing:	SOT <- Client
	URL set to '/	
	ngsi-ld/v1/entities/{entityld}/attrs/{attributeld}'	
	method set to 'PATCH'	
	Request Partial Update Entity Attributes and	
	Query Parameter: entityId set to '\$	
	{entity_id}' and	
	Query Parameter: Attributeld set to 'street' and	
	Query Parameter: fragment_filename set to 'building-	
	language-property-fragment.jsonId' and	
	Query Parameter: content_type set to 'application/	
	ld+json'	
	<u>}</u>	OLIT OF C
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Partial Update Entity Attributes with Response Status Code set to	
	204 and	
	Retrieve Entity by Id with Check Updated Entity and	
	Query Parameter: 'ignored_keys' set to '\$	
	[None]'	
	1}	

I== · ·		
TP ld	TP/NGSI-LD/CI/Prov/EA/012_03	
Test objective	Check that one cannot perform a partial update on an entity attribute if the ent	ity id or
	attribute is not known to the system	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.4	
Config Id	CF_01	
PICS Selection	PICS_ParAttUpd	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing an initial Entit	y \${entity} with
	an id set to \${entityId} }	
Permutations		
TP Permutation id	TP/NGSI-LD/CI/Prov/EA/012_03_01	
Name	012_03_01 Partial update when the Entity Id is not known to the system	
Parent Release	v1.3.1	
Description	Check that one cannot perform a partial update on an entity attribute if the entity id or	
	attribute is not known to the system	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/{entityld}/attrs/{attributeld}'	
	method set to 'PATCH'	
	Request Partial Update Entity Attributes and	

	Query Parameter: entityld set to '\$	
	{not_found_entity_id}' and	
	Query Parameter: Attributeld set to 'speed' and	
	Query Parameter: fragment_filename set to 'vehicle-	
	isparked-fragment.jsonId' and	
	Query Parameter: content_type set to 'application/	
	ld+json'	
	1	
	than	SUT -> Client
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Partial Update Entity Attributes with Response Status Code set to	
	404 and	
	Partial Update Entity Attributes with Response Body containing the type	
	'https://	
	uri.etsi.org/ngsi-ld/errors/	
	ResourceNotFound' and	
	Partial Update Entity Attributes with Response body containing 'title'	
	element	
)	
TP Permutation id	TP/NGSI-LD/CI/Prov/EA/012_03_02	<u> </u>
Name	012_03_02 Partial update when no default instance and no datasetId specified	1
		ı
Parent Release	v1.3.1	a de la la
Description	Check that one cannot perform a partial update on an entity attribute if the enti	ty id or
	attribute is not known to the system	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}'	
	method set to 'PATCH'	
	Request Partial Update Entity Attributes and	
	Query Parameter: entityld set to '\$	
	{valid_entity_id}' and	
	Query Parameter: Attributeld set to 'speed' and	
	Query Parameter: fragment_filename set to 'vehicle-	
	speed-no-datasetid-fragment.jsonld' and	
	Query Parameter: content_type set to 'application/	
	ld+json'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Partial Update Entity Attributes with Response Status Code set to	
	404 and	
	Partial Update Entity Attributes with Response Body containing the type	
	https://	
	uri.etsi.org/ngsi-ld/errors/	
	ResourceNotFound' and	
	Partial Update Entity Attributes with Response body containing 'title'	
	element	
	}	
TP Permutation id	TP/NGSI-LD/CI/Prov/EA/012_03_03	
Name	012_03_03 Partial update when no instance with the datasetId specified	
Parent Release	v1.3.1	
Description	Check that one cannot perform a partial update on an entity attribute if the enti	ity id or
•	attribute is not known to the system	,
Expected behaviour	Test events	Direction
Exposited Bollation	when {	SUT <- Client
	the SUT receives a Request from the client containing:	OUT V- CHELL
	URL set to '/	
	ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}'	
	method set to 'PATCH'	
	Request Partial Update Entity Attributes and	
	Query Parameter: entityId set to '\$	
	{valid_entity_id}' and	
	Query Parameter: Attributeld set to 'speed' and	
	Query Parameter: fragment_filename set to 'vehicle-	
	speed-unknown-datasetid-fragment.jsonId' and	
	Query Parameter: content_type set to 'application/	

		1
	ld+json'	
	then { the SUT sends a valid Response for the operations: Partial Update Entity Attributes with Response Status Code set to 404 and Partial Update Entity Attributes with Response Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/ ResourceNotFound' and	SUT -> Client
	Partial Update Entity Attributes with Response body containing 'title' element	
	}	
TP Permutation id	TP/NGSI-LD/CI/Prov/EA/012_03_04	
Name	012_03_04 Partial update when the Attribute Name does not exist in the entity	<u> </u>
Parent Release	v1.3.1	
Description	Check that one cannot perform a partial update on an entity attribute if the ent attribute is not known to the system	ity id or
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Request Partial Update Entity Attributes and Query Parameter: entityId set to '\$ {valid_entity_id}' and Query Parameter: AttributeId set to 'isParked2' and Query Parameter: fragment_filename set to 'vehicle- isparked-fragment.jsonId' and Query Parameter: content_type set to 'application/ Id+json' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Partial Update Entity Attributes with Response Status Code set to 404 and Partial Update Entity Attributes with Response Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/ ResourceNotFound' and Partial Update Entity Attributes with Response body containing 'title' element	SUT -> Client

TP ld	TD/NCSLLD/CI/Drov/EA/012 01	
	TP/NGSI-LD/CI/Prov/EA/012_01	
Test objective	Check that one can perform a partial update on an entity attribute	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.4	
Config Id	CF_01	
PICS Selection	PICS_ParAttUpd	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing an initial Entity	y \${entity} with
	an id set to \${entityId} and an attribute with an id set to \${atrId} }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/EA/012_01_01	
Name	012_01_01 Check that one can partially update an attribute	
Parent Release	v1.3.1	
Description	Check that one can perform a partial update on an entity attribute	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to //	
	ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}'	
	method set to 'PATCH'	
	Request Partial Update Entity Attributes and	
	Query Parameter: entityld set to '\$	
	{entity_id}' and	

	Query Parameter: AttributeId set to 'isParked' and	
	Query Parameter: fragment_filename set to 'vehicle-	
	isparked-fragment.jsonId' and	
	Query Parameter: content_type set to 'application/	
	ld+json'	
	<u>}</u>	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Partial Update Entity Attributes with Response Status Code set to	
	204 and	
	Retrieve Entity by Id with Check Updated Entity and	
	Query Parameter: 'updated_resource' set to	
	'entity_expectation_payload' and	
	Query Parameter: 'response_body' set to 'response1.json()' and	
	Query Parameter: 'ignored_keys' set to 'ignored_attributes'	
]	
TP Permutation id	TP/NGSI-LD/CI/Prov/EA/012_01_02	
Name	012_01_02 Check that one can partially update an attribute by specifying the	e datasetId
Parent Release	v1.3.1	
Description	Check that one can perform a partial update on an entity attribute	
Expected behaviour	Test events	Direction
-	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	The obtaining.	
	URL set to '/	
	URL set to '/ ngsi-ld/v1/entities/{entityld}/attrs/{attributeld}'	
	URL set to '/ ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH'	
	URL set to '/ ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Request Partial Update Entity Attributes and	
	URL set to '/ ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Request Partial Update Entity Attributes and	
	URL set to '/ ngsi-Id/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Request Partial Update Entity Attributes and Query Parameter: entityId set to '\$	
	URL set to '/ ngsi-ld/v1/entities/{entityld}/attrs/{attributeId}' method set to 'PATCH' Request Partial Update Entity Attributes and Query Parameter: entityld set to '\$ {entity_id}' and	
	URL set to '/ ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Request Partial Update Entity Attributes and Query Parameter: entityId set to '\$ {entity_id}' and Query Parameter: AttributeId set to 'speed' and	
	URL set to '/ ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Request Partial Update Entity Attributes and Query Parameter: entityId set to '\$ {entity_id}' and Query Parameter: AttributeId set to 'speed' and Query Parameter: fragment_filename set to 'vehicle-	
	URL set to '/ ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Request Partial Update Entity Attributes and Query Parameter: entityId set to '\$ {entity_id}' and Query Parameter: AttributeId set to 'speed' and Query Parameter: fragment_filename set to 'vehicle- speed-equal-datasetid-fragment.jsonId' and	
	URL set to '/ ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Request Partial Update Entity Attributes and Query Parameter: entityId set to '\$ {entity_id}' and Query Parameter: AttributeId set to 'speed' and Query Parameter: fragment_filename set to 'vehicle- speed-equal-datasetid-fragment.jsonId' and Query Parameter: content_type set to 'application/	
	URL set to '/ ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Request Partial Update Entity Attributes and Query Parameter: entityId set to '\$ {entity_id}' and Query Parameter: AttributeId set to 'speed' and Query Parameter: fragment_filename set to 'vehicle- speed-equal-datasetid-fragment.jsonId' and	
	URL set to '/ ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Request Partial Update Entity Attributes and Query Parameter: entityId set to '\$ {entity_id}' and Query Parameter: AttributeId set to 'speed' and Query Parameter: fragment_filename set to 'vehicle- speed-equal-datasetid-fragment.jsonId' and Query Parameter: content_type set to 'application/	
	URL set to '/ ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Request Partial Update Entity Attributes and Query Parameter: entityId set to '\$ {entity_id}' and Query Parameter: AttributeId set to 'speed' and Query Parameter: fragment_filename set to 'vehicle- speed-equal-datasetid-fragment.jsonId' and Query Parameter: content_type set to 'application/	SUT -> Client
	URL set to '/ ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Request Partial Update Entity Attributes and Query Parameter: entityId set to '\$ {entity_id}' and Query Parameter: AttributeId set to 'speed' and Query Parameter: fragment_filename set to 'vehicle- speed-equal-datasetid-fragment.jsonId' and Query Parameter: content_type set to 'application/ Id+json' }	SUT -> Client
	URL set to '/ ngsi-Id/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Request Partial Update Entity Attributes and Query Parameter: entityId set to '\$ {entity_id}' and Query Parameter: AttributeId set to 'speed' and Query Parameter: fragment_filename set to 'vehicle- speed-equal-datasetid-fragment.jsonId' and Query Parameter: content_type set to 'application/ Id+json' } then { the SUT sends a valid Response for the operations:	SUT -> Client
	URL set to '/ ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Request Partial Update Entity Attributes and Query Parameter: entityId set to '\$ {entity_id}' and Query Parameter: AttributeId set to 'speed' and Query Parameter: fragment_filename set to 'vehicle- speed-equal-datasetid-fragment.jsonId' and Query Parameter: content_type set to 'application/ Id+json' } then { the SUT sends a valid Response for the operations: Partial Update Entity Attributes with Response Status Code set to	SUT -> Client
	URL set to '/ ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Request Partial Update Entity Attributes and Query Parameter: entityId set to '\$ {entity_id}' and Query Parameter: AttributeId set to 'speed' and Query Parameter: fragment_filename set to 'vehicle- speed-equal-datasetid-fragment.jsonId' and Query Parameter: content_type set to 'application/ Id+json' } then { the SUT sends a valid Response for the operations: Partial Update Entity Attributes with Response Status Code set to	SUT -> Client
	URL set to '/ ngsi-Id/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Request Partial Update Entity Attributes and Query Parameter: entityId set to '\$ {entity_id}' and Query Parameter: AttributeId set to 'speed' and Query Parameter: fragment_filename set to 'vehicle- speed-equal-datasetid-fragment.jsonId' and Query Parameter: content_type set to 'application/ Id+json' } then { the SUT sends a valid Response for the operations: Partial Update Entity Attributes with Response Status Code set to and Retrieve Entity by Id with Check Updated Entity and	SUT -> Client
	URL set to '/ ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Request Partial Update Entity Attributes and Query Parameter: entityId set to '\$ {entity_id}' and Query Parameter: AttributeId set to 'speed' and Query Parameter: fragment_filename set to 'vehicle- speed-equal-datasetid-fragment.jsonId' and Query Parameter: content_type set to 'application/ Id+json' } then { the SUT sends a valid Response for the operations: Partial Update Entity Attributes with Response Status Code set to and Retrieve Entity by Id with Check Updated Entity and Query Parameter: 'updated_resource' set to	SUT -> Client
	URL set to '/ ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Request Partial Update Entity Attributes and Query Parameter: entityId set to '\$ {entity_id}' and Query Parameter: AttributeId set to 'speed' and Query Parameter: fragment_filename set to 'vehicle- speed-equal-datasetid-fragment.jsonId' and Query Parameter: content_type set to 'application/ Id+json' } then { the SUT sends a valid Response for the operations: Partial Update Entity Attributes with Response Status Code set to and Retrieve Entity by Id with Check Updated Entity and Query Parameter: 'updated_resource' set to 'entity_expectation_payload' and	SUT -> Client
	URL set to '/ ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Request Partial Update Entity Attributes and Query Parameter: entityId set to '\$ {entity_id}' and Query Parameter: AttributeId set to 'speed' and Query Parameter: fragment_filename set to 'vehicle- speed-equal-datasetid-fragment.jsonId' and Query Parameter: content_type set to 'application/ Id+json' } then { the SUT sends a valid Response for the operations: Partial Update Entity Attributes with Response Status Code set to and Retrieve Entity by Id with Check Updated Entity and Query Parameter: 'updated_resource' set to 'entity_expectation_payload' and Query Parameter: 'response_body' set to 'response1.json()' and	SUT -> Client
	URL set to '/ ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Request Partial Update Entity Attributes and Query Parameter: entityId set to '\$ {entity_id}' and Query Parameter: AttributeId set to 'speed' and Query Parameter: fragment_filename set to 'vehicle- speed-equal-datasetid-fragment.jsonId' and Query Parameter: content_type set to 'application/ Id+json' } then { the SUT sends a valid Response for the operations: Partial Update Entity Attributes with Response Status Code set to and Retrieve Entity by Id with Check Updated Entity and Query Parameter: 'updated_resource' set to 'entity_expectation_payload' and	SUT -> Client
	URL set to '/ ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Request Partial Update Entity Attributes and Query Parameter: entityId set to '\$ {entity_id}' and Query Parameter: AttributeId set to 'speed' and Query Parameter: fragment_filename set to 'vehicle- speed-equal-datasetid-fragment.jsonId' and Query Parameter: content_type set to 'application/ Id+json' } then { the SUT sends a valid Response for the operations: Partial Update Entity Attributes with Response Status Code set to and Retrieve Entity by Id with Check Updated Entity and Query Parameter: 'updated_resource' set to 'entity_expectation_payload' and Query Parameter: 'response_body' set to 'response1.json()' and	SUT -> Client

4.4.3.3.3 UpdateEntityAttributes

··		
TP ld	TP/NGSI-LD/CI/Prov/EA/011_04	
Test objective	Check that one cannot update entity attributes with invalid request body	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.2	
Config Id	CF_01	
PICS Selection	PICS_UpdAttEnt	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing an initial Entit	y \${entity} with
	an id set to \${entityId} }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/EA/011_04_01	
Name	011_04_01 Update entity attributes with invalid entity fragments	
Parent Release	v1.3.1	
Description	Check that one cannot update an attribute if the entity fragment is invalid	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}'	
	method set to 'PATCH'	
	Update Entity Attributes and	

Query Parameter: id set to '\${entity_id}' and Query Parameter: fragment_filename set to '\${fragment_filename}' and Query Parameter: content_type set to 'application/ld+json' }	
then { the SUT sends a valid Response for the operations: Update Entity Attributes with Response Status Code set to 400 and Update Entity Attributes with Response Body containing the type 'https:// uri.etsi.org/ngsi-Id/errors/ InvalidRequest' and Update Entity Attributes with Response body containing 'title' element }	SUT -> Client

TP ld	TP/NGSI-LD/CI/Prov/EA/011_02	
Test objective	Check that one cannot update entity attributes with invalid/missing id or invalid	request body
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.2	Toquoot body
Config Id	CF_01	
PICS Selection	PICS_UpdAttEnt	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing an initial Entit	v \${entity} with
	an id set to \${entityId} }	y φ(ortacy) with
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/EA/011_02_01	
Name	011_02_01 Update an attribute if the Entity Id is not present	
Parent Release	v1.3.1	
Description	Check that one cannot update entity attributes with invalid/missing id or invalid	request body
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Update Entity Attributes and Query Parameter: id set to '\$ {EMPTY}' and Query Parameter: fragment_filename set to 'vehicle- speed-two-datasetid-01-fragment.jsonId' and Query Parameter: content_type set to 'application/ Id+json' } then { the SUT sends a valid Response for the operations: Update Entity Attributes with Response Status Code set to 400 and Update Entity Attributes with Response Body containing the type 'https:// uri.etsi.org/ngsi-Id/errors/ BadRequestData' and Update Entity Attributes with Response body containing 'title' element	SUT <- Client SUT -> Client
	} TD (1001 D (01/D /54/044 00 00	
TP Permutation id	TP/NGSI-LD/CI/Prov/EA/011_02_02	
Name	011_02_02 Update an attribute if the Entity Id is not a valid URI	
Parent Release	v1.3.1 Check that one cannot update entity attributes with invalid/missing id or invalid	I request body
Description Expected behaviour	Test events	Direction
Expedied beliavious	when {	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Update Entity Attributes and Query Parameter: id set to 'thisisaninvaliduri' and Query Parameter: fragment_filename set to 'vehicle- speed-two-datasetid-01-fragment.jsonld' and Query Parameter: content_type set to 'application/	SOT A SHORT

ld+json'	
then {	SUT -> Client
the SUT sends a valid Response for the operations:	
Update Entity Attributes with Response Status Code set to	
400 and	
Update Entity Attributes with Response Body containing the type https://	
uri.etsi.org/ngsi-ld/errors/	
BadRequestData' and	
Update Entity Attributes with Response body containing 'title' element }	

Reference ETSI GS CIM 009 V1.5.1 [1], clauses 4.18 and 5.6.2	TP Id	TP/NGSI-LD/CI/Prov/EA/011_05	
Reference			
Config Id			
PICS Scopes and PICS UpdattEnt			
Initial conditions		_	
In id set to \${entity/d}} Permutations			v ¢(optity) with
Permutation id TP/NGSI-LD/CI/Prov/EA/011_05_01	ilitiai conditions		y φ{enitty} with
TP Permutation id			
Name	TP Permutation id		
Description			
Check that scope is replaced if entity already has a scope Test events Direction	Parent Release		
Test events Direction	Description		
when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/ method set to 'GET' Request Retrieve Entity by Id and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: context set to 'application/ ison' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite-/raw/develop/resources/ isonId-contexts/ngsi-Id-test-suite-compound.jsonId' } then { the SUT sends a valid Response for the operations: Update Entity Attributes with Response Status Code set to 204 and Update Entity Attributes with Check Updated Entity and Query Parameter: 'ignored_keys' set to '\$ {\notentione} TP Permutation id TP/NGSI-LD/CI/Prov/EA/011_05_02 Name			Direction
Name Description Check that scope is not added if entity does not already have a scope		the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/' method set to 'GET' Request Retrieve Entity by Id and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ json' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' } then { the SUT sends a valid Response for the operations: Update Entity Attributes with Response Status Code set to and Update Entity Attributes with Check Updated Entity and Query Parameter: 'ignored_keys' set to '\$	SUT -> Client
Name Description Check that scope is not added if entity does not already have a scope		}	
Parent Release v1.5.1 Description Check that scope is not added if entity does not already have a scope Expected behaviour Test events Direction when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/' method set to 'GET' Request Retrieve Entity by Id and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ json' and			
Check that scope is not added if entity does not already have a scope Test events Direction			
### Test events When { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/' method set to 'GET' Request Retrieve Entity by Id and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ json' and SUT <- Client Application SUT <- Client SUT			
when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/' method set to 'GET' Request Retrieve Entity by Id and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ json' and			Direction
the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/' method set to 'GET' Request Retrieve Entity by Id and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/ json' and	Expected beliaviour		
Query Parameter: context set to 'nttps://		the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/' method set to 'GET' Request Retrieve Entity by Id and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: accept set to 'application/	O I X- OHGIIL

jsonld-contexts/ngsi-ld-test-suite-compound.jsonld'	
<u>}</u>	
then {	SUT -> Client
the SUT sends a valid Response for the operations:	
Update Entity Attributes with Response Status Code set to	
207 and	
Update Entity Attributes with Check Updated Entity and	
Query Parameter: 'ignored_keys' set to '\$	
{None}'	
\}	

TP Id	TP/NGSI-LD/CI/Prov/EA/011_01	
Test objective	Check that one can update entity attributes	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.2	
Config Id	CF_01	
PICS Selection	PICS_UpdAttEnt	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing an initial Enti	ty \${entity} with
	an id set to \${entityId} }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/EA/011_01_01	
Name	011_01_01 Check that one can update existing attributes with no datasetId	
Parent Release	v1.3.1	
Description	Check that one can update entity attributes	•
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}'	
	method set to 'PATCH'	
	Update Entity Attributes and	
	Query Parameter: id set to '\$	
	{entity_id}' and	
	Query Parameter: fragment_filename set to 'vehicle-	
	speed-two-datasetid-01-fragment.jsonld' and	
	Query Parameter: content_type set to 'application/	
	ld+json'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Update Entity Attributes with Response Status Code set to	
	204 and	
	Update Entity Attributes with Check Response Body Content and	
	Query Parameter: expectation_filename set to	
	'expectation_resp_filename' and	
	Query Parameter: response_body set to 'response.json()' and	
	Query Parameter: additional_ignored_path set to '\$	
	{EMPTY}' and	
	Retrieve Entity by Id with Check Updated Entity and	
	Query Parameter: 'updated_resource' set to	
	'entity_expectation_payload' and	
	Query Parameter: 'response_body' set to 'response1.json()' and Query Parameter: 'ignored_keys' set to 'ignored_attributes'	
	Query Parameter. Ignored_keys set to ignored_attributes	
TP Permutation id	TP/NGSI-LD/CI/Prov/EA/011_01_02	1
Name	011_01_02 Check that one can update existing attributes with the datasetId	
Parent Release		
Description	v1.3.1 Check that one can update entity attributes	
Expected behaviour	Test events	Direction
Expected beliaviour		SUT <- Client
	when { the SUT receives a Request from the client containing:	SUT <- CHEIR
	URL set to '/	
	ngsi-ld/v1/entities/{entityld}/attrs/{attributeId}'	
	method set to 'PATCH'	
	Update Entity Attributes and	
	Query Parameter: id set to '\$	
	{entity_id}' and	

T		
	Query Parameter: fragment_filename set to 'vehicle-	
	speed-two-datasetid-02-fragment.jsonld' and	
	Query Parameter: content_type set to 'application/	
	ld+json'	
	About (SUT -> Client
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Update Entity Attributes with Response Status Code set to	
	204 and	
	Update Entity Attributes with Check Response Body Content and	
	Query Parameter: expectation_filename set to	
	'expectation_resp_filename' and	
	Query Parameter: response_body set to 'response.json()' and	
	Query Parameter: additional_ignored_path set to '\$	
	{EMPTY}' and	
	Retrieve Entity by Id with Check Updated Entity and	
	Query Parameter: 'updated_resource' set to	
	'entity_expectation_payload' and	
	Query Parameter: 'response_body' set to 'response1.json()' and	
	Query Parameter: 'ignored_keys' set to 'ignored_attributes'	
	}	
TP Permutation id	TP/NGSI-LD/CI/Prov/EA/011_01_03	
Name	011_01_03 Check that one can update only some attributes while unknown a	re ignored
Parent Release	v1.3.1	
Description	Check that one can update entity attributes	T
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	•	
	URL set to '/	
	ngsi-ld/v1/entities/{entityld}/attrs/{attributeld}'	
	ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH'	
	ngsi-ld/v1/entities/{entityld}/attrs/{attributeId}' method set to 'PATCH' Update Entity Attributes and	
	ngsi-ld/v1/entities/{entityld}/attrs/{attributeId}' method set to 'PATCH' Update Entity Attributes and Query Parameter: id set to '\$	
	ngsi-Id/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Update Entity Attributes and Query Parameter: id set to '\$ {entity_id}' and	
	ngsi-Id/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Update Entity Attributes and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: fragment_filename set to 'vehicle-	
	ngsi-Id/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Update Entity Attributes and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: fragment_filename set to 'vehicle-speed-two-datasetid-03-fragment.jsonId' and	
	ngsi-Id/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Update Entity Attributes and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: fragment_filename set to 'vehicle- speed-two-datasetid-03-fragment.jsonId' and Query Parameter: content_type set to 'application/	
	ngsi-Id/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Update Entity Attributes and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: fragment_filename set to 'vehicle-speed-two-datasetid-03-fragment.jsonId' and	
	ngsi-Id/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Update Entity Attributes and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: fragment_filename set to 'vehicle- speed-two-datasetid-03-fragment.jsonId' and Query Parameter: content_type set to 'application/	
	ngsi-Id/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Update Entity Attributes and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: fragment_filename set to 'vehicle-speed-two-datasetid-03-fragment.jsonId' and Query Parameter: content_type set to 'application/ Id+json' } then {	SUT -> Client
	ngsi-Id/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Update Entity Attributes and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: fragment_filename set to 'vehicle-speed-two-datasetid-03-fragment.jsonId' and Query Parameter: content_type set to 'application/ Id+json' } then { the SUT sends a valid Response for the operations:	SUT -> Client
	ngsi-Id/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Update Entity Attributes and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: fragment_filename set to 'vehicle-speed-two-datasetid-03-fragment.jsonId' and Query Parameter: content_type set to 'application/ Id+json' } then {	SUT -> Client
	ngsi-Id/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Update Entity Attributes and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: fragment_filename set to 'vehicle- speed-two-datasetid-03-fragment.jsonId' and Query Parameter: content_type set to 'application/ Id+json' } then { the SUT sends a valid Response for the operations: Update Entity Attributes with Response Status Code set to	SUT -> Client
	ngsi-Id/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Update Entity Attributes and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: fragment_filename set to 'vehicle- speed-two-datasetid-03-fragment.jsonId' and Query Parameter: content_type set to 'application/ Id+json' } then { the SUT sends a valid Response for the operations: Update Entity Attributes with Response Status Code set to and Update Entity Attributes with Check Response Body Content and	SUT -> Client
	ngsi-Id/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Update Entity Attributes and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: fragment_filename set to 'vehicle- speed-two-datasetid-03-fragment.jsonId' and Query Parameter: content_type set to 'application/ Id+json' } then { the SUT sends a valid Response for the operations: Update Entity Attributes with Response Status Code set to	SUT -> Client
	ngsi-Id/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Update Entity Attributes and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: fragment_filename set to 'vehicle- speed-two-datasetid-03-fragment.jsonId' and Query Parameter: content_type set to 'application/ Id+json' } then { the SUT sends a valid Response for the operations: Update Entity Attributes with Response Status Code set to 4 and Update Entity Attributes with Check Response Body Content and Query Parameter: expectation_filename set to 'expectation_resp_filename' and	SUT -> Client
	ngsi-Id/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Update Entity Attributes and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: fragment_filename set to 'vehicle- speed-two-datasetid-03-fragment.jsonId' and Query Parameter: content_type set to 'application/ Id+json' } then { the SUT sends a valid Response for the operations: Update Entity Attributes with Response Status Code set to 4 and Update Entity Attributes with Check Response Body Content and Query Parameter: expectation_filename set to 'expectation_resp_filename' and Query Parameter: response_body set to 'response.json()' and	SUT -> Client
	ngsi-Id/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Update Entity Attributes and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: fragment_filename set to 'vehicle- speed-two-datasetid-03-fragment.jsonId' and Query Parameter: content_type set to 'application/ Id+json' } then { the SUT sends a valid Response for the operations: Update Entity Attributes with Response Status Code set to 4 and Update Entity Attributes with Check Response Body Content and Query Parameter: expectation_filename set to 'expectation_resp_filename' and Query Parameter: response_body set to 'response.json()' and Query Parameter: additional_ignored_path set to '\$	SUT -> Client
	ngsi-Id/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Update Entity Attributes and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: fragment_filename set to 'vehicle- speed-two-datasetid-03-fragment.jsonId' and Query Parameter: content_type set to 'application/ Id+json' } then { the SUT sends a valid Response for the operations: Update Entity Attributes with Response Status Code set to 4 and Update Entity Attributes with Check Response Body Content and Query Parameter: expectation_filename set to 'expectation_resp_filename' and Query Parameter: response_body set to 'response.json()' and Query Parameter: additional_ignored_path set to '\$ {EMPTY}' and	SUT -> Client
	ngsi-Id/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Update Entity Attributes and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: fragment_filename set to 'vehicle- speed-two-datasetid-03-fragment.jsonId' and Query Parameter: content_type set to 'application/ Id+json' } then { the SUT sends a valid Response for the operations: Update Entity Attributes with Response Status Code set to 204 and Update Entity Attributes with Check Response Body Content and Query Parameter: expectation_filename set to 'expectation_resp_filename' and Query Parameter: response_body set to 'response.json()' and Query Parameter: additional_ignored_path set to '\$ {EMPTY}' and Retrieve Entity by Id with Check Updated Entity and	SUT -> Client
	ngsi-Id/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Update Entity Attributes and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: fragment_filename set to 'vehicle- speed-two-datasetid-03-fragment.jsonId' and Query Parameter: content_type set to 'application/ Id+json' } then { the SUT sends a valid Response for the operations: Update Entity Attributes with Response Status Code set to 4 and Update Entity Attributes with Check Response Body Content and Query Parameter: expectation_filename set to 'expectation_resp_filename' and Query Parameter: response_body set to 'response.json()' and Query Parameter: additional_ignored_path set to '\$ {EMPTY}' and	SUT -> Client
	ngsi-Id/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Update Entity Attributes and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: fragment_filename set to 'vehicle- speed-two-datasetid-03-fragment.jsonId' and Query Parameter: content_type set to 'application/ Id+json' } then { the SUT sends a valid Response for the operations: Update Entity Attributes with Response Status Code set to 204 and Update Entity Attributes with Check Response Body Content and Query Parameter: expectation_filename set to 'expectation_resp_filename' and Query Parameter: response_body set to 'response.json()' and Query Parameter: additional_ignored_path set to '\$ {EMPTY}' and Retrieve Entity by Id with Check Updated Entity and Query Parameter: 'updated_resource' set to 'entity_expectation_payload' and	SUT -> Client
	ngsi-Id/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Update Entity Attributes and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: fragment_filename set to 'vehicle- speed-two-datasetid-03-fragment.jsonId' and Query Parameter: content_type set to 'application/ Id+json' } then { the SUT sends a valid Response for the operations: Update Entity Attributes with Response Status Code set to 4 and Update Entity Attributes with Check Response Body Content and Query Parameter: expectation_filename set to 'expectation_resp_filename' and Query Parameter: response_body set to 'response.json()' and Query Parameter: additional_ignored_path set to '\$ {EMPTY}' and Retrieve Entity by Id with Check Updated Entity and Query Parameter: 'updated_resource' set to 'entity_expectation_payload' and Query Parameter: 'response_body' set to 'response1.json()' and	SUT -> Client
	ngsi-Id/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Update Entity Attributes and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: fragment_filename set to 'vehicle- speed-two-datasetid-03-fragment.jsonId' and Query Parameter: content_type set to 'application/ Id+json' } then { the SUT sends a valid Response for the operations: Update Entity Attributes with Response Status Code set to 204 and Update Entity Attributes with Check Response Body Content and Query Parameter: expectation_filename set to 'expectation_resp_filename' and Query Parameter: response_body set to 'response.json()' and Query Parameter: additional_ignored_path set to '\$ {EMPTY}' and Retrieve Entity by Id with Check Updated Entity and Query Parameter: 'updated_resource' set to 'entity_expectation_payload' and	SUT -> Client

TP ld	TP/NGSI-LD/CI/Prov/EA/011_03		
Test objective	Check that one cannot update entity attributes if the entity id or attributes are not known to		
	the system		
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.2		
Config Id	CF_01		
PICS Selection	PICS_UpdAttEnt		
Initial conditions	with { the SUT containing an initial state }		
	Permutations		
TP Permutation id	TP/NGSI-LD/CI/Prov/EA/011_03_01		
Name	011_03_01 Update entity attributes when the entity id is not known to the system		
Parent Release	v1.3.1		

Description	Check that one cannot update entity attributes if the entity id or attributes are the system	not known to
Expected behaviour	Test events	Direction
Expected benaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/{entityId}/attrs/{attributeId}' method set to 'PATCH' Update Entity Attributes and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: fragment_filename set to 'vehicle- speed-two-datasetid-01-fragment.jsonId' and Query Parameter: content_type set to 'application/ Id+json' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Update Entity Attributes with Response Status Code set to 404 and Update Entity Attributes with Response Body containing the type 'https:// uri.etsi.org/ngsi-Id/errors/ ResourceNotFound' and Update Entity Attributes with Response body containing 'title' element }	SUT -> Client

4.4.3.3.4 AppendEntityAttributes

TP ld	TP/NGSI-LD/CI/Prov/EA/010_04	
Test objective	Check that one can append entity attributes	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.3	
Config Id	CF 01	
PICS Selection	PICS_AppAttEnt	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing an initial Entit	v \${entity} with
	an id set to \${entityId}}	y φ(ortacy) with
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/EA/010_04_01	
Name	010_04_01 Append entity attributes and ignore existing multi-attribute instance	е
Parent Release	v1.3.1	
Description	Check that one can append entity attributes	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/{id}/attrs?options=\${options}'	
	method set to 'POST'	
	Append entity attributes with parameters: and	
	Query Parameter: id set to '\$	
	{entity_id}' and	
	Query Parameter: fragment_filename set to 'vehicle-	
	attribute-to-add-fragment.jsonld' and	
	Query Parameter: content_type set to 'application/	
	ld+json' and	
	Query Parameter: options set to 'noOverwrite'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Append Entity Attributes With Parameters with Response Status Code	
	set to	
	204 and	
	Retrieve Entity by Id with Check Updated Entity and	
	Query Parameter: 'updated_resource' set to	
	entity_expectation_payload and	
	Query Parameter: 'response_body' set to 'response1.json()' and	

	Query Parameter: 'ignored_keys' set to 'ignored_attributes'	
TD Dammartation id	TD/NOOLL D/O//Draw/EA/040, 04, 00	
TP Permutation id	TP/NGSI-LD/CI/Prov/EA/010_04_02	
Name	010_04_02 Append entity attributes with a new multi-attribute instance	
Parent Release	v1.3.1	
Description	Check that one can append entity attributes	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/	
	ngsi-ld/v1/entities/{id}/attrs?options=\${options}' method set to 'POST'	
	Append entity attributes with parameters: and	
	Query Parameter: id set to '\$	
	{entity_id}' and	
	Query Parameter: fragment_filename set to 'vehicle-	
	speed-different-datasetid-fragment.jsonId' and	
	Query Parameter: content_type set to 'application/	
	Id+ison' and	
	Query Parameter: options set to 'noOverwrite'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Append Entity Attributes With Parameters with Response Status Code	
	set to	
	204 and	
	Retrieve Entity by Id with Check Updated Entity and	
	Query Parameter: 'updated_resource' set to	
	'entity_expectation_payload' and	
	Query Parameter: 'response_body' set to 'response1.json()' and	
	Query Parameter: 'ignored_keys' set to 'ignored_attributes' }	

TP ld	TD/N/CCLL D/CL/Dray/FA/040, OC	
	TP/NGSI-LD/CI/Prov/EA/010_06	
Test objective	Check that one can append a LanguageProperty property to an entity	
Reference	ETSI GS CIM 009 V1.4.1 [1], clauses 4.5.18 and 5.6.3	
Config Id	CF_01	
PICS Selection	PICS_LANG and PICS_AppAttEnt	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing an initial En	tity \${entity} with
	an id set to \${entityId} }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/EA/010_06_01	
Name	010_06_01 Append a LanguageProperty property	
Parent Release	v1.4.1	
Description	Check that one can append a LanguageProperty property to an entity	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/{id}/attrs' method set to 'POST' Append Entity Attributes and Query Parameter: id set to '\$ {entity_id}' and Query Parameter: fragment_filename set to 'vehicle- new-language-property-fragment.jsonld' and Query Parameter: content_type set to 'application/ Id+json' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Append Entity Attributes with Response Status Code set to 204 and Retrieve Entity by Id with Check Updated Entity and Query Parameter: 'ignored_keys' set to '\$	SUT -> Client



TP Id	TP/NGSI-LD/CI/Prov/EA/010_02	
Test objective	Check that one cannot append entity attributes with invalid/missing id or inval	id request body
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.3	ia roquoot boay
Config Id	CF_01	
PICS Selection	PICS AppAttEnt	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing an initial Enti	tv \${entity} with
	an id set to \${entityId} }	ity φίσιπτης with
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/EA/010_02_01	
Name	010_02_01 Append entity attributes if the entity ld is not present	
Parent Release	v1.3.1	
Description	Check that one cannot append entity attributes with invalid/missing id or inval	id request hody
Expected behaviour	Test events	Direction
Expedica Bellaviou	when {	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entities/{id}/attrs' method set to 'POST' Append Entity Attributes and Query Parameter: id set to '\$ {EMPTY}' and Query Parameter: fragment_filename set to 'vehicle-attribute-to-add-fragment.jsonld' and Query Parameter: content_type set to 'application/ Id+json' } then { the SUT sends a valid Response for the operations: Append Entity Attributes with Response Status Code set to 400 and Append Entity Attributes with Response Body containing the type 'https:// uri.etsi.org/ngsi-Id/errors/ BadRequestData' and	SUT -> Client
TP Permutation id	Append Entity Attributes with Response body containing 'title' element } TP/NGSI-LD/CI/Prov/EA/010_02_02	
Name	010_02_02 Append entity attributes if the Entity Id is not a valid URI	
Parent Release	v1.3.1	
Description	Check that one cannot append entity attributes with invalid/missing id or inval	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entities/{id}/attrs' method set to 'POST' Append Entity Attributes and Query Parameter: id set to 'thisisaninvaliduri' and Query Parameter: fragment_filename set to 'vehicle- attribute-to-add-fragment.jsonld' and Query Parameter: content_type set to 'application/ Id+json' } then { the SUT sends a valid Response for the operations: Append Entity Attributes with Response Status Code set to 400 and Append Entity Attributes with Response Body containing the type 'https:// uri.etsi.org/ngsi-Id/errors/ BadRequestData' and Append Entity Attributes with Response body containing 'title' element }	SUT <- Client SUT -> Client

TP Id	TP/NGSI-LD/CI/Prov/EA/010_05	
Test objective	Check that one cannot append entity attributes with invalid/missing id or invalid	d request body
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.3	
Config Id	CF_01	
PICS Selection	PICS_AppAttEnt	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing an initial Entit	y \${entity} with
	an id set to \${entityId} }	
	Permutations Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/EA/010_05_01	
Name	010_05_01 Append entity attributes with invalid entity fragments	
Parent Release	v1.3.1	
Description	Check that one cannot append entity attributes with invalid entity fragments	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ngsi-ld/v1/entities/{id}/attrs' method set to 'POST' Append Entity Attributes and Query Parameter: id set to '\${entity_id}' and Query Parameter: fragment_filename set to 'invalid-fragment.jsonld' and Query Parameter: content_type set to 'application/ld+json' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Append Entity Attributes with Response Status Code set to 400 and Append Entity Attributes with Response Body containing the type 'https:// uri.etsi.org/ngsi-Id/errors/ InvalidRequest' and Append Entity Attributes with Response body containing 'title' element }	SUT -> Client

TP Id	TP/NGSI-LD/CI/Prov/EA/010_01	
Test objective	Check that one can append entity attributes	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.3	
Config Id	CF_01	
PICS Selection	PICS_AppAttEnt	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing an initial Enti	ty \${entity} with
	an id set to \${entityId} }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/EA/010_01_01	
Name	010_01_01 Append entity attributes	
Parent Release	v1.3.1	
Description	Check that one can append entity attributes	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/{id}/attrs'	
	method set to 'POST'	
	Append Entity Attributes and	
	Query Parameter: id set to '\$	
	{entity_id}' and	
	Query Parameter: fragment_filename set to 'vehicle-	
	new-attribute-fragment.jsonld' and	
	Query Parameter: content_type set to 'application/	
	ld+json'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Append Entity Attributes with Response Status Code set to	
	204 and	
	Retrieve Entity by Id with Check Updated Entity and	
	Query Parameter: 'updated_resource' set to	
	entity_expectation_payload and	

	Query Parameter: 'response_body' set to 'response1.json()' and	
	Query Parameter: 'ignored_keys' set to '\$	
	{None}'	
	}	
TP Permutation id	TP/NGSI-LD/CI/Prov/EA/010_01_02	
Name	010_01_02 Append entity attributes with different datasetid	
Parent Release	v1.3.1	
Description	Check that one can append entity attributes	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/{id}/attrs'	
	method set to 'POST'	
	Append Entity Attributes and	
	Query Parameter: id set to '\$	
	{entity_id}' and	
	Query Parameter: fragment_filename set to 'vehicle-	
	speed-different-datasetid-fragment.jsonld' and	
	Query Parameter: content_type set to 'application/	
	ld+json'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Append Entity Attributes with Response Status Code set to	
	204 and	
	Retrieve Entity by Id with Check Updated Entity and	
	Query Parameter: 'updated_resource' set to	
	entity_expectation_payload and	
	Query Parameter: 'response_body' set to 'response1.json()' and	
	Query Parameter: 'ignored_keys' set to '\$	
	{None}'	
]}	

TP Id	TP/NGSI-LD/CI/Prov/EA/010 03	
	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	no evetem
Test objective	Check that one cannot append entity attributes if the entity id is not known to the	ie system
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.3	
Config Id	CF_01	
PICS Selection	PICS_AppAttEnt	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/EA/010_03_01	
Name	010_03_01 Append entity attributes when the entity id is not known to the syst	em
Parent Release	v1.3.1	
Description	Check that one cannot append entity attributes if the entity id is not known to the	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/{id}/attrs'	
	method set to 'POST'	
	Append Entity Attributes and	
	Query Parameter: id set to '\$	
	{entity_id}' and	
	Query Parameter: fragment_filename set to 'vehicle-	
	attribute-to-add-fragment.jsonld' and	
	Query Parameter: content_type set to 'application/	
	ld+json'	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Append Entity Attributes with Response Status Code set to	
	404 and	
	Append Entity Attributes with Response Body containing the type	
	https://	
	uri.etsi.org/ngsi-ld/errors/	

ResourceNotFound' and	
Append Entity Attributes with Response body containing 'title' element	
}	

4.4.3.4 Entities

4.4.3.4.1 CreateEntity

TP Id	TP/NGSI-LD/CI/Prov/E/001_07	
Test objective	Check that the @context is obtained from the request payload body itself if the Content-	
-	Type header is "application/ld+json"	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 6.3.5	
Config Id	CF_01	
PICS Selection	PICS_HTTP_CONTEXT_RESOLV	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/E/001_07_01	
Name	001_07_01 Create one entity using a JSON-LD @context obtained from the re	equest
	payload with ld context	
Parent Release	v1.3.1	
Description	Check that the @context is obtained from the request payload body itself if the	Content-
	Type header is "application/ld+json" and retrieve with Id context	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/'	
	method set to 'POST'	
	Request Header['Content-	
	Type'] set to 'application/ld+json' and	
	payload defined in file: 'building-simple-	
	attributes-sample.jsonld'	
	}	OUT OF
	then {	SUT -> Client
	the SUT sends a valid Response for the operation:	
	Retrieve Entity by Id with Check Response Body containing an Attribute set to and	
	Query Parameter: expected_attribute_name set to 'almostFull' and	
	Query Parameter: response_body set to 'response1.json()'	
	Receive a diameter: response_body set to response rijson()	
TP Permutation id	TP/NGSI-LD/CI/Prov/E/001_07_02	
Name	001_07_02 Create one entity using a JSON-LD @context obtained from the re	equest
	payload without context	,44001
Parent Release	v1.3.1	
Description	Check that the @context is obtained from the request payload body itself if the	Content-
	Type header is "application/ld+json" and retrieve without context	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to /	
	ngsi-ld/v1/entities/'	
	method set to 'POST'	
	Request Header['Content-	
	Type'] set to 'application/ld+json' and	
	payload defined in file: 'building-simple-	
	attributes-sample.jsonld'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operation:	
	Retrieve Entity by Id with Check Response Body containing an Attribute	
	set to and	
	Query Parameter: expected_attribute_name set to 'https://	
	ngsi-ld-test-suite/context#almostFull' and	
	Query Parameter: response_body set to 'response1.json()'	
	}	

TP Id	TP/NGSI-LD/CI/Prov/E/001_05	
Test objective	Check that the default @context is used if the Content-Type header is "applicated by the content of the Content	ation/icon" and
rest objective	the Link header does not contain a JSON-LD @context	alion/json and
Deference		
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 6.3.5	
Config Id	CF_01	
PICS Selection	PICS_HTTP_CONTEXT_RESOLV	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/E/001_05_01	
Name	001_05_01 Create one entity using the default context with JSON content type without context	e and request
Parent Release	v1.3.1	
Description	Check that the default @context is used if the Content-Type header is "applicated and the Content-Type header and the Content-Type he	ation/json" and
•	the Link header does not contain a JSON-LD @context, requesting without co	ntext
Expected behaviour	Test events	Direction
•	when {	SUT <- Client
	the SUT receives a Request from the client containing:	oo i v olloni
	URL set to '/	
	ngsi-ld/v1/entities/	
	method set to 'POST'	
	Request Header['Content-	
	Type'] set to 'application/json' and	
	payload defined in file: 'building-simple-	
	attributes-sample.json'	
	attributes-sample.json	
	thon	SUT -> Client
	then {	SUT -> Client
	the SUT sends a valid Response for the operation:	
	Retrieve Entity by Id with Check Response Body containing an Attribute	
	set to and	
	Query Parameter: expected_attribute_name set to 'almostFull' and	
	Query Parameter: response_body set to 'response1.json()'	
TD D	TD (NOON LD (OVD /E (OO L OE OO	
TP Permutation id	TP/NGSI-LD/CI/Prov/E/001_05_02	
Name	001_05_02 Create one entity using the default context with JSON content type	e and request
	with context	
Parent Release	v1.3.1	
Description	Check that the default @context is used if the Content-Type header is "applicated by the Content Conte	
	the Link header does not contain a JSON-LD @context, requesting with conte	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/'	
	method set to 'POST'	
	Request Header['Content-	
	Type'] set to 'application/json' and	
	payload defined in file: 'building-simple-	
	attributes-sample.json'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operation:	3.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0
	Retrieve Entity by Id with Check Response Body containing an Attribute	
	set to and	
	Query Parameter: expected_attribute_name set to 'ngsi-	
	Id:default-context/almostFull and	
	Query Parameter: response_body set to 'response1.json()'	
	\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\\	
	I.	<u> </u>

TP Id	TP/NGSI-LD/CI/Prov/E/001_01
Test objective	Check that one can create an entity
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.1
Config Id	CF_01
PICS Selection	PICS_EntCre
Initial conditions	with { the SUT containing an initial state }
Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/E/001 01 01

Name	001_01_01 MinimalEntity	
Parent Release	v1.3.1	
Description	Check that one can create an entity	
		Direction
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entities/' method set to 'POST' Request Header['Content- Type'] set to 'application/json' and payload defined in file: 'building-minimal- sample.json' } then { the SUT sends a valid Response for the operations: Create Entity Selecting Content Type with Response Status Code set to 201 and Create Entity Selecting Content Type with Response Header: Location containing \$\${entity_id} and Retrieve Entity by Id with Check Created Resource Set To and Query Parameter: 'created_resource' set to 'created_entity' and Query Parameter: 'response_body' set to 'response1.json()' and Query Parameter: 'ignored_keys' set to '\$	Direction SUT <- Client SUT -> Client
	{None}' list of keys	
TD Democratica id	} 	
TP Permutation id	TP/NGSI-LD/CI/Prov/E/001_01_02	
Name Parent Release	001_01_02 EntityWithSimpleProperties v1.3.1	
Description		
Expected behaviour	Check that one can create an entity Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entities/' method set to 'POST' Request Header['Content- Type'] set to 'application/Id+json' and payload defined in file: 'building-simple- attributes-sample.jsonId' } then { the SUT sends a valid Response for the operations: Create Entity Selecting Content Type with Response Status Code set to 201 and Create Entity Selecting Content Type with Response Header: Location containing \$\${entity_id} and Retrieve Entity by Id with Check Created Resource Set To and	SUT -> Client
TP Permutation id	TP/NGSI-LD/CI/Prov/E/001_01_03	
Name	001_01_03 EntityWithRelationshipsProperties	
Parent Release	v1.3.1	
Description	Check that one can create an entity	
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/' method set to 'POST' Request Header!'Content-	SUT <- Client
	Request Header['Content-	

		1
	Type'] set to 'application/ld+json' and	
	payload defined in file: 'building-	
	relationship-of-property-sample.jsonld'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Create Entity Selecting Content Type with Response Status Code set	
	to	
	201 and	
	Create Entity Selecting Content Type with Response Header: Location	
	containing	
	\$\${entity_id} and	
	Retrieve Entity by Id with Check Created Resource Set To and	
	Query Parameter: 'created_resource' set to 'created_entity' and	
	Query Parameter: 'response_body' set to 'response1.json()' and	
	Query Parameter: 'ignored_keys' set to '\$	
	{None}' list of keys	
	(Notice) list of keys	
TP Permutation id	TP/NGSI-LD/CI/Prov/E/001_01_04	l
Name	001_01_04 EntityWithLocationAttribute	
Parent Release	v1.3.1	
Description	Check that one can create an entity	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
		SOT < CHEFT
	the SUT receives a Request from the client containing:	SOT < SHELL
	the SUT receives a Request from the client containing: URL set to '/	OUT V- Offerit
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/'	SOT <- OHER
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/' method set to 'POST'	OUT C SHETT
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/' method set to 'POST' Request Header['Content-	OUT CONGIN
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/' method set to 'POST' Request Header['Content- Type'] set to 'application/ld+json' and	OUT CONCIN
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/' method set to 'POST' Request Header['Content- Type'] set to 'application/ld+json' and payload defined in file: 'building-location-	OOT CONCIN
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/' method set to 'POST' Request Header['Content- Type'] set to 'application/ld+json' and	OOT CONCIN
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/' method set to 'POST' Request Header['Content- Type'] set to 'application/ld+json' and payload defined in file: 'building-location- attribute-sample.jsonld' }	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/' method set to 'POST' Request Header['Content- Type'] set to 'application/ld+json' and payload defined in file: 'building-location- attribute-sample.jsonld' } then {	SUT -> Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/' method set to 'POST' Request Header['Content- Type'] set to 'application/ld+json' and payload defined in file: 'building-location- attribute-sample.jsonId' } then { the SUT sends a valid Response for the operations:	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entities/' method set to 'POST' Request Header['Content- Type'] set to 'application/Id+json' and payload defined in file: 'building-location- attribute-sample.jsonId' } then { the SUT sends a valid Response for the operations: Create Entity Selecting Content Type with Response Status Code set	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entities/' method set to 'POST' Request Header['Content- Type'] set to 'application/Id+json' and payload defined in file: 'building-location- attribute-sample.jsonId' } then { the SUT sends a valid Response for the operations: Create Entity Selecting Content Type with Response Status Code set to	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entities/' method set to 'POST' Request Header['Content- Type'] set to 'application/Id+json' and payload defined in file: 'building-location- attribute-sample.jsonId' } then { the SUT sends a valid Response for the operations: Create Entity Selecting Content Type with Response Status Code set to 201 and	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entities/' method set to 'POST' Request Header['Content- Type'] set to 'application/Id+json' and payload defined in file: 'building-location- attribute-sample.jsonId' } then { the SUT sends a valid Response for the operations: Create Entity Selecting Content Type with Response Status Code set to 201 and Create Entity Selecting Content Type with Response Header: Location	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entities/' method set to 'POST' Request Header['Content- Type'] set to 'application/Id+json' and payload defined in file: 'building-location- attribute-sample.jsonId' } then { the SUT sends a valid Response for the operations: Create Entity Selecting Content Type with Response Status Code set to 201 and Create Entity Selecting Content Type with Response Header: Location containing	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/' method set to 'POST' Request Header['Content- Type'] set to 'application/ld+json' and payload defined in file: 'building-location- attribute-sample.jsonId' } then { the SUT sends a valid Response for the operations: Create Entity Selecting Content Type with Response Status Code set to 201 and Create Entity Selecting Content Type with Response Header: Location containing \$\${entity_id} and	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/' method set to 'POST' Request Header['Content- Type'] set to 'application/ld+json' and payload defined in file: 'building-location- attribute-sample.jsonId' } then { the SUT sends a valid Response for the operations: Create Entity Selecting Content Type with Response Status Code set to 201 and Create Entity Selecting Content Type with Response Header: Location containing \$\${entity_id} and Retrieve Entity by Id with Check Created Resource Set To and	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/' method set to 'POST' Request Header['Content- Type'] set to 'application/ld+json' and payload defined in file: 'building-location- attribute-sample.jsonId' } then { the SUT sends a valid Response for the operations: Create Entity Selecting Content Type with Response Status Code set to 201 and Create Entity Selecting Content Type with Response Header: Location containing \$\${entity_id} and	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/' method set to 'POST' Request Header['Content- Type'] set to 'application/ld+json' and payload defined in file: 'building-location- attribute-sample.jsonId' } then { the SUT sends a valid Response for the operations: Create Entity Selecting Content Type with Response Status Code set to 201 and Create Entity Selecting Content Type with Response Header: Location containing \$\${entity_id} and Retrieve Entity by Id with Check Created Resource Set To and	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/' method set to 'POST' Request Header['Content- Type'] set to 'application/ld+json' and payload defined in file: 'building-location- attribute-sample.jsonld' } then { the SUT sends a valid Response for the operations: Create Entity Selecting Content Type with Response Status Code set to 201 and Create Entity Selecting Content Type with Response Header: Location containing \$\${entity_id} and Retrieve Entity by Id with Check Created Resource Set To and Query Parameter: 'created_resource' set to 'created_entity' and	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/' method set to 'POST' Request Header['Content- Type'] set to 'application/ld+json' and payload defined in file: 'building-location- attribute-sample.jsonld' } then { the SUT sends a valid Response for the operations: Create Entity Selecting Content Type with Response Status Code set to 201 and Create Entity Selecting Content Type with Response Header: Location containing \$\${entity_id} and Retrieve Entity by Id with Check Created Resource Set To and Query Parameter: 'created_resource' set to 'created_entity' and Query Parameter: 'response_body' set to 'response1.json()' and	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/' method set to 'POST' Request Header['Content- Type'] set to 'application/ld+json' and payload defined in file: 'building-location- attribute-sample.jsonld' } then { the SUT sends a valid Response for the operations: Create Entity Selecting Content Type with Response Status Code set to 201 and Create Entity Selecting Content Type with Response Header: Location containing \$\${entity_id} and Retrieve Entity by Id with Check Created Resource Set To and Query Parameter: 'created_resource' set to 'created_entity' and Query Parameter: 'response_body' set to 'response1.json()' and Query Parameter: 'ignored_keys' set to '\$	

TP Id	TP/NGSI-LD/CI/Prov/E/001 03	
Test objective	Check that one cannot create an entity with an existing id	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.1	
Config Id	CF_01	
PICS Selection	PICS_EntCre	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/E/001_03_01	
Name	001_03_01 Create one valid entity and one invalid entity	
Parent Release	v1.3.1	
Description	Check that one cannot create an entity with an existing id	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/'	
	method set to 'POST'	
	Request Header['Content-	

Type'] set to 'application/ld+json' and payload defined in file: 'building-minimal-sample.jsonId'	
}	
then { the SUT sends a valid Response for the operations: Create Entity Selecting Content Type with Response Status Code set to 201 and Create Entity Selecting Content Type with Response Status Code set to 409 and Create Entity Selecting Content Type with Response Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/ AlreadyExists' and Create Entity Selecting Content Type with Response body containing 'title' element Type with Response body containing 'title' element	SUT -> Client

TP ld	TP/NGSI-LD/CI/Prov/E/001_04	
Test objective	Check that the @context is obtained from a Link Header if the Content-Type header is	
•	"application/json"	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 6.3.5	
Config Id	CF_01	
PICS Selection	PICS_HTTP_CONTEXT_RESOLV	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/E/001_04_01	
Name	001_04_01 Create one entity using a provided Link header with JSON content	type and
	NGSILD context	71
Parent Release	v1.3.1	
Description	Check that the @context is obtained from a Link Header if the Content-Type h	eader is
	"application/json"	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/'	
	method set to 'POST'	
	Request Header['Content-	
	Type'] set to 'application/json' and	
	payload defined in file: 'building-simple-	
	attributes-sample.json'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operation:	
	Retrieve Entity by Id with Check Response Body containing an Attribute	
	set to and	
	Query Parameter: expected_attribute_name set to 'almostFull' and	
	Query Parameter: response_body set to 'response1.json()'	
	}	
TP Permutation id	TP/NGSI-LD/CI/Prov/E/001_04_02	
Name	001_04_02 Create one entity using a provided Link header with JSON content	type and no
	context	
Parent Release	v1.3.1	
Description	Check that the @context is obtained from a Link Header if the Content-Type h	eader is
	"application/json"	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/'	
	method set to 'POST'	
	Request Header['Content- Type'] set to 'application/json' and	

	payload defined in file: 'building-simple- attributes-sample.json' }	
:	then { the SUT sends a valid Response for the operation: Retrieve Entity by Id with Check Response Body containing an Attribute set to and	SUT -> Client

TP Id	TP/NGSI-LD/CI/Prov/E/001_06	
Test objective	Check that an HTTP error response of type BadRequestData is raised if the C	
	header is "application/json" and the request payload body (as JSON) contains	a "@context"
	term	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 6.3.5	
Config Id	CF_01	
PICS Selection	PICS_HTTP_CONTEXT_RESOLV	
Initial conditions	with { the SUT containing an initial state }	
	Permutations (7 to 20 to	
TP Permutation id	TP/NGSI-LD/CI/Prov/E/001_06_01	
Name	001_06_01 Create one entity containing a JSON-LD @context with a JSON context with a J	ontent type
Parent Release	v1.3.1	
Description	Check that an HTTP error response of type BadRequestData is raised if the C	
	header is "application/json" and the request payload body (as JSON) contains	a "@context"
Expected behaviour	Test events	Direction
Expected behaviour		SUT <- Client
	when {	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/	
	ngsi-Id/v1/entities/'	
	method set to 'POST'	
	Request Header['Content-	
	Type'] set to 'application/json' and	
	payload defined in file: 'building-simple-	
	attributes-sample.jsonld'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Create Entity Selecting Content Type with Response Status Code set	
	to	
	400 and	
	Create Entity Selecting Content Type with Response Body containing	
	the type 'https://	
	uri.etsi.org/ngsi-ld/errors/	
	BadRequestData' and	
	Create Entity Selecting Content Type with Response body containing	
	'title' element	
	I)	

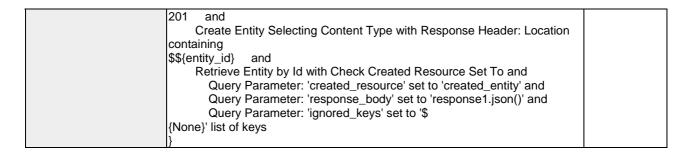
TP ld	TP/NGSI-LD/CI/Prov/E/001_02
Test objective	Check that one cannot create an entity with an invalid request
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.1
Config Id	CF_01
PICS Selection	PICS_EntCre
Initial conditions	with { the SUT containing an initial state }
	Permutations
TP Permutation id	TP/NGSI-LD/CI/Prov/E/001_02_01
Name	001_02_01 InvalidJson
Parent Release	v1.3.1
Description	Check that one cannot create an entity with an invalid request

Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/	
	method set to 'POST'	
	Request creation of an entity from filename 'invalid-	
	json-sample.jsonId' and Content-	
	Type set to 'application/ld+json'	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	OOT > OHOTIC
	Create Entity From File with Response Status Code set to	
	400 and	
	Create Entity From File with Response Body containing the type	
	'https://	
	uri.etsi.org/ngsi-ld/errors/	
	InvalidRequest' and	
	Create Entity From File with Response body containing 'title' element	
TP Permutation id	TP/NGSI-LD/CI/Prov/E/001_02_02	
Name	001_02_02 EmptyJson	
Parent Release	v1.3.1	
Description	Check that one cannot create an entity with an invalid request	
Expected behaviour	Test events	Direction
•	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/	
	method set to 'POST'	
	Request creation of an entity from filename 'empty-	
	sample.jsonld' and Content-	
	Type set to 'application/ld+json'	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	OOT > OHOTIC
	Create Entity From File with Response Status Code set to	
	400 and	
	Create Entity From File with Response Body containing the type	
	'https://	
	uri.etsi.org/ngsi-ld/errors/	
	InvalidRequest' and	
	Create Entity From File with Response body containing 'title' element	
TP Permutation id	TP/NGSI-LD/CI/Prov/E/001_02_03	
Name	001_02_03 EntityWithNoContext	
Parent Release	V1.3.1	
Description	Check that one cannot create an entity with an invalid request	
Expected behaviour	Test events	Direction
•	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/	
	method set to 'POST'	
	Request creation of an entity from filename 'building-	
	minimal-sample.json' and Content-	
	Type set to 'application/ld+json'	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	Jos Silent
	Create Entity From File with Response Status Code set to	
	400 and	
	Create Entity From File with Response Body containing the type	
	'https://	
	uri.etsi.org/ngsi-ld/errors/	
	BadRequestData' and	
	Create Entity From File with Response body containing 'title' element	
	1)	

TP Id TP/NGSI-LD/CI/Prov/E/001_11 Test objective Check that one can create an entity with one or more scopes Reference ETSI GS CIM 009 V1.5.1 [1], clause 4.18 Config Id CF_01 PICS Sclection PICS_SCOPES Initial conditions with { the SUT containing an initial state } Permutations TP Permutation id TP/NGSI-LD/CI/Prov/E/001_11_01 Name 001_11_01 EntityWithOneScope Parent Release V1.5.1 Description Check that one can create an entity with one or more scopes Expected behaviour Test events Direct When { the SUT receives a Request from the client containing:	Client
Reference ETSI GS CIM 009 V1.5.1 [1], clause 4.18 Config Id CF_01 PICS_SCOPES Initial conditions with { the SUT containing an initial state } Permutations TP Permutation id TP/NGSI-LD/CI/Prov/E/001_11_01 Name 001_11_01 EntityWithOneScope Parent Release v1.5.1 Description Check that one can create an entity with one or more scopes Expected behaviour Test events Direct when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/n/Initities/' method set to 'POST' Request Header['Content- Type'] set to 'application/json' and payload defined in file: 'building-minimal-with-one-scope.json' } then { the SUT sends a valid Response for the operations: Create Entity Selecting Content Type with Response Status Code set to 201 and Create Entity Selecting Content Type with Response Header: Location containing \$\${entity_id}\$ and Retrieve Entity by Id with Check Created Resource Set To and Query Parameter: 'ignored_keys' set to '\$	Client
Config Id	Client
PICS_SCOPES Initial conditions with { the SUT containing an initial state }	Client
Initial conditions with { the SUT containing an initial state } Permutations	Client
TP Permutation id TP/NGSI-LD/CI/Prov/E/001_11_01 Name 001_11_01 EntityWithOneScope Parent Release v1.5.1 Description Check that one can create an entity with one or more scopes Expected behaviour Test events Direct when { the SUT receives a Request from the client containing: URL set to '/ nethod set to 'POST' Request Header['Content- Type'] set to 'application/json' and payload defined in file: 'building-minimal-with-one-scope.json' } then { the SUT sends a valid Response for the operations: Create Entity Selecting Content Type with Response Status Code set to 201 and Create Entity Selecting Content Type with Response Header: Location containing \$\${entity_id} and Retrieve Entity by Id with Check Created Resource Set To and Query Parameter: 'ignored_keys' set to '\$	Client
TP Permutation id Name O01_11_01 EntityWithOneScope Parent Release v1.5.1 Description Check that one can create an entity with one or more scopes Test events When { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entities/' method set to 'POST' Request Header['Content- Type'] set to 'application/json' and payload defined in file: 'building-minimal-with-one-scope.json' } then { the SUT sends a valid Response for the operations: Create Entity Selecting Content Type with Response Status Code set to 201 and Create Entity Selecting Content Type with Response Header: Location containing \${entity_id} and Retrieve Entity by Id with Check Created Resource Set To and Query Parameter: 'ignored_keys' set to '\$	Client
Name	Client
Parent Release V1.5.1 Description Check that one can create an entity with one or more scopes Test events Direct	Client
Check that one can create an entity with one or more scopes	Client
## Test events When { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/' method set to 'POST' Request Header['Content-Type'] set to 'application/json' and payload defined in file: 'building-minimal-with-one-scope.json' }	Client
when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/entities/' method set to 'POST' Request Header['Content- Type'] set to 'application/json' and payload defined in file: 'building-minimal- with-one-scope.json' } then { the SUT sends a valid Response for the operations: Create Entity Selecting Content Type with Response Status Code set to 201 and Create Entity Selecting Content Type with Response Header: Location containing \$\${entity_id} and Retrieve Entity by Id with Check Created Resource Set To and Query Parameter: 'ignored_keys' set to '\$	Client
(Notice) list of keys	
TP Permutation id TP/NGSI-LD/CI/Prov/E/001_11_02	
Name 001_11_02 EntityWithManyScopes	
Parent Release v1.5.1	
Description Check that one can create an entity with one or more scopes	
Expected behaviour Test events Direct	tion
when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/' method set to 'POST' Request Header['Content- Type'] set to 'application/json' and payload defined in file: 'building-minimal- with-many-scopes.json' } then { the SUT sends a valid Response for the operations: Create Entity Selecting Content Type with Response Status Code set to 201 and Create Entity Selecting Content Type with Response Header: Location containing \$\${entity_id} and Retrieve Entity by Id with Check Created Resource Set To and	
Query Parameter: 'ignored_keys' set to '\$ {None}' list of keys	

TP ld	TP/NGSI-LD/CI/Prov/E/001_08	
Test objective	Check that an HTTP error response of type BadRequestData is raised if the Cheader is "application/ld+json" and the request payload body does not contain term	Content-Type a @context
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 6.3.5	
Config Id	CF_01	
PICS Selection	PICS_HTTP_CONTEXT_RESOLV	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/E/001_08_01	
Name	001_08_01 Create one entity not containing a JSON-LD @context with a JSC type	N-LD content
Parent Release	v1.3.1	
Description	Check that an HTTP error response of type BadRequestData is raised if the Cheader is "application/ld+json" and the request payload body does not contain term	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/' method set to 'POST' Request Header['Content- Type'] set to 'application/ld+json' and payload defined in file: 'building-simple- attributes-sample.json' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Create Entity Selecting Content Type with Response Status Code set to 400 and Create Entity Selecting Content Type with Response Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/ BadRequestData' and Create Entity Selecting Content Type with Response body containing 'title' element }	SUT -> Client

TP Id	TD/NCCLL D/CI/Drov/E/004_40		
	TP/NGSI-LD/CI/Prov/E/001_10		
Test objective	Check that one can create an entity with a LanguageProperty property		
Reference	ETSI GS CIM 009 V1.4.1 [1], clauses 4.5.18 and 5.6.1		
Config Id	CF_01		
PICS Selection	PICS_LANG and PICS_EntCre		
Initial conditions	with { the SUT containing an initial state }		
Permutations			
TP Permutation id	TP/NGSI-LD/CI/Prov/E/001_10_01		
Name	001_10_01 EntityWithLanguageProperty		
Parent Release	v1.4.1		
Description	Check that one can create an entity with a LanguageProperty property		
Expected behaviour	Test events	Direction	
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/' method set to 'POST' Request Header['Content- Type'] set to 'application/ld+json' and payload defined in file: 'building-language- property-sample.jsonId' } then { the SUT sends a valid Response for the operations: Create Entity Selecting Content Type with Response Status Code set	SUT <- Client	
	to		



TP ld	TP/NGSI-LD/CI/Prov/E/001_09			
Test objective	Check that an HTTP error response of type BadRequestData is raised if the Content-Type			
	header is "application/ld+json" and a JSON-LD Link header is present in the i	ncoming HTTP		
	request			
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 6.3.5			
Config Id	CF_01			
PICS Selection	PICS_HTTP_CONTEXT_RESOLV			
Initial conditions	with { the SUT containing an initial state }			
Permutations				
TP Permutation id	TP/NGSI-LD/CI/Prov/E/001_09_01			
Name	001_09_01 Create one entity with a Link header and a JSON-LD content type	9		
Parent Release	v1.3.1			
Description	Check that an HTTP error response of type BadRequestData is raised if the C			
	header is "application/ld+json" and a JSON-LD Link header is present in the i	ncoming HTTP		
	request			
Expected behaviour	Test events	Direction		
	when {	SUT <- Client		
	the SUT receives a Request from the client containing:			
	URL set to '/			
	ngsi-ld/v1/entities/			
	method set to 'POST'			
	Request Header['Content-			
	Type'] set to 'application/ld+json' and			
	payload defined in file: 'building-simple-			
	attributes-sample.jsonld'			
	}			
	then {	SUT -> Client		
	the SUT sends a valid Response for the operations:			
	Create Entity Selecting Content Type with Response Status Code set			
	to			
	400 and			
	Create Entity Selecting Content Type with Response Body containing			
	the type 'https://			
	uri.etsi.org/ngsi-ld/errors/			
	BadRequestData' and			
	Create Entity Selecting Content Type with Response body containing			
	'title' element			
	1)			

4.4.3.4.2 DeleteEntity

TP Id	TP/NGSI-LD/CI/Prov/E/002_03	
Test objective	Check that one cannot delete an entity if the entity id is not known to the system	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.6	
Config Id	CF_01	
PICS Selection	PICS_EntDel	
Initial conditions	with { the SUT containing an initial state }	
Permutations		
TP Permutation id	TP/NGSI-LD/CI/Prov/E/002_03_01	
Name	002_03_01 Delete an entity with an id not known to the system	
Parent Release	v1.3.1	
Description	Check that one cannot delete an entity if the entity id is not known to the system	

Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/{id}'	
	method set to 'DELETE'	
	Delete Entity Request with id set to '\$	
	{entity_id}'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Delete Entity by Id with Response Status Code set to	
	404 and	
	Delete Entity by Id with Response Body containing the type 'https://	
	uri.etsi.org/ngsi-ld/errors/	
	ResourceNotFound' and	
	Delete Entity by Id with Response body containing 'title' element	
	}	

TP ld	TP/NGSI-LD/CI/Prov/E/002_01	
Test objective	Check that one can delete an entity by id	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.6	
Config Id	CF_01	
PICS Selection	PICS_EntDel	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/E/002_01_01	
Name	002_01_01 Delete an entity	
Parent Release	v1.3.1	
Description	Check that one can delete an entity by id	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/entities/' method set to 'POST' Request Header['Content- Type'] set to 'application/ld+json' and payload defined in file: 'building-simple- attributes-sample.jsonId' }	SUT <- Client
	then { the SUT sends a valid Response for the operations: Create Entity Selecting Content Type with Response Status Code set to 201 and Delete Entity by Id with Response Status Code set to 204 and Retrieve Entity by Id with Response Status Code set to \${response2.status_code} }	SUT -> Client

TP ld	TP/NGSI-LD/CI/Prov/E/002_02
Test objective	Check that one cannot delete an entity with invalid/missing id
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.6
Config Id	CF_01
PICS Selection	PICS_EntDel
Initial conditions	with { the SUT containing an initial state }
	Permutations
TP Permutation id	TP/NGSI-LD/CI/Prov/E/002_02_01
Name	002_02_01 Delete an entity if the Entity Id is not present
Parent Release	v1.3.1
Description	Check that one cannot delete an entity with invalid/missing id

Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/{id}'	
	method set to 'DELETE'	
	Delete Entity Request with id set to '\$	
	{EMPTY}'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Delete Entity by Id with Response Status Code set to	
	405 and	
	Delete Entity by Id with Response Body containing the type '\$	
	{EMPTY}' and	
	Delete Entity by Id with Response body containing 'title' element	
	}	
TP Permutation id	TP/NGSI-LD/CI/Prov/E/002_02_02	
Name	002_02_02 Delete an entity if the Entity Id is not a valid URI	
Parent Release	v1.3.1	
Description	Check that one cannot delete an entity with invalid/missing id	1 5: .:
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/entities/{id}'	
	method set to 'DELETE'	
	Delete Entity Request with id set to 'thisisaninvaliduri'	
	than (SUT -> Client
	then {	SUT -> Client
	the SUT sends a valid Response for the operations: Delete Entity by Id with Response Status Code set to	
	400 and	
	Delete Entity by Id with Response Body containing the type 'https://	
	uri.etsi.org/ngsi-ld/errors/	
	BadRequestData' and	
	Delete Entity by Id with Response body containing 'title' element	
	n	1

4.4.3.5 TemporalEntityAttributes

4.4.3.5.1 DeleteAttribute

TP Id	TP/NGSI-LD/CI/Prov/TEA/015 03	
Test objective	Check that an error is raised if one deletes a temporal entity with an unknown	
	EntityId/Attribute Id	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.13	
Config Id	CF_01	
PICS Selection	PICS_AttTemEvoEntDel	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing an initial Tem	poral Entity
	\${entity} with an id set to \${temporal_entity_representation_id} }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/TEA/015_03_01	
Name	015_03_01 Delete an attribute to a temporal entity if the entity id does not exist	st
Parent Release	v1.3.1	
Description	Check that an error is raised if one deletes a temporal entity with an unknown	
	EntityId/Attribute Id	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/temporal/entities/{entityId}/attrs/{attributeId}?\$	
	{params_as_string}'	
	method set to 'DELETE'	
	Delete Attribute From Temporal Entity: and	

	Query Parameter: entityId set to '\$	
	{unknown_temporal_entity_id}' and	
	Query Parameter: attributeld set to 'fuelLevel' and	
	Query Parameter: content_type set to 'application/	
	ison' and	
	Query Parameter: datasetId set to '\$	
	{EMPTY}' and	
	Query Parameter: deleteAll set to 'false' and	
	Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	jsonld-contexts/ngsi-ld-test-suite-compound.jsonld'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Delete Attribute From Temporal Entity with Response Status Code set	
	lto	
	404 and	
	Delete Attribute From Temporal Entity with Response Body containing	
	the type 'https://	
	uri.etsi.org/ngsi-ld/errors/	
	Delete Attribute From Temporal Entity with Response body containing	
	'title' element	
	}	
TP Permutation id	TP/NGSI-LD/CI/Prov/TEA/015_03_02	
Name	015_03_02 Delete an attribute to a temporal entity if the entity does not contain	n the target
	attribute	
Parent Release	v1.3.1	
Description	Check that an error is raised if one deletes a temporal entity with an unknown	
	EntityId/Attribute Id	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	· ·	SUT <- Client
	the SUT receives a Request from the client containing:	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{entityId}/attrs/{attributeId}?\$	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{entityId}/attrs/{attributeId}?\$ {params_as_string}'	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{entityId}/attrs/{attributeId}?\$ {params_as_string}' method set to 'DELETE'	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{entityId}/attrs/{attributeId}?\$ {params_as_string}' method set to 'DELETE' Delete Attribute From Temporal Entity: and	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{entityId}/attrs/{attributeId}?\$ {params_as_string}' method set to 'DELETE' Delete Attribute From Temporal Entity: and Query Parameter: entityId set to '\$	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{entityId}/attrs/{attributeId}?\$ {params_as_string}' method set to 'DELETE' Delete Attribute From Temporal Entity: and Query Parameter: entityId set to '\$ {valid_temporal_entity_id}' and	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{entityId}/attrs/{attributeId}?\$ {params_as_string}' method set to 'DELETE' Delete Attribute From Temporal Entity: and Query Parameter: entityId set to '\$ {valid_temporal_entity_id}' and Query Parameter: attributeId set to 'notExistingAttribute' and	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{entityId}/attrs/{attributeId}?\$ {params_as_string}' method set to 'DELETE' Delete Attribute From Temporal Entity: and Query Parameter: entityId set to '\$ {valid_temporal_entity_id}' and	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{entityId}/attrs/{attributeId}?\$ {params_as_string}' method set to 'DELETE' Delete Attribute From Temporal Entity: and Query Parameter: entityId set to '\$ {valid_temporal_entity_id}' and Query Parameter: attributeId set to 'notExistingAttribute' and	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{entityld}/attrs/{attributeId}?\$ {params_as_string}' method set to 'DELETE' Delete Attribute From Temporal Entity: and Query Parameter: entityld set to '\$ {valid_temporal_entity_id}' and Query Parameter: attributeId set to 'notExistingAttribute' and Query Parameter: content_type set to 'application/	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{entityld}/attrs/{attributeId}?\$ {params_as_string}' method set to 'DELETE' Delete Attribute From Temporal Entity: and Query Parameter: entityld set to '\$ {valid_temporal_entity_id}' and Query Parameter: attributeId set to 'notExistingAttribute' and Query Parameter: content_type set to 'application/ json' and	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{entityld}/attrs/{attributeId}?\$ {params_as_string}' method set to 'DELETE' Delete Attribute From Temporal Entity: and Query Parameter: entityld set to '\$ {valid_temporal_entity_id}' and Query Parameter: attributeId set to 'notExistingAttribute' and Query Parameter: content_type set to 'application/ json' and Query Parameter: datasetId set to '\$ {EMPTY}' and	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities/{entityId}/attrs/{attributeId}?\$ {params_as_string}' method set to 'DELETE' Delete Attribute From Temporal Entity: and Query Parameter: entityId set to '\$ {valid_temporal_entity_id}' and Query Parameter: attributeId set to 'notExistingAttribute' and Query Parameter: content_type set to 'application/ json' and Query Parameter: datasetId set to '\$ {EMPTY}' and Query Parameter: deleteAll set to 'false' and	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities/{entityId}/attrs/{attributeId}?\$ {params_as_string}' method set to 'DELETE' Delete Attribute From Temporal Entity: and Query Parameter: entityId set to '\$ {valid_temporal_entity_id}' and Query Parameter: attributeId set to 'notExistingAttribute' and Query Parameter: content_type set to 'application/ json' and Query Parameter: datasetId set to '\$ {EMPTY}' and Query Parameter: deleteAll set to 'false' and Query Parameter: context set to 'https://	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities/{entityId}/attrs/{attributeId}?\$ {params_as_string}' method set to 'DELETE' Delete Attribute From Temporal Entity: and Query Parameter: entityId set to '\$ {valid_temporal_entity_id}' and Query Parameter: attributeId set to 'notExistingAttribute' and Query Parameter: content_type set to 'application/ json' and Query Parameter: datasetId set to '\$ {EMPTY}' and Query Parameter: deleteAll set to 'false' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities/{entityId}/attrs/{attributeId}?\$ {params_as_string}' method set to 'DELETE' Delete Attribute From Temporal Entity: and Query Parameter: entityId set to '\$ {valid_temporal_entity_id}' and Query Parameter: attributeId set to 'notExistingAttribute' and Query Parameter: content_type set to 'application/ json' and Query Parameter: datasetId set to '\$ {EMPTY}' and Query Parameter: deleteAll set to 'false' and Query Parameter: context set to 'https://	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{entityld}/attrs/{attributeId}?\$ {params_as_string}' method set to 'DELETE' Delete Attribute From Temporal Entity: and Query Parameter: entityld set to '\$ {valid_temporal_entity_id}' and Query Parameter: attributeId set to 'notExistingAttribute' and Query Parameter: content_type set to 'application/ json' and Query Parameter: datasetId set to '\$ {EMPTY}' and Query Parameter: deleteAll set to 'false' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld' }	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities/{entityId}/attrs/{attributeId}?\$ {params_as_string}' method set to 'DELETE' Delete Attribute From Temporal Entity: and Query Parameter: entityId set to '\$ {valid_temporal_entity_id}' and Query Parameter: attributeId set to 'notExistingAttribute' and Query Parameter: content_type set to 'application/ json' and Query Parameter: datasetId set to '\$ {EMPTY}' and Query Parameter: deleteAll set to 'false' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite-/raw/develop/resources/ jsonId-contexts/ngsi-Id-test-suite-compound.jsonId' } then {	SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities/{entityId}/attrs/{attributeId}?\$ {params_as_string}' method set to 'DELETE' Delete Attribute From Temporal Entity: and Query Parameter: entityId set to '\$ {valid_temporal_entity_id}' and Query Parameter: attributeId set to 'notExistingAttribute' and Query Parameter: content_type set to 'application/ json' and Query Parameter: datasetId set to '\$ {EMPTY}' and Query Parameter: deleteAll set to 'false' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/ jsonId-contexts/ngsi-Id-test-suite-compound.jsonId' } then { the SUT sends a valid Response for the operations:	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities/{entityId}/attrs/{attributeId}?\$ {params_as_string}' method set to 'DELETE' Delete Attribute From Temporal Entity: and Query Parameter: entityId set to '\$ {valid_temporal_entity_id}' and Query Parameter: attributeId set to 'notExistingAttribute' and Query Parameter: content_type set to 'application/ json' and Query Parameter: datasetId set to '\$ {EMPTY}' and Query Parameter: deleteAll set to 'false' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/ jsonId-contexts/ngsi-Id-test-suite-compound.jsonId' } then { the SUT sends a valid Response for the operations: Delete Attribute From Temporal Entity with Response Status Code set	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities/{entityId}/attrs/{attributeId}?\$ {params_as_string}' method set to 'DELETE' Delete Attribute From Temporal Entity: and Query Parameter: entityId set to '\$ {valid_temporal_entity_id}' and Query Parameter: attributeId set to 'notExistingAttribute' and Query Parameter: content_type set to 'application/ json' and Query Parameter: datasetId set to '\$ {EMPTY}' and Query Parameter: deleteAll set to 'false' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/ jsonId-contexts/ngsi-Id-test-suite-compound.jsonId' } then { the SUT sends a valid Response for the operations: Delete Attribute From Temporal Entity with Response Status Code set	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities/{entityId}/attrs/{attributeId}?\$ {params_as_string}' method set to 'DELETE' Delete Attribute From Temporal Entity: and Query Parameter: entityId set to '\$ {valid_temporal_entity_id}' and Query Parameter: attributeId set to 'notExistingAttribute' and Query Parameter: content_type set to 'application/ json' and Query Parameter: datasetId set to '\$ {EMPTY}' and Query Parameter: deleteAll set to 'false' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/ jsonId-contexts/ngsi-Id-test-suite-compound.jsonId' } then { the SUT sends a valid Response for the operations: Delete Attribute From Temporal Entity with Response Status Code set to 404 and	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{entityId}/attrs/{attributeId}?\$ {params_as_string}' method set to 'DELETE' Delete Attribute From Temporal Entity: and Query Parameter: entityId set to '\$ {valid_temporal_entity_id}' and Query Parameter: attributeId set to 'notExistingAttribute' and Query Parameter: content_type set to 'application/ json' and Query Parameter: datasetId set to '\$ {EMPTY}' and Query Parameter: deleteAll set to 'false' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/ jsonId-contexts/ngsi-Id-test-suite-compound.jsonId' } then { the SUT sends a valid Response for the operations: Delete Attribute From Temporal Entity with Response Body containing	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{entityId}/attrs/{attributeId}?\$ {params_as_string}' method set to 'DELETE' Delete Attribute From Temporal Entity: and Query Parameter: entityId set to '\$ {valid_temporal_entity_id}' and Query Parameter: attributeId set to 'notExistingAttribute' and Query Parameter: content_type set to 'application/ json' and Query Parameter: datasetId set to '\$ {EMPTY}' and Query Parameter: deleteAll set to 'false' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/ jsonId-contexts/ngsi-Id-test-suite-compound.jsonId' } then { the SUT sends a valid Response for the operations: Delete Attribute From Temporal Entity with Response Body containing the type 'https://	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities/{entityId}/attrs/{attributeId}?\$ {params_as_string}' method set to 'DELETE' Delete Attribute From Temporal Entity: and Query Parameter: entityId set to '\$ {valid_temporal_entity_id}' and Query Parameter: attributeId set to 'notExistingAttribute' and Query Parameter: content_type set to 'application/ json' and Query Parameter: datasetId set to '\$ {EMPTY}' and Query Parameter: deleteAll set to 'false' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/ jsonId-contexts/ngsi-Id-test-suite-compound.jsonId' } then { the SUT sends a valid Response for the operations: Delete Attribute From Temporal Entity with Response Status Code set to 404 and Delete Attribute From Temporal Entity with Response Body containing the type 'https:// uri.etsi.org/ngsi-Id/errors/	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{entityId}/attrs/{attributeId}?\$ {params_as_string}' method set to 'DELETE' Delete Attribute From Temporal Entity: and Query Parameter: entityId set to '\$ {valid_temporal_entity_id}' and Query Parameter: attributeId set to 'notExistingAttribute' and Query Parameter: content_type set to 'application/ json' and Query Parameter: datasetId set to '\$ {EMPTY}' and Query Parameter: deleteAll set to 'false' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/ jsonId-contexts/ngsi-Id-test-suite-compound.jsonId' } then { the SUT sends a valid Response for the operations: Delete Attribute From Temporal Entity with Response Status Code set to 404 and Delete Attribute From Temporal Entity with Response Body containing the type 'https:// uri.etsi.org/ngsi-Id/errors/ ResourceNotFound' and	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities/{entityId}/attrs/{attributeId}?\$ {params_as_string}' method set to 'DELETE' Delete Attribute From Temporal Entity: and Query Parameter: entityId set to '\$ {valid_temporal_entity_id}' and Query Parameter: attributeId set to 'notExistingAttribute' and Query Parameter: content_type set to 'application/ json' and Query Parameter: datasetId set to '\$ {EMPTY}' and Query Parameter: deleteAll set to 'false' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/ jsonId-contexts/ngsi-Id-test-suite-compound.jsonId' } then { the SUT sends a valid Response for the operations: Delete Attribute From Temporal Entity with Response Status Code set to 404 and Delete Attribute From Temporal Entity with Response Body containing the type 'https:// uri.etsi.org/ngsi-Id/errors/	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{entityId}/attrs/{attributeId}?\$ {params_as_string}' method set to 'DELETE' Delete Attribute From Temporal Entity: and Query Parameter: entityId set to '\$ {valid_temporal_entity_id}' and Query Parameter: attributeId set to 'notExistingAttribute' and Query Parameter: content_type set to 'application/ json' and Query Parameter: datasetId set to '\$ {EMPTY}' and Query Parameter: deleteAll set to 'false' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/ jsonId-contexts/ngsi-Id-test-suite-compound.jsonId' } then { the SUT sends a valid Response for the operations: Delete Attribute From Temporal Entity with Response Status Code set to 404 and Delete Attribute From Temporal Entity with Response Body containing the type 'https:// uri.etsi.org/ngsi-Id/errors/ ResourceNotFound' and	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities/{entityId}/attrs/{attributeId}?\$ {params_as_string}' method set to 'DELETE' Delete Attribute From Temporal Entity: and Query Parameter: entityId set to '\$ {valid_temporal_entity_id}' and Query Parameter: attributeId set to 'notExistingAttribute' and Query Parameter: content_type set to 'application/ json' and Query Parameter: datasetId set to '\$ {EMPTY}' and Query Parameter: deleteAll set to 'false' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/ jsonId-contexts/ngsi-Id-test-suite-compound.jsonId' } then { the SUT sends a valid Response for the operations: Delete Attribute From Temporal Entity with Response Body containing the type 'https:// uri.etsi.org/ngsi-Id/errors/ ResourceNotFound' and Delete Attribute From Temporal Entity with Response body containing	

TP Id	TP/NGSI-LD/CI/Prov/TEA/015_01
	Check that one can delete an attribute of a temporal representation of an entity with simple temporal properties
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.13

Config Id	CF_01	
PICS Selection	PICS_AttTemEvoEntDel	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing an initial Tem	nporal Entity
	\${entity} with an id set to \${temporal_entity_representation_id} } Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/TEA/015_01_01	
Name	015_01_01 Delete an attribute from a temporal representation of an entity wit	hout
	deleteAll/datasetId	
Parent Release	v1.3.1	
Description	Check that one can delete an attribute of a temporal representation of an enti-	ty with simple
Expected behaviour	temporal properties Test events	Direction
Expected beliavious	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/temporal/entities/{entityld}/attrs/{attributeld}?\$	
	{params_as_string}'	
	method set to 'DELETE' Delete Attribute From Temporal Entity: and	
	Query Parameter: entityld set to '\$	
	{temporal_entity_representation_id}' and	
	Query Parameter: attributeId set to 'fuelLevel' and	
	Query Parameter: content_type set to 'application/	
	json' and Query Parameter: datasetId set to '\$	
	{datasetid}' and	
	Query Parameter: deleteAll set to '\$	
	{deleteall}' and	
	Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	jsonld-contexts/ngsi-ld-test-suite-compound.jsonld'	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	OCT > CHOIN
	Delete Attribute From Temporal Entity with Response Status Code set	
	to	
	204 and	
	Retrieve Temporal Representation Of Entity with Check Updated Entity and	
	Query Parameter: 'updated_resource' set to	
	'temporal_entity_expectation_payload' and	
	Query Parameter: 'response_body' set to 'response1.json()' and	
	Query Parameter: 'ignored_keys' set to 'ignored_attributes'	
TP Permutation id	TP/NGSI-LD/CI/Prov/TEA/015_01_02	
Name	015_01_02 Delete an attribute from a temporal representation of an entity wit	h datasetId
Parent Release	v1.3.1	
Description	Check that one can delete an attribute of a temporal representation of an enti	ty with simple
	temporal properties	1
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing:	SUT <- Client
	URL set to '/	
	ngsi-ld/v1/temporal/entities/{entityld}/attrs/{attributeld}?\$	
	{params_as_string}'	
	method set to 'DELETE'	
	Delete Attribute From Temporal Entity: and	
	Query Parameter: entityId set to '\$ {temporal_entity_representation_id}' and	
	Query Parameter: attributeld set to 'fuelLevel' and	
	Query Parameter: content_type set to 'application/	
	json' and	
	Query Parameter: datasetId set to '\$	
	{datasetid}' and	
	Query Parameter: deleteAll set to '\$	
	{deleteall}' and Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	prorganationary/reproducting in tool duttor / raw/dovelop/resources/	1

	licented contexts/ngoi led test quite compound is and!	T
	jsonld-contexts/ngsi-ld-test-suite-compound.jsonld'	
	then { the SUT sends a valid Response for the operations: Delete Attribute From Temporal Entity with Response Status Code set	SUT -> Client
	to 204 and Retrieve Temporal Representation Of Entity with Check Updated Entity	
	and Query Parameter: 'updated_resource' set to	
	'temporal_entity_expectation_payload' and Query Parameter: 'response_body' set to 'response1.json()' and Query Parameter: 'ignored_keys' set to 'ignored_attributes'	
TP Permutation id	TP/NGSI-LD/CI/Prov/TEA/015_01_03	
Name	015_01_03 Delete an attribute from a temporal representation of an entity wit	h deleteAll
Parent Release	v1.3.1	
Description	Check that one can delete an attribute of a temporal representation of an enti temporal properties	
Expected behaviour	Test events	Direction SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{entityId}/attrs/{attributeId}?\$ {params_as_string}' method set to 'DELETE' Delete Attribute From Temporal Entity: and Query Parameter: entityId set to '\$ {temporal_entity_representation_id}' and Query Parameter: attributeId set to 'fuelLevel' and Query Parameter: content_type set to 'application/ json' and Query Parameter: datasetId set to '\$ {datasetid}' and Query Parameter: deleteAll set to '\$ {deleteall}' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonId-contexts/ngsi-ld-test-suite-compound.jsonId' } then /	SUT -> Client
	then { the SUT sends a valid Response for the operations: Delete Attribute From Temporal Entity with Response Status Code set to 204 and Retrieve Temporal Representation Of Entity with Check Updated Entity and Query Parameter: 'updated_resource' set to 'temporal_entity_expectation_payload' and Query Parameter: 'response_body' set to 'response1.json()' and Query Parameter: 'ignored_keys' set to 'ignored_attributes' }	SUI -> Client

TP Id	TP/NGSI-LD/CI/Prov/TEA/015_02		
Test objective	Check that an error is raised if one deletes an attribute to temporal entity with an		
	unknown/invalid Entity/Attribute Id		
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.13		
Config Id	CF_01		
PICS Selection	PICS_AttTemEvoEntDel		
Initial conditions	with { the SUT being in the "initial state" and the SUT containing an initial Temporal Entity \${entity} with an id set to \${temporal_entity_representation_id} and an attribute with an id set to \${atrld} }		
	Permutations		
TP Permutation id	TP/NGSI-LD/CI/Prov/TEA/015_02_01		
Name	015_02_01 Delete an attribute to a temporal representation of an entity with a missing entity		
	id		
Parent Release	v1.3.1		

Check that an error is raised if one deletes an attribute to temporal entity with an unknown/invalid Entity/Attribute Id	
Test events	Direction
unknown/invalid Entity/Attribute Id Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities/{entityId}/attrs/{attributeId}?\$ {params_as_string}' method set to 'DELETE' Delete Attribute From Temporal Entity: and Query Parameter: entityId set to '\$ {EMPTY}' and Query Parameter: attributeId set to 'speed' and Query Parameter: content_type set to 'application/ json' and Query Parameter: datasetId set to '\$ {EMPTY}' and Query Parameter: deleteAll set to 'false' and Query Parameter: context set to '\$ {EMPTY}' }	Direction SUT <- Client
then { the SUT sends a valid Response for the operations: Delete Attribute From Temporal Entity with Response Status Code set to 400 and Delete Attribute From Temporal Entity with Response Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/ BadRequestData' and Delete Attribute From Temporal Entity with Response body containing 'title' element	SUT -> Client
TD/NCCLLD/CI/Droy/TEA/045_02_02	
	na invalid antitu
	in invalid entity
Check that an error is raised if one deletes an attribute to temporal entity with	an
	Direction
when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{entityId}/attrs/{attributeId}?\$ {params_as_string}' method set to 'DELETE' Delete Attribute From Temporal Entity: and Query Parameter: entityId set to 'invalidId' and Query Parameter: attributeId set to 'speed' and Query Parameter: content_type set to 'application/ json' and Query Parameter: datasetId set to '\$ {EMPTY}' and Query Parameter: deleteAll set to 'false' and Query Parameter: context set to '\$ {EMPTY}' } then { the SUT sends a valid Response for the operations: Delete Attribute From Temporal Entity with Response Body containing Delete Attribute From Temporal Entity with Response Body containing	SUT -> Client
	unknown/invalid Entity/Attribute Id Test events when { the SUT receives a Request from the client containing: URL set to / ngsi-id/v1/temporal/entities/(entityld/)attrs/(attributeld)?\$ (params_as_string)' method set to 'DeLETE' Delete Attribute From Temporal Entity: and Query Parameter: entityld set to '\$ (EMPTY)' and Query Parameter: content_type set to 'application/ ison' and Query Parameter: datasetId set to 'false' and Query Parameter: context set to '\$ (EMPTY)' and Query Parameter: context set to '\$ (EMPTY)' and Query Parameter: context set to 'false' and Query Parameter: context set to '\$ (EMPTY)' } then { the SUT sends a valid Response for the operations: Delete Attribute From Temporal Entity with Response Body containing the type 'https:// url.etsi.org/ngsi-Id/errors/ BadRequestData' and Delete Attribute From Temporal Entity with Response body containing title' element) TP/NGSI-LD/CI/Prov/TEA/015_02_02 O15_02_02 Delete an attribute to a temporal representation of an entity with a id v1.3.1 Check that an error is raised if one deletes an attribute to temporal entity with a id v1.3.1 Check that an error is raised if one deletes an attribute to temporal entity with a id v1.3.1 Check that an error is raised if one deletes an attribute to temporal entity with a id v1.3.1 Check that an error is raised if one deletes an attribute to temporal entity with a id v1.3.1 Check that an error is raised if one deletes an attribute to temporal entity with a id v1.3.1 Check that an error is raised if one deletes an attribute to temporal entity with a id v1.3.1 Check that an error is raised if one deletes an attribute to temporal entity with a id v1.3.1 Check that an error is raised if one deletes an attribute to temporal entity with a id v1.3.1 Check that an error is raised if one deletes an attributed to temporal entity with a id v1.3.1 Check that an error is raised if one deletes an attributed to temporal entity with a id v1.3.1 Check that an error is raised if one deletes an attributed to temporal entity with a id

	'title' element	
	}	
TP Permutation id	TP/NGSI-LD/CI/Prov/TEA/015_02_03	
Name	015_02_03 Delete an attribute to a temporal representation of an entity with an invalid	
	attribute id	
Parent Release	v1.3.1	
Description	Check that an error is raised if one deletes an attribute to temporal entity with unknown/invalid Entity/Attribute Id	an
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{entityId}/attrs/{attributeId}?\$ {params_as_string}' method set to 'DELETE' Delete Attribute From Temporal Entity: and Query Parameter: entityId set to '\$ {valid_temporal_entity_id}' and Query Parameter: attributeId set to 'invalid(Name' and Query Parameter: content_type set to 'application/ json' and Query Parameter: datasetId set to '\$ {EMPTY}' and Query Parameter: deleteAll set to 'false' and Query Parameter: context set to '\$ {EMPTY}'	SUT <- Client
	then { the SUT sends a valid Response for the operations: Delete Attribute From Temporal Entity with Response Status Code set to 400 and Delete Attribute From Temporal Entity with Response Body containing the type 'https:// uri.etsi.org/ngsi-ld/errors/ BadRequestData' and Delete Attribute From Temporal Entity with Response body containing 'title' element }	SUT -> Client

4.4.3.5.2 DeleteAttributeInstance

TP Id	TP/NGSI-LD/CI/Prov/TEA/017_01	
Test objective	Check that one can delete an attribute instance in temporal representation of a	n entity
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.15	•
Config Id	CF_01	
PICS Selection	PICS_AttInsTemEvoEnt	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/TEA/017_01_01	
Name	017_01_01 Delete an attribute instance in temporal representation of an entity	
Parent Release	v1.3.1	
Description	Check that one can delete an attribute instance in temporal representation of a	n entity
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to /	
	ngsi-ld/v1/temporal/entities/{temporal_entity_id}/attrs/speed/	
	{instanceId}'	
	method set to 'DELETE'	
	Delete Attribute Instance from Temporal Entity: and	
	Query Parameter: temporal_entity_id set to '\$	
	{temporal_entity_representation_id}' and	
	Query Parameter: attributeld set to 'speed' and	
	Query Parameter: instanceld set to '\$	
	{instanceId}' and	

Query Parameter: content_type set to 'application/ json' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld'	
then { the SUT sends a valid Response for the operations: Create Or Update Temporal Representation Of Entity Selecting Content Type with Response Status Code set to 201 and Delete Attribute Instance From Temporal Entity with Response Status Code set to 204 and Retrieve Temporal Representation Of Entity with Check Updated Entity and Query Parameter: 'updated_resource' set to 'temporal_entity_expectation_payload' and Query Parameter: 'response_body' set to 'response1.json()' and Query Parameter: 'ignored_keys' set to 'ignored_attributes' }	SUT -> Client

TP ld	TD/NCCLLD/CI/Dray/TEA/047, 02	
	TP/NGSI-LD/CI/Prov/TEA/017_03	-f
Test objective	Check that one cannot delete an attribute instance in temporal representation the EntityId/AttributeId/InstanceId is not right	or an entity if
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.15	
Config Id	CF 01	
PICS Selection	PICS_AttInsTemEvoEnt	
Initial conditions		noral Entity
initial conditions	with { the SUT being in the "initial state" and the SUT containing an initial Tem	iporai Enilly
	\${entity} with an id set to \${temporal_entity_representation_id} } Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/TEA/017_03_01	
Name	017_03_01 Delete an attribute instance in temporal representation of an entity	, if the entity
Name	with given id is not found	y ii trie eritity
Parent Release	v1.3.1	
Description	Check that one cannot delete an attribute instance in temporal representation	of an antity if
Description	the EntityId/AttributeId/InstanceId is not found	or an entity if
Expected behaviour	Test events	Direction
Expected bellavious	when {	SUT <- Client
	the SUT receives a Request from the client containing:	301 <- Client
	URL set to '/	
	ngsi-ld/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/	
	{instanceId}'	
	method set to 'DELETE'	
	Delete Attribute Instance from Temporal Entity: and	
	Query Parameter: temporal_entity_id set to '\$	
	{unknown_temporal_entity_id}' and	
	Query Parameter: attributeld set to '\$	
	{attributeId}' and	
	Query Parameter: instanceld set to '\$	
	{instanceId}' and	
	Query Parameter: content_type set to 'application/	
	json' and	
	Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/	
	jsonld-contexts/ngsi-ld-test-suite-compound.jsonld'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operation:	
	Delete Attribute Instance From Temporal Entity with Response Status	
	Code set to	
	404	
	}	
TP Permutation id	TP/NGSI-LD/CI/Prov/TEA/017_03_02	
Name	017_03_02 Delete an attribute instance in temporal representation of an entity	y if the target
	attribute is not found	
Parent Release	v1.3.1	

Description	Check that one cannot delete an attribute instance in temporal representation the EntityId/AttributeId/InstanceId is not found	of an entity if
Expected behaviour	Test events	Direction
Expected behaviour		Direction SUT <- Client
	then { the SUT sends a valid Response for the operation: Delete Attribute Instance From Temporal Entity with Response Status Code set to 404	SUT -> Client
TD D (
TP Permutation id	TP/NGSI-LD/CI/Prov/TEA/017_03_03	if also it
Name	017_03_03 Delete an attribute instance in temporal representation of an entit attribute instance is not found	y if the target
Parent Release	v1.3.1	
Description	Check that one cannot delete an attribute instance in temporal representation the Entityld/AttributeId/InstanceId is not found	of an entity if
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'DELETE' Delete Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attributeId set to '\$ {attributeId}' and Query Parameter: instanceId set to '\$ {instanceId}' and Query Parameter: content_type set to 'application/ json' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/ jsonId-contexts/ngsi-Id-test-suite-compound.jsonId' }	SUT <- Client
	then { the SUT sends a valid Response for the operation: Delete Attribute Instance From Temporal Entity with Response Status Code set to 404 }	SUT -> Client

TP Id	TP/NGSI-LD/CI/Prov/TEA/017_02
Test objective	Check that one cannot delete an attribute instance in temporal representation of an entity if
	the EntityId/AttributeId/InstanceId is not right
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.15
Config Id	CF_01
PICS Selection	PICS_AttInsTemEvoEnt

Initial conditions	with { the SUT being in the "initial state" and the SUT containing an initial Ten \${entity} with an id set to \${temporal_entity_representation_id} and an attribut to \${atrld} }	
	Permutations	
TP Permutation id Name	TP/NGSI-LD/CI/Prov/TEA/017_02_01 017_02_01 Delete an attribute instance in temporal representation of an entit is not valid	y if the entity id
Parent Release	v1.3.1	
Description	Check that one cannot delete an attribute instance in temporal representation	of an entity if
-	the EntityId/AttributeId/InstanceId is not right	•
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/	SUT <- Client
	ngsi-ld/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'DELETE'	
	Delete Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to 'invalidId' and Query Parameter: attributeId set to '\$ {attributeId}' and	
	Query Parameter: instanceld set to '\$ {instanceld}' and	
	Query Parameter: content_type set to 'application/ json' and Query Parameter: context set to 'https://	
	forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonld-contexts/ngsi-ld-test-suite-compound.jsonld'	
	then { the SUT sends a valid Response for the operation: Delete Attribute Instance From Temporal Entity with Response Status	SUT -> Client
	Code set to 400 }	
TP Permutation id	TP/NGSI-LD/CI/Prov/TEA/017_02_02	
Name	017_02_02 Delete an attribute instance in temporal representation of an entit is not present	y if the entity id
Parent Release	v1.3.1	
Description	Check that one cannot delete an attribute instance in temporal representation the EntityId/AttributeId/InstanceId is not right	
Expected behaviour	Test events when {	Direction SUT <- Client
	the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'DELETE' Delete Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to '\$ {EMPTY}' and Query Parameter: attributeId set to '\$ {attributeId}' and Query Parameter: instanceId set to '\$ {instanceId}' and Query Parameter: content_type set to 'application/ json' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-ld-test-suite/-/raw/develop/resources/ jsonId-contexts/ngsi-Id-test-suite-compound.jsonId' } then { the SUT sends a valid Response for the operation: Delete Attribute Instance From Temporal Entity with Response Status Code set to 400	SUT -> Client
	}	
TP Permutation id	TP/NGSI-LD/CI/Prov/TEA/017_02_03	

Name	017_02_03 Delete an attribute instance in temporal representation of an entity instance id is not valid	y if the
Parent Release	V1.3.1	
Description	Check that one cannot delete an attribute instance in temporal representation the Entityld/Attributeld/InstanceId is not right	of an entity if
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'DELETE' Delete Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attributeId set to '\$	SUT <- Client
	{attributeId}' and Query Parameter: instanceId set to '\$ {instanceId}' and Query Parameter: content_type set to 'application/ json' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/ jsonId-contexts/ngsi-Id-test-suite-compound.jsonId' }	
	then { the SUT sends a valid Response for the operation: Delete Attribute Instance From Temporal Entity with Response Status Code set to 400 }	SUT -> Client
TP Permutation id	TP/NGSI-LD/CI/Prov/TEA/017_02_04	
Name	017_02_04 Delete an attribute instance in temporal representation of an entity	y if the attribute
	name is not a valid name	
Parent Release	V1.3.1	
Description	Check that one cannot delete an attribute instance in temporal representation the EntityId/AttributeId/InstanceId is not right	or an entity if
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'DELETE' Delete Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attributeId set to '\$ {attributeId}' and Query Parameter: instanceId set to '\$ {instanceId}' and Query Parameter: content_type set to 'application/ json' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/ jsonId-contexts/ngsi-Id-test-suite-compound.jsonId' }	SUT <- Client
	then { the SUT sends a valid Response for the operation: Delete Attribute Instance From Temporal Entity with Response Status Code set to 400	SUT -> Client
TP Permutation id	the SUT sends a valid Response for the operation: Delete Attribute Instance From Temporal Entity with Response Status Code set to 400 }	SUT -> Client
TP Permutation id	the SUT sends a valid Response for the operation: Delete Attribute Instance From Temporal Entity with Response Status Code set to 400 } TP/NGSI-LD/CI/Prov/TEA/017_02_05	
	the SUT sends a valid Response for the operation: Delete Attribute Instance From Temporal Entity with Response Status Code set to 400 }	

Description	Check that one cannot delete an attribute instance in temporal representation	of an entity if
Expected behaviour	the EntityId/AttributeId/InstanceId is not right Test events	Direction
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'DELETE' Delete Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attributeId set to '\$ {attributeId}' and Query Parameter: instanceId set to '\$ {instanceId}' and Query Parameter: content_type set to 'application/ json' and Query Parameter: context set to 'https:// forge.etsi.org/rep/cim/ngsi-Id-test-suite/-/raw/develop/resources/ jsonId-contexts/ngsi-Id-test-suite-compound.jsonId' }	SUT <- Client
	then { the SUT sends a valid Response for the operation: Delete Attribute Instance From Temporal Entity with Response Status Code set to 400 }	SUT -> Client

4.4.3.5.3 AddAttributes

TP ld	TP/NGSI-LD/CI/Prov/TEA/014_03	
Test objective	Check that an error is raised if one adds an attribute to a non-existent entity	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.12	
Config Id	CF_01	
PICS Selection	PICS_TemEvoAttEntIns	
Initial conditions	with { the SUT containing an initial state }	
TD Dames della milet	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/TEA/014_03_01	
Name	014_03_01 Add Attribute To Temporal Entity	
Parent Release	v1.3.1	
Description	Check that an error is raised if one adds an attribute to a non-existent entity	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities/{temporal_entity_id}/attrs' method set to 'POST' Append Attribute to Temporal Entity and Query Parameter: id set to '\$ {not_found_temporal_entity_representation_id}' and Query Parameter: fragment_filename set to 'vehicle-temporal-representation-fragment.jsonld' and Query Parameter: content_type set to 'application/ Id+json' } then {	SUT -> Client
	then { the SUT sends a valid Response for the operations: Create Or Update Temporal Representation Of Entity Selecting Content Type with Response Status Code set to 201 and Append Attribute To Temporal Entity with Response Status Code set to 404 }	ISUT -> Client

TP Id	TP/NGSI-LD/CI/Prov/TEA/014_01	
Test objective	Check that one can add a simple temporal attribute to a temporal representation	on of an entity
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.12	
Config Id	CF_01	
PICS Selection	PICS_TemEvoAttEntIns	
Initial conditions	with { the SUT containing an initial state }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/TEA/014_01_01	
Name	014_01_01 Add an attribute to a temporal entity with simple temporal properties	es
Parent Release	v1.3.1	
Description	Check that one can add a simple temporal attribute to a temporal representation	on of an entity
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/	SUT <- Client
	ngsi-ld/v1/temporal/entities/{temporal_entity_id}/attrs' method set to 'POST'	
	Append Attribute to Temporal Entity and Query Parameter: id set to '\$	
	{temporal_entity_representation_id}' and	
	Query Parameter: fragment_filename set to 'vehicle-	
	temporal-representation-fragment.jsonId' and	
	Query Parameter: content_type set to 'application/	
	ld+json'	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations: Create Or Update Temporal Representation Of Entity Selecting Content Type with Response Status Code set to 201 and	301 -> Client
	Append Attribute To Temporal Entity with Response Status Code set to 204 and	
	Retrieve Temporal Representation Of Entity with Check Updated Entity and	
	Query Parameter: 'updated_resource' set to 'temporal_entity_expectation_payload' and	
	Query Parameter: 'response_body' set to 'response1.json()' and Query Parameter: 'ignored_keys' set to 'ignored_attributes'	
	}	

TP ld	TD/NCCLLD/CI/Droy/TEA/044_04	
	TP/NGSI-LD/CI/Prov/TEA/014_04	
Test objective	Check that an error is raised if one adds a temporal entity attribute with empty	/invalid
	content	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.12	
Config Id	CF_01	
PICS Selection	PICS_TemEvoAttEntIns	
Initial conditions	with { the SUT being in the "initial state" and the SUT containing an initial Tem	poral Entity
	\${entity} with an id set to \${temporal_entity_representation_id} }	
	Permutations	
TP Permutation id	TP/NGSI-LD/CI/Prov/TEA/014_04_01	
Name	014_04_01 Add an attribute to a temporal representation of an entity with invalid content	
Parent Release	v1.3.1	
Description	Check that an error is raised if one adds a temporal entity attribute with empty	/invalid
	content	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/temporal/entities/{temporal_entity_id}/attrs'	
	method set to 'POST'	
	Append Attribute to Temporal Entity and	
	Query Parameter: id set to '\$	
	{temporal_entity_representation_id}' and	
	Query Parameter: fragment_filename set to 'vehicle-	
	temporal-representation-invalid-json-fragment.jsonId' and	
	Query Parameter: content_type set to 'application/	

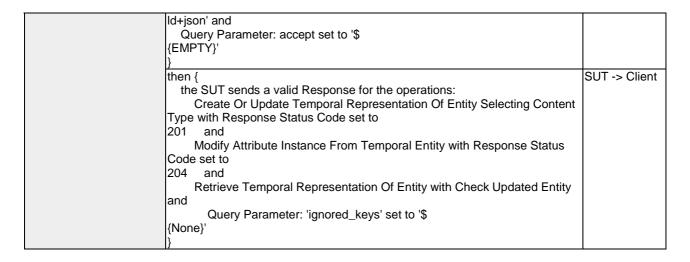
	ld+json'	
	then {	SUT -> Client
	the SUT sends a valid Response for the operations:	
	Append Attribute To Temporal Entity with Response Status Code set to	
	400 and Append Attribute To Temporal Entity with Response Body containing	
	the type 'https://	
	uri.etsi.org/ngsi-ld/errors/	
	InvalidRequest' and	
	Append Attribute To Temporal Entity with Response body containing	
	'title' element	
	}	
TP Permutation id	TP/NGSI-LD/CI/Prov/TEA/014_04_02	4
Name	014_04_02 Add an attribute to a temporal representation of an entity with emp	ty content
Parent Release Description	v1.3.1 Check that an error is raised if one adds a temporal entity attribute with empty.	/invalid
Description	content	iiivaliu
Expected behaviour	Test events	Direction
•	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/temporal/entities/{temporal_entity_id}/attrs'	
	method set to 'POST' Append Attribute to Temporal Entity and	
	Query Parameter: id set to '\$	
	{temporal_entity_representation_id}' and	
	Query Parameter: fragment_filename set to 'vehicle-	
	temporal-representation-empty-json-fragment.jsonld' and	
	Query Parameter: content_type set to 'application/	
	ld+json'	
	than (SUT -> Client
	then { the SUT sends a valid Response for the operations:	SUT -> Client
	Append Attribute To Temporal Entity with Response Status Code set to	
	400 and	
	Append Attribute To Temporal Entity with Response Body containing	
	the type 'https://	
	uri.etsi.org/ngsi-Id/errors/	
	InvalidRequest' and	
	Append Attribute To Temporal Entity with Response body containing	
	Append Attribute To Temporal Entity with Response body containing 'title' element }	

I		1	
TP ld	TP/NGSI-LD/CI/Prov/TEA/014_02		
Test objective	Check that an error is raised if one adds an attribute to a temporal entity with in	nvalid content	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.12		
Config Id	CF_01		
PICS Selection	PICS_TemEvoAttEntIns		
Initial conditions	with { the SUT containing an initial state }		
	Permutations		
TP Permutation id	TP/NGSI-LD/CI/Prov/TEA/014_02_01		
Name	014_02_01 Add an attribute to a temporal representation of an entity with an e	mpty entity id	
Parent Release	v1.3.1		
Description	Check that an error is raised if one adds a temporal entity attribute with a non-		
	existing/invalid EntityId		
Expected behaviour	Test events	Direction	
	when {	SUT <- Client	
	the SUT receives a Request from the client containing:		
	URL set to '/		
	ngsi-ld/v1/temporal/entities/{temporal_entity_id}/attrs'		
	method set to 'POST'		
	Append Attribute to Temporal Entity and		
	Query Parameter: id set to '\$		
	{EMPTY}' and		
	Query Parameter: fragment_filename set to 'vehicle-		

	temporal-representation-fragment.jsonld' and Query Parameter: content_type set to 'application/ Id+json' } then { the SUT sends a valid Response for the operations: Create Or Update Temporal Representation Of Entity Selecting Content The with Response Status Code set to	
	Type with Response Status Code set to 201 and Append Attribute To Temporal Entity with Response Status Code set to 400 }	
TP Permutation id	TP/NGSI-LD/CI/Prov/TEA/014_02_02	
Name	014_02_02 Add an attribute to a temporal representation of an entity with an in	nvalid entity id
Parent Release	v1.3.1	
Description	Check that an error is raised if one adds a temporal entity attribute with a non- existing/invalid EntityId	
Expected behaviour	Test events	Direction
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_id}/attrs' method set to 'POST' Append Attribute to Temporal Entity and Query Parameter: id set to 'thisIsAninvalidId' and Query Parameter: fragment_filename set to 'vehicle- temporal-representation-fragment.jsonId' and Query Parameter: content_type set to 'application/ Id+json'	
	then { the SUT sends a valid Response for the operations: Create Or Update Temporal Representation Of Entity Selecting Content Type with Response Status Code set to 201 and Append Attribute To Temporal Entity with Response Status Code set to 400 }	SUT -> Client

4.4.3.5.4 PartialUpdateAttributeInstance

TP ld	TP/NGSI-LD/CI/Prov/TEA/016_01		
Test objective	Check that one can modify an attribute instance in temporal representation of an entity		
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.14		
Config Id	CF_01		
PICS Selection	PICS_AttInsTemEvoEnt		
Initial conditions	with { the SUT containing an initial state }		
	Permutations		
TP Permutation id	TP/NGSI-LD/CI/Prov/TEA/016_01_01		
Name	016_01_01 Modify attribute instance in temporal representation of an entity		
Parent Release	v1.3.1		
Description	Check that one can partially update an attribute instance of a temporal representation	entation of an	
	entity		
	Test events Direction		
Expected behaviour	Test events	Direction	
Expected behaviour	Test events when {	Direction SUT <- Client	
Expected behaviour			
Expected behaviour	when {		
Expected behaviour	when { the SUT receives a Request from the client containing:		
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/		
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities'		
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'POST'		
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'POST' Create or Update Temporal Representation of Entity Selecting Content		
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'POST' Create or Update Temporal Representation of Entity Selecting Content Type: and		
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'POST' Create or Update Temporal Representation of Entity Selecting Content Type: and Query Parameter: temporal_entity_representation_id set to '\$		
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities' method set to 'POST' Create or Update Temporal Representation of Entity Selecting Content Type: and Query Parameter: temporal_entity_representation_id set to '\$ {temporal_entity_representation_id}' and		



TP ld	TP/NGSI-LD/CI/Prov/TEA/016_03		
Test objective	Check that one cannot partially modify attribute instance in temporal represen	tation of an	
rest objective	entity if the EntityId/AttributeId/InstanceId is not found	tation of an	
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.14		
Config Id	CF_01		
PICS Selection	PICS_AttInsTemEvoEnt		
Initial conditions	with { the SUT being in the "initial state" and the SUT containing an initial Temporal Entity		
	\$\text{entity} with an id set to \$\text{temporal_entity_representation_id} \}		
	Permutations		
TP Permutation id	TP/NGSI-LD/CI/Prov/TEA/016_03_01		
Name	016_03_01 Modify attribute instance in temporal representation of an entity if	the entity with	
	given id is not found		
Parent Release	v1.3.1		
Description	Check that one cannot partially modify attribute instance in temporal represen	tation of an	
•	entity if the EntityId/AttributeId/InstanceId is not found		
Expected behaviour	Test events	Direction	
-	when {	SUT <- Client	
	the SUT receives a Request from the client containing:		
	URL set to '/		
	ngsi-ld/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/		
	{instanceId}'		
	method set to 'PATCH'		
	Modify Attribute Instance from Temporal Entity: and		
	Query Parameter: temporal_entity_id set to '\$		
	{unknown_temporal_entity_id}' and		
	Query Parameter: attributeld set to '\$		
	{attributeId}' and		
	Query Parameter: instanceld set to '\$		
	{instanceId}' and Query Parameter: fragment_filename set to 'vehicle-		
	temporal-modify-attribute-instance-fragment.jsonld' and		
	Query Parameter: content_type set to 'application/		
	ison' and		
	Query Parameter: context set to '\$		
	{EMPTY}'		
	\{\begin{align*}		
	then {	SUT -> Client	
	the SUT sends a valid Response for the operation:		
	Modify Attribute Instance From Temporal Entity with Response Status		
	Code set to		
	404		
	}		
TP Permutation id	TP/NGSI-LD/CI/Prov/TEA/016_03_02		
Name	016_03_02 Modify attribute instance in temporal representation of an entity if	the target	
	attribute is not found		
Parent Release	v1.3.1		
Description	Check that one cannot partially modify attribute instance in temporal represen	tation of an	
	entity if the EntityId/AttributeId/InstanceId is not found		

Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}'	
	method set to 'PATCH'	
	Modify Attribute Instance from Temporal Entity: and	
	Query Parameter: temporal_entity_id set to '\$	
	{temporal_entity_representation_id}' and	
	Query Parameter: attributeld set to '\$	
	{attributeId}' and Query Parameter: instanceId set to '\$	
	{instanceId}' and	
	Query Parameter: fragment_filename set to 'vehicle-	
	temporal-modify-attribute-instance-fragment.jsonId' and	
	Query Parameter: content_type set to 'application/	
	json' and	
	Query Parameter: context set to '\$	
	{EMPTY}'	
	then {	SUT -> Client
	the SUT sends a valid Response for the operation:	Jon / Onorit
	Modify Attribute Instance From Temporal Entity with Response Status	
	Code set to	
	404	
TD Dormutation id	TD/N/CCLL D/CL/Droy/TEA/046, 02, 02	
TP Permutation id Name	TP/NGSI-LD/CI/Prov/TEA/016_03_03 016_03_03 Modify attribute instance in temporal representation of an entity if	the target
Name	attribute instance is not found	the target
Danaud Dalaasa		
Parent Release	v1.3.1	
Description	v1.3.1 Check that one cannot partially modify attribute instance in temporal representations.	ntation of an
Description	Check that one cannot partially modify attribute instance in temporal representative if the EntityId/AttributeId/InstanceId is not found	
	Check that one cannot partially modify attribute instance in temporal representity if the EntityId/AttributeId/InstanceId is not found Test events	Direction
Description	Check that one cannot partially modify attribute instance in temporal representity if the EntityId/AttributeId/InstanceId is not found Test events when {	
Description	Check that one cannot partially modify attribute instance in temporal representity if the EntityId/AttributeId/InstanceId is not found Test events when { the SUT receives a Request from the client containing:	Direction
Description	Check that one cannot partially modify attribute instance in temporal representative if the EntityId/AttributeId/InstanceId is not found Test events when { the SUT receives a Request from the client containing: URL set to '/	Direction
Description	Check that one cannot partially modify attribute instance in temporal representity if the EntityId/AttributeId/InstanceId is not found Test events when { the SUT receives a Request from the client containing:	Direction
Description	Check that one cannot partially modify attribute instance in temporal representative if the EntityId/AttributeId/InstanceId is not found Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'PATCH'	Direction
Description	Check that one cannot partially modify attribute instance in temporal representative if the EntityId/AttributeId/InstanceId is not found Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'PATCH' Modify Attribute Instance from Temporal Entity: and	Direction
Description	Check that one cannot partially modify attribute instance in temporal representative if the EntityId/AttributeId/InstanceId is not found Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'PATCH' Modify Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to '\$	Direction
Description	Check that one cannot partially modify attribute instance in temporal represent entity if the EntityId/AttributeId/InstanceId is not found Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'PATCH' Modify Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to '\$ {temporal_entity_representation_id}' and	Direction
Description	Check that one cannot partially modify attribute instance in temporal representation if the EntityId/AttributeId/InstanceId is not found Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'PATCH' Modify Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attributeId set to '\$	Direction
Description	Check that one cannot partially modify attribute instance in temporal representation if the EntityId/AttributeId/InstanceId is not found Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'PATCH' Modify Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to '\$ {temporal_entity_representation_id}' and	Direction
Description	Check that one cannot partially modify attribute instance in temporal represent entity if the EntityId/AttributeId/InstanceId is not found Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'PATCH' Modify Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attributeId set to '\$ {attributeId}' and Query Parameter: instanceId set to '\$ {instanceId}' and	Direction
Description	Check that one cannot partially modify attribute instance in temporal represent entity if the EntityId/AttributeId/InstanceId is not found Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'PATCH' Modify Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attributeId set to '\$ {attributeId}' and Query Parameter: instanceId set to '\$ {instanceId}' and Query Parameter: fragment_filename set to 'vehicle-	Direction
Description	Check that one cannot partially modify attribute instance in temporal represent entity if the EntityId/AttributeId/InstanceId is not found Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'PATCH' Modify Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attributeId set to '\$ {attributeId}' and Query Parameter: instanceId set to '\$ {instanceId}' and Query Parameter: fragment_filename set to 'vehicle-temporal-modify-attribute-instance-fragment.jsonId' and	Direction
Description	Check that one cannot partially modify attribute instance in temporal representity if the EntityId/AttributeId/InstanceId is not found Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'PATCH' Modify Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attributeId set to '\$ {attributeId}' and Query Parameter: instanceId set to '\$ {instanceId}' and Query Parameter: fragment_filename set to 'vehicle-temporal-modify-attribute-instance-fragment.jsonId' and Query Parameter: content_type set to 'application/	Direction
Description	Check that one cannot partially modify attribute instance in temporal representative if the EntityId/AttributeId/InstanceId is not found Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'PATCH' Modify Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attributeId set to '\$ {attributeId}' and Query Parameter: instanceId set to '\$ {instanceId}' and Query Parameter: fragment_filename set to 'vehicle-temporal-modify-attribute-instance-fragment.jsonId' and Query Parameter: content_type set to 'application/ json' and	Direction
Description	Check that one cannot partially modify attribute instance in temporal representity if the EntityId/AttributeId/InstanceId is not found Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'PATCH' Modify Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attributeId set to '\$ {attributeId}' and Query Parameter: instanceId set to '\$ {instanceId}' and Query Parameter: fragment_filename set to 'vehicle-temporal-modify-attribute-instance-fragment.jsonId' and Query Parameter: content_type set to 'application/	Direction
Description	Check that one cannot partially modify attribute instance in temporal representativy if the EntityId/AttributeId/InstanceId is not found Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'PATCH' Modify Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attributeId set to '\$ {attributeId}' and Query Parameter: instanceId set to '\$ {instanceId}' and Query Parameter: fragment_filename set to 'vehicle-temporal-modify-attribute-instance-fragment.jsonId' and Query Parameter: content_type set to 'application/ json' and Query Parameter: context set to '\$	Direction SUT <- Client
Description	Check that one cannot partially modify attribute instance in temporal representity if the EntityId/AttributeId/InstanceId is not found Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'PATCH' Modify Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attributeId set to '\$ {attributeId}' and Query Parameter: instanceId set to '\$ {instanceId}' and Query Parameter: fragment_filename set to 'vehicle-temporal-modify-attribute-instance-fragment.jsonId' and Query Parameter: content_type set to 'application/ json' and Query Parameter: context set to '\$ {EMPTY}' } then {	Direction
Description	Check that one cannot partially modify attribute instance in temporal representity if the EntityId/AttributeId/InstanceId is not found Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'PATCH' Modify Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attributeId set to '\$ {attributeId}' and Query Parameter: instanceId set to '\$ {instanceId}' and Query Parameter: fragment_filename set to 'vehicle-temporal-modify-attribute-instance-fragment.jsonId' and Query Parameter: content_type set to 'application/ json' and Query Parameter: context set to '\$ {EMPTY}' } then { the SUT sends a valid Response for the operation:	Direction SUT <- Client
Description	Check that one cannot partially modify attribute instance in temporal representity if the EntityId/AttributeId/InstanceId is not found Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'PATCH' Modify Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attributeId set to '\$ {attributeId}' and Query Parameter: instanceId set to '\$ {instanceId}' and Query Parameter: fragment_filename set to 'vehicle-temporal-modify-attribute-instance-fragment.jsonId' and Query Parameter: content_type set to 'application/ json' and Query Parameter: context set to '\$ {EMPTY}' } then { the SUT sends a valid Response for the operation: Modify Attribute Instance From Temporal Entity with Response Status	Direction SUT <- Client
Description	Check that one cannot partially modify attribute instance in temporal representity if the EntityId/AttributeId/InstanceId is not found Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'PATCH' Modify Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attributeId set to '\$ {attributeId}' and Query Parameter: instanceId set to '\$ {instanceId}' and Query Parameter: fragment_filename set to 'vehicle-temporal-modify-attribute-instance-fragment.jsonId' and Query Parameter: content_type set to 'application/ json' and Query Parameter: context set to '\$ {EMPTY}' } then { the SUT sends a valid Response for the operation: Modify Attribute Instance From Temporal Entity with Response Status Code set to	Direction SUT <- Client
Description	Check that one cannot partially modify attribute instance in temporal representity if the EntityId/AttributeId/InstanceId is not found Test events when { the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'PATCH' Modify Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attributeId set to '\$ {attributeId}' and Query Parameter: instanceId set to '\$ {instanceId}' and Query Parameter: fragment_filename set to 'vehicle-temporal-modify-attribute-instance-fragment.jsonId' and Query Parameter: content_type set to 'application/ json' and Query Parameter: context set to '\$ {EMPTY}' } then { the SUT sends a valid Response for the operation: Modify Attribute Instance From Temporal Entity with Response Status	Direction SUT <- Client

TP Id	TP/NGSI-LD/CI/Prov/TEA/016_02
Test objective	Check that one cannot modify an attribute instance in temporal representation of an entity if
	the EntityId/AttributeId/InstanceId is not right
Reference	ETSI GS CIM 009 V1.5.1 [1], clause 5.6.14
Config Id	CF_01
PICS Selection	PICS_AttInsTemEvoEnt

Initial conditions	with { the SUT being in the "initial state" and the SUT containing an initial Temporal Entity \${entity} with an id set to \${temporal_entity_representation_id} }		
	Permutations Table 19 19 19 19 19 19 19 19 19 19 19 19 19		
TP Permutation id	TP/NGSI-LD/CI/Prov/TEA/016_02_01		
Name	016_02_01 Modify attribute instance in temporal representation of an entity if not valid	the entity id is	
Parent Release	v1.3.1		
Description	Check that one cannot partially modify attribute instance in temporal represent entity if the EntityId/AttributeId/InstanceId is not right	tation of an	
Expected behaviour	Test events	Direction	
	the SUT receives a Request from the client containing: URL set to '/ ngsi-Id/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'PATCH' Modify Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to 'invalidId' and Query Parameter: attributeId set to '\$ {attributeId}' and Query Parameter: instanceId set to '\$ {instanceId}' and Query Parameter: fragment_filename set to 'vehicle-temporal-modify-attribute-instance-fragment.jsonId' and Query Parameter: content_type set to 'application/ json' and Query Parameter: context set to '\$ {EMPTY}'		
TP Permutation id	then { the SUT sends a valid Response for the operation: Modify Attribute Instance From Temporal Entity with Response Status Code set to 400 } TP/NGSI-LD/CI/Prov/TEA/016_02_02	SUT -> Client	
Name	016_02_02 Modify attribute instance in temporal representation of an entity if not present	the entity id is	
Parent Release	v1.3.1		
Description	Check that one cannot partially modify attribute instance in temporal represent entity if the EntityId/AttributeId/InstanceId is not right	tation of an	
Expected behaviour	Test events	Direction	
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'PATCH' Modify Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to '\$ {EMPTY}' and Query Parameter: attributeId set to '\$ {attributeId}' and Query Parameter: instanceId set to '\$ {instanceId}' and Query Parameter: fragment_filename set to 'vehicle-temporal-modify-attribute-instance-fragment.jsonId' and Query Parameter: content_type set to 'application/ json' and Query Parameter: context set to '\$ {EMPTY}' } then { the SUT sends a valid Response for the operation: Modify Attribute Instance From Temporal Entity with Response Status	SUT> Client	

TP Permutation id	TP/NGSI-LD/CI/Prov/TEA/016_02_03	
Name	016_02_03 Modify attribute instance in temporal representation of an entity if	the instance id
	is not valid	
Parent Release	v1.3.1	
Description	Check that one cannot partially modify attribute instance in temporal represen	ntation of an
	entity if the EntityId/AttributeId/InstanceId is not right	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/	
	{instanceId}'	
	method set to 'PATCH'	
	Modify Attribute Instance from Temporal Entity: and	
	Query Parameter: temporal_entity_id set to '\$	
	{temporal_entity_representation_id}' and Query Parameter: attributeId set to '\$	
	{attributeId}' and	
	Query Parameter: instanceld set to '\$	
	{instanceId}' and	
	Query Parameter: fragment_filename set to 'vehicle-	
	temporal-modify-attribute-instance-fragment.jsonld' and	
	Query Parameter: content_type set to 'application/	
	ison' and	
	Query Parameter: context set to '\$	
	{EMPTY}'	
	}	
	then {	SUT -> Client
	the SUT sends a valid Response for the operation:	OOT > OHOII
	Modify Attribute Instance From Temporal Entity with Response Status	
	Code set to	
	400	
	}	
TP Permutation id	TP/NGSI-LD/CI/Prov/TEA/016_02_04	•
Name	016_02_04 Modify attribute instance in temporal representation of an entity if	the instance id
	is not present	
Parent Release	v1.3.1	
Description	Check that one cannot partially modify attribute instance in temporal represen	ntation of an
	entity if the EntityId/AttributeId/InstanceId is not right	
Expected behaviour	Test events	Direction
	when {	SUT <- Client
	the SUT receives a Request from the client containing:	
	URL set to '/	
	ngsi-ld/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/	
	ngsi-ld/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}'	
	ngsi-ld/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'PATCH'	
	ngsi-ld/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'PATCH' Modify Attribute Instance from Temporal Entity: and	
	ngsi-ld/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'PATCH' Modify Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to '\$	
	ngsi-ld/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'PATCH' Modify Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to '\$ {temporal_entity_representation_id}' and	
	ngsi-ld/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'PATCH' Modify Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attributeId set to '\$	
	ngsi-ld/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'PATCH' Modify Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attributeId set to '\$ {attributeId}' and	
	ngsi-ld/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'PATCH' Modify Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attributeId set to '\$ {attributeId}' and Query Parameter: instanceId set to '\$	
	ngsi-ld/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'PATCH' Modify Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attributeId set to '\$ {attributeId}' and Query Parameter: instanceId set to '\$ {instanceId}' and	
	ngsi-ld/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'PATCH' Modify Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attributeId set to '\$ {attributeId}' and Query Parameter: instanceId set to '\$ {instanceId}' and Query Parameter: fragment_filename set to 'vehicle-	
	ngsi-ld/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'PATCH' Modify Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attributeId set to '\$ {attributeId}' and Query Parameter: instanceId set to '\$ {instanceId}' and Query Parameter: fragment_filename set to 'vehicle-temporal-modify-attribute-instance-fragment.jsonId' and	
	ngsi-ld/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'PATCH' Modify Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attributeId set to '\$ {attributeId}' and Query Parameter: instanceId set to '\$ {instanceId}' and Query Parameter: fragment_filename set to 'vehicle-temporal-modify-attribute-instance-fragment.jsonId' and Query Parameter: content_type set to 'application/	
	ngsi-ld/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'PATCH' Modify Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attributeId set to '\$ {attributeId}' and Query Parameter: instanceId set to '\$ {instanceId}' and Query Parameter: fragment_filename set to 'vehicle-temporal-modify-attribute-instance-fragment.jsonId' and Query Parameter: content_type set to 'application/json' and	
	ngsi-ld/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'PATCH' Modify Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attributeId set to '\$ {attributeId}' and Query Parameter: instanceId set to '\$ {instanceId}' and Query Parameter: fragment_filename set to 'vehicle-temporal-modify-attribute-instance-fragment.jsonId' and Query Parameter: content_type set to 'application/ json' and Query Parameter: context set to '\$	
	ngsi-ld/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'PATCH' Modify Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attributeId set to '\$ {attributeId}' and Query Parameter: instanceId set to '\$ {instanceId}' and Query Parameter: fragment_filename set to 'vehicle-temporal-modify-attribute-instance-fragment.jsonId' and Query Parameter: content_type set to 'application/json' and	
	ngsi-ld/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'PATCH' Modify Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attributeId set to '\$ {attributeId}' and Query Parameter: instanceId set to '\$ {instanceId}' and Query Parameter: fragment_filename set to 'vehicle-temporal-modify-attribute-instance-fragment.jsonId' and Query Parameter: content_type set to 'application/json' and Query Parameter: context set to '\$ {EMPTY}' }	SLIT -> Client
	ngsi-ld/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'PATCH' Modify Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attributeId set to '\$ {attributeId}' and Query Parameter: instanceId set to '\$ {instanceId}' and Query Parameter: fragment_filename set to 'vehicle-temporal-modify-attribute-instance-fragment.jsonId' and Query Parameter: content_type set to 'application/json' and Query Parameter: context set to '\$ {EMPTY}' } then {	SUT -> Client
	ngsi-ld/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'PATCH' Modify Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attributeId set to '\$ {attributeId}' and Query Parameter: instanceId set to '\$ {instanceId}' and Query Parameter: fragment_filename set to 'vehicle-temporal-modify-attribute-instance-fragment.jsonId' and Query Parameter: content_type set to 'application/json' and Query Parameter: context set to '\$ {EMPTY}' } then { the SUT sends a valid Response for the operation:	SUT -> Client
	ngsi-ld/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'PATCH' Modify Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attributeId set to '\$ {attributeId}' and Query Parameter: instanceId set to '\$ {instanceId}' and Query Parameter: fragment_filename set to 'vehicle- temporal-modify-attribute-instance-fragment.jsonId' and Query Parameter: content_type set to 'application/ json' and Query Parameter: context set to '\$ {EMPTY}' } then { the SUT sends a valid Response for the operation: Modify Attribute Instance From Temporal Entity with Response Status	SUT -> Client
	ngsi-ld/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'PATCH' Modify Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attributeId set to '\$ {attributeId}' and Query Parameter: instanceId set to '\$ {instanceId}' and Query Parameter: fragment_filename set to 'vehicle- temporal-modify-attribute-instance-fragment.jsonId' and Query Parameter: content_type set to 'application/ json' and Query Parameter: context set to '\$ {EMPTY}' } then { the SUT sends a valid Response for the operation: Modify Attribute Instance From Temporal Entity with Response Status Code set to	SUT -> Client
	ngsi-ld/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'PATCH' Modify Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attributeId set to '\$ {attributeId}' and Query Parameter: instanceId set to '\$ {instanceId}' and Query Parameter: fragment_filename set to 'vehicle- temporal-modify-attribute-instance-fragment.jsonId' and Query Parameter: content_type set to 'application/ json' and Query Parameter: context set to '\$ {EMPTY}' } then { the SUT sends a valid Response for the operation: Modify Attribute Instance From Temporal Entity with Response Status	SUT -> Client

Name	016_02_05 Modify attribute instance in temporal representation of an entity if the attribute name is not a valid name		
Parent Release	v1.3.1		
Description Description	Check that one cannot partially modify attribute instance in temporal represer entity if the EntityId/AttributeId/InstanceId is not right	ntation of an	
Expected behaviour	Test events	Direction	
Expected behaviour	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'PATCH' Modify Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attributeId set to '\$ {attributeId}' and Query Parameter: instanceId set to '\$ {instanceId}' and Query Parameter: fragment_filename set to 'vehicle-temporal-modify-attribute-instance-fragment.jsonId' and Query Parameter: content_type set to 'application/ json' and Query Parameter: context set to '\$	SUT <- Client	
	{EMPTY}' } then { the SUT sends a valid Response for the operation: Modify Attribute Instance From Temporal Entity with Response Status Code set to 400 }	SUT -> Client	
TP Permutation id	TP/NGSI-LD/CI/Prov/TEA/016_02_06		
Name	016_02_06 Modify attribute instance in temporal representation of an entity if	the attribute	
	name is not present		
Parent Release	v1.3.1		
Description Expected behaviour	Check that one cannot partially modify attribute instance in temporal represer entity if the Entityld/Attributeld/Instanceld is not right Test events	Direction	
	when { the SUT receives a Request from the client containing: URL set to '/ ngsi-ld/v1/temporal/entities/{temporal_entity_id}/attrs/{attributeId}/ {instanceId}' method set to 'PATCH' Modify Attribute Instance from Temporal Entity: and Query Parameter: temporal_entity_id set to '\$ {temporal_entity_representation_id}' and Query Parameter: attributeId set to '\$ {attributeId}' and Query Parameter: instanceId set to '\$ {instanceId}' and Query Parameter: fragment_filename set to 'vehicle-temporal-modify-attribute-instance-fragment.jsonId' and Query Parameter: content_type set to 'application/json' and Query Parameter: context set to '\$ {EMPTY}' } then { the SUT sends a valid Response for the operation: Modify Attribute Instance From Temporal Entity with Response Status Code set to	SUT <- Client	

Annex A (informative): Bibliography

• ISO 8601:2004: "Data elements and interchange formats -- Information interchange -- Representation of dates and times".

Annex B (informative): Change History

Date	Version	Information about changes
October, 19th 2020	V0.0.1	First draft of document
February, 4th 2021	V1.0.1	Stable draft approved by ISG-CIM
March, 23 rd 2021	V1.0.2	
April 28 th 2021	V1.1.1	Last Technical Officer review for Publication pre-processing with ETSI EditHelp
October, 4th 2023	V1.2.1	Stable draft approved by ISG-CIM
November 2023	V1.3.1	First draft version of automatically generated content
December 2023	V1.4.1	Second draft version of automatically generated content
February 2024	V1.5.1	Third draft version of automatically generated content
March 2024	V1.6.1	Fourth draft version of automatically generated content. Cleanup revisions
March 2024	V1.7.1	Fifth draft version of automatically generated content, including v1.4 and v1.5 tests. Technical Officer review for Publication pre-processing with ETSI EditHelp

History

Document history		
V1.1.1	May 2021	Publication
V2.1.1	April 2024	Publication