ETSI GS CIM 013 V1.1.1 (2021-05)

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

Disclaimer

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
DGS/CIM-0013v111

Keywords
API, IoT, 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

Contents

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.

DECT™, **PLUGTESTS™**, **UMTS™** and the ETSI logo are trademarks of ETSI registered for the benefit of its Members. **3GPP™** and **LTE™** are trademarks of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners. **oneM2M™** 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 ETSI Drafting Rules (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.

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 (V1.3.1) (08-2020): "Context Information Management (CIM); NGSI-LD API".

[2] ETSI GS CIM 012 (V1.1.1) (03-2021): "Context Information Management (CIM); NGSI-LD Test Suite Structure".

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

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) (04-2021): "Context Information Management (CIM); NGSI-LD Testing Framework: Test Purposes Description Language (TPDL)".

[i.2] ETSI GS CIM 016 (V1.1.1) (04-2021): "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:

context registry: software functional element where Context Sources register the information that they can provide

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

entity: informational representative of something that is supposed to exist in the real world, physically or conceptually

valid: According to the specification references in the normative reference. For example a datatype needs to be represented as stated in the specification.

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 Context Information

4.1.1 Provision

4.1.1.1 Entities

4.1.1.1.1 Create Entity

TP Id	TP/NGSI-LD/CI/Prov/E/001_01
Test objective	Check that you can create an entity
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.1
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_5_6_1
Initial conditions	**with {** **the SUT being** in the "initial state" **}**

Expected behaviour	Test events	Direction
	when { **the SUT receives** a valid Create Entity Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entities **and** method **set to** POST **and** Header: Content-Type **set to** ${contentType} **and** body **set to** ${entity} to be created **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 201 (CREATED) Created Entity **set to** ${entity} **}**	SUT → Client

Permutation on TP Id	${contentType}	${entity}
001_01_01	Application/json	Minimal entity
001_01_02	Application/json+ld	Simple properties
001_01_03	Application/json+ld	Relationship of properties
001_01_04	Application/json+ld	With a location attribute

TP Id	TP/NGSI-LD/CI/Prov/E/001_02	
Test objective	Check that you cannot create an entity with invalid content	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.1	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_1	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a Create Entity Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entities **and** method **set to** POST **and** Header: Content-Type **set to** application/ld+json **and** body **set to ${invalid_body}**	SUT ←Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request) **and** Response Body **containing** ProblemDetails element **containing** type element **set to ${problem_type} and**	SUT → Client

	title element **containing** more information about the error **}**	

Permutation on TP Id	${invalid_body}	${problem_type}
TP/NGSI-LD/CI/Prov/E/001_02_01	invalid JSON document	https://uri.etsi.org/ngsi-ld/errors/InvalidRequest
TP/NGSI-LD/CI/Prov/E/001_02_02	empty	https://uri.etsi.org/ngsi-ld/errors/BadRequestData
TP/NGSI-LD/CI/Prov/E/001_02_03	entity with no context	https://uri.etsi.org/ngsi-ld/errors/BadRequestData

TP Id	TP/NGSI-LD/CI/Prov/E/001_03	
Test objective	Check that you cannot create an entity with an existing id	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.1	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_1	
Initial conditions	**with {** **the SUT being** in the "initial state" and **containing** an initial Entity with an id **set to** ${entityId} **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a Create Entity Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entities **and** method **set to** POST **and** Header: Content-Type **set to** application/ld+json **and** body **containing** ${entity} with the id **set to** ${entityId} **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 409 (Already Exists) **and** Response Body **containing** ProblemDetails element **containing** type element **set to** https://uri.etsi.org/ngsi-ld/errors/AlreadyExists **and** title element **containing** more information about the error **}**	SUT → Client

TP Id	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 V1.3.1 [1], clause 6.3.5	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_6_3_5	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Entity Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entities **and** Header: Content-Type **set to** application/json **and** Header: Link **set to** a @context **containing** terms used by the entity to create body **set to** entity to be created **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 201 (Created) **and** Persisted Entity contains type and attributes expanded as per the supplied @context **}**	SUT → Client

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 "application/json" and the Link header does not contain a JSON-LD @context
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 6.3.5
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_6_3_5
Initial conditions	**with {** **the SUT being** in the "initial state" **}**

Expected behaviour	Test events	Direction
	when { **the SUT receives** a valid Create Entity Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entities **and** Header: Content-Type **set to** application/json **and** body **set to** entity to be created **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 201 (Created) **and** Persisted Entity contains type and attributes expanded as per the default @context **}**	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 Content-Type header is "application/json" and the request payload body (as JSON) contains a "@context" term	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 6.3.5	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_6_3_5	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Entity Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entities **and** Header: Content-Type **set to** application/json **and** body **set to** entity **containing** a @context term **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request) **and** Response Body **containing** ProblemDetails element **containing**	SUT → Client

	type element **set to** https://uri.etsi.org/ngsi-ld/errors/BadRequestData **and** title element **containing** more information about the error **}**	

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 V1.3.1 [1], clause 6.3.5	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_6_3_5	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Entity Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entities **and** Header: Content-Type **set to** application/ld+json **and** body **set to** entity **containing** a @context term **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 201 (Created) **and** Persisted Entity contains type and attributes expanded as per the supplied @context **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/Prov/E/001_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 V1.3.1 [1], clause 6.3.5
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_6_3_5

Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Entity Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entities **and** Header: Content-Type **set to** application/ld+json **and** body **set to** entity **not containing** a @context term **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request) **and** Response Body **containing** ProblemDetails element **containing** type element **set to** https://uri.etsi.org/ngsi-ld/errors/BadRequestData **and** title element **containing** more information about the error **}**	SUT → Client

TP Id	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 incoming HTTP request
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 6.3.5
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_6_3_5
Initial conditions	**with {** **the SUT being** in the "initial state" **}**
Expected behaviour	**Test events** / **Direction**
	when { **the SUT receives** a valid Batch Entity Create Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entities **and** Header: Content-Type **set to** application/ld+json **and** SUT ← Client

	Header: Link **set to** a @context **containing** terms used by the entity to create body **set to** entity to be created **}**	
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request) **and** Response Body **containing** ProblemDetails element **containing** type element **set to** https://uri.etsi.org/ngsi-ld/errors/BadRequestData **and** title element **containing** more information about the error **}**	SUT → Client

4.1.1.1.2 Delete Entity

TP Id	TP/NGSI-LD/CI/Prov/E/002_01	
Test objective	Check that you can delete an entity by id	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.6	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_6	
Initial conditions	**with {** **the SUT being** in the "initial state" and **containing** an initial Entity with an id **set to** ${entityId} **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Delete Entity Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entities/${entityId} **and** method **set to** DEL **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 204 (No Content) **and does not contain** an entity **with** ${entityId} **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/Prov/E/002_02	
Test objective	Check that you cannot delete an entity with invalid/missing id	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.6	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_6	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** an invalid Delete Entity Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entities/**{entityId_invalid} and** method **set to** DEL **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request) **and** Response Body **containing** ProblemDetails element **containing** type element **set to ${problem_type} and** title element **containing** more information about the error **}**	SUT → Client

Permutation on TP Id	${entityId_invalid}	${problem_type}
TP/NGSI-LD/CI/Prov/E/002_02_01	Empty	https://uri.etsi.org/ngsi-ld/errors/BadRequestData
TP/NGSI-LD/CI/Prov/E/002_02_02	invalid URI	https://uri.etsi.org/ngsi-ld/errors/BadRequestData

TP Id	TP/NGSI-LD/CI/Prov/E/002_03
Test objective	Check that you cannot delete an entity if the entity id is not known to the system
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.6
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_5_6_6
Initial conditions	**with {** **the SUT being** in the "initial state" **}**

Expected behaviour	Test events	Direction
	when { **the SUT receives** a valid Delete Entity Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entities/{entityId_notFound} **and** method **set to** DEL **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 404 (Not Found) **and** Response Body **containing** ProblemDetails element **containing** type element **set to** https://uri.etsi.org/ngsi-ld/errors/ResourceNotFound **and** title element **containing** more information about the error**}**	SUT → Client

4.1.1.2 Batch Entities

4.1.1.2.1 Create batch of Entities

TP Id	TP/NGSI-LD/CI/Prov/BE/003_01	
Test objective	Check that you can create a batch of entities	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.7	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_7	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Batch Entity Create Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entityOperations/create **and** Header: Content-Type **set to** application/ld+json **and** body **set to** array of **${entities}** to be created **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 201 (Created) **and**	SUT → Client

	Response Body **set to** an array of created entities ids **and** created resources **set to ${entities}** **}**	

Permutation on TP Id	${entities}
TP/NGSI-LD/CI/Prov/BE/003_01_01	A list of (minimal entity)
TP/NGSI-LD/CI/Prov/BE/003_01_02	A list of (entity having only properties)
TP/NGSI-LD/CI/Prov/BE/003_01_03	A list of (entity having only relationships)
TP/NGSI-LD/CI/Prov/BE/003_01_04	A list of (entity having properties and relationships)

TP Id	TP/NGSI-LD/CI/Prov/BE/003_02
Test objective	Check that you can create a batch of entities where some will succeed and others will fail
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.7
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_5_6_7

Initial conditions	with {
	the SUT being in the "initial state"
	}

Expected behaviour	**Test events**
	when { **the SUT receives** a valid Batch Entity Create Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entityOperations/create **and** Header: Content-Type **set to** application/ld+json **and** body **set to** JSON-LD Array of two valid entities and one invalid entity to be created **}**
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 207 (Multi Status) **and** Response Body **containing** BatchOperationResult element **containing** success element **set to** URIs of the successfully created entities **and** errors element **containing** information about the error for each of the entities that could not be created **and** created resources **set to the** two valid entities **}**

TP Id	TP/NGSI-LD/CI/Prov/BE/003_03	
Test objective	Check that you cannot create a batch of entities with an invalid request	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.7	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_7	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** an invalid Batch Entity Create Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entityOperations/create **and** Header: Content-Type **set to** application/ld+json **and** body **set to ${invalid_body}** **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request) **and** Response Body **containing** ProblemDetails element **containing** type element **set to ${problem_type} and** title element **containing** more information about the error **}**	SUT → Client

Permutation on TP Id	${invalid_body}	${problem_type}
TP/NGSI-LD/CI/Prov/BE/003_03_01	invalid JSON document	https://uri.etsi.org/ngsi-ld/errors/InvalidRequest
TP/NGSI-LD/CI/Prov/BE/003_03_02	empty	https://uri.etsi.org/ngsi-ld/errors/BadRequestData

TP Id	TP/NGSI-LD/CI/Prov/BE/003_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 V1.3.1 [1], clause 5.6.7
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_6_3_5
Initial conditions	**with {** **the SUT being** in the "initial state"

	}	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Batch Entity Create Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entityOperations/create **and** Header: Content-Type **set to** application/json **and** Header: Link **set to** a @context **containing** terms used by the entity to create body **set to** entity to be created **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 201 (Created) **and** Response Body **set to** an array with the created entity id **and** Persisted Entity contains type and attributes expanded as per the supplied @context **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/Prov/BE/003_05
Test objective	Check that the default @context is used if the Content-Type header is "application/json" and the Link header does not contain a JSON-LD @context
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.7
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_6_3_5
Initial conditions	**with {** **the SUT being** in the "initial state" **}**

Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Batch Entity Create Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entityOperations/create **and** Header: Content-Type **set to** application/json **and** body **set to** entity to be created **}**	SUT ← Client

	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 201 (Created) **and** Response Body **set to** an array of created entities ids **and** Persisted Entity contains type and attributes expanded as per the default @context **}**	SUT → Client

TP Id	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 V1.3.1 [1], clause 5.6.7	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_6_3_5	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Batch Entity Create Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entityOperations/create **and** Header: Content-Type **set to** application/json **and** body **set to** entity **containing** a @context term **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request) **and** Response Body **containing** ProblemDetails element **containing** type element set to https://uri.etsi.org/ngsi-ld/errors/BadRequestData **and** title element **containing** more information about the error **}**	SUT → Client

TP Id	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 V1.3.1 [1], clause 5.6.7	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_6_3_5	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Batch Entity Create Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entityOperations/create **and** Header: Content-Type **set to** application/ld+json **and** body **set to** entity **containing** a @context term **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 201 (Created) **and** Response Body **set to** an array with the created entity id **and** Persisted Entity contains type and attributes expanded as per the supplied @context **}**	SUT → Client

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 V1.3.1 [1], clause 5.6.7	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_6_3_5	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Batch Entity Create Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entityOperations/create **and**	SUT ← Client

	Header: Content-Type **set to** application/ld+json **and** body **set to** entity **not containing** a @context term **}**	
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request) **and** Response Body **containing** ProblemDetails element **containing** type element **set to** https://uri.etsi.org/ngsi-ld/errors/BadRequestData **and** title element **containing** more information about the error **}**	SUT → Client

TP Id	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 V1.3.1 [1], clause 5.6.7	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_6_3_5	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Batch Entity Create Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entityOperations/create **and** Header: Content-Type **set to** application/ld+json **and** Header: Link **set to** a @context **containing** terms used by the entity to create body **set to** entity to be created **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request) **and** Response Body **containing** ProblemDetails element **containing**	SUT → Client

	type element **set** to https://uri.etsi.org/ngsi-ld/errors/BadRequestData **and** title element **containing** more information about the error **}**	

4.1.1.2.2 Upsert batch of Entities

TP Id	TP/NGSI-LD/CI/Prov/BE/004_01	
Test objective	Check that you can upsert a batch of non-existing entities and they will be created	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.8	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_8	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Batch Entity Upsert Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entityOperations/upsert **and** Header: Content-Type **set to** application/ld+json **and** body **set to** array of **${entities}** to be upserted **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 201 (Created) **and** Response Body **set to** an array of created entities ids **and** updated resources **set to ${entities}** **}**	SUT → Client

Permutation on TP Id	${entities}
TP/NGSI-LD/CI/Prov/BE/004_01_01	A list of (entity having only properties)
TP/NGSI-LD/CI/Prov/BE/004_01_02	A list of (entity having only relationships)
TP/NGSI-LD/CI/Prov/BE/004_01_03	A list of (entity having properties and relationships)

TP Id	TP/NGSI-LD/CI/Prov/BE/004_02
Test objective	Check that you can upsert a batch of non-existing and existing entities where non-existing will be created and existing will be replaced
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.8

Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_8	
Initial conditions	**with {** **the SUT being** in the "initial state" and **containing ${existing_entities}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Batch Entity Upsert Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entityOperations/upsert **and** Header: Content-Type **set to** application/ld+json **and** body **set to** array of **${new_existing_entities} and ${non_existing_entities}** **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 201 (Created) **and** Response Body **set to** an array of **${non_existing_entities}** ids **and** created resources **set to ${non_existing_entities}** **and** updated resources **set to ${new_existing_entities}** **}**	SUT → Client

Permutation on TP Id	${existing_entities}	${non_existing_entities}
TP/NGSI-LD/CI/Prov/BE/004_02_01	A list of (entity having properties and relationships)	A list of (entity having only properties)
TP/NGSI-LD/CI/Prov/BE/004_02_02	A list of (entity having properties and relationships)	A list of (entity having only relationships)
TP/NGSI-LD/CI/Prov/BE/004_02_03	A list of (entity having properties and relationships)	A list of (entity having properties and relationships)

TP Id	TP/NGSI-LD/CI/Prov/BE/004_03
Test objective	Check that you can upsert a batch of existing entities and they will be replaced
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.8
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_5_6_8
Initial conditions	**with {** **the SUT being** in the "initial state" and **containing** existing entities **}**

Expected behaviour	Test events	Direction
	when { **the SUT receives** a valid Batch Entity Upsert Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entityOperations/upsert **and** Header: Content-Type **set to** application/ld+json **and** body **set to** array of **${entities}** **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 204 (No Content) **and** updated resources **set to ${entities}** **}**	SUT → Client

Permutation on TP Id	${existing_entities}
TP/NGSI-LD/CI/Prov/BE/004_03_01	A list of (entity having only properties)
TP/NGSI-LD/CI/Prov/BE/004_03_02	A list of (entity having only relationships)
TP/NGSI-LD/CI/Prov/BE/004_03_03	A list of (entity having properties and relationships)

TP Id	TP/NGSI-LD/CI/Prov/BE/004_04	
Test objective	Check that you can upsert a batch of entities with update option	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.8	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_8	
Initial conditions	**with {** **the SUT being** in the "initial state" and **containing ${existing_entities}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Batch Entity Upsert Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entityOperations/upsert **and** Header: Content-Type **set to** application/ld+json **and** Query parameter: options **set to** update **and** body **set to** array of **${new_existing_entites} and ${non_existing_entities}** **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 201 (Created) **and** Response Body **set to** an array of created entities ids	SUT → Client

	and created resources **set to the ${non_existing_entities}** **and** updated resources **set to ${existing_entities}** updated with attributes **in ${new_existing_entites}** **}**	

Permutation on TP Id	${existing_entities}	${non_existing_entities}
TP/NGSI-LD/CI/Prov/BE/004_04_01	A list of (entity having properties and relationships)	A list of (entity having only properties)
TP/NGSI-LD/CI/Prov/BE/004_04_02	A list of (entity having properties and relationships)	A list of (entity having only relationships)
TP/NGSI-LD/CI/Prov/BE/004_04_03	A list of (entity having properties and relationships)	A list of (entity having properties and relationships)

TP Id	TP/NGSI-LD/CI/Prov/BE/004_05	
Test objective	Check that you can upsert a batch of entities where some will succeed and others will fail	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.8	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_8	
Initial conditions	**with {** **the SUT being** in the "initial state" and **containing** existing entities **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Batch Entity Upsert Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entityOperations/upsert **and** Header: Content-Type **set to** application/ld+json **and** body **set to** JSON-LD Array of two valid entities and one invalid entity **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 207 (Multi Status) **and** Response Body **containing** BatchOperationResult element **containing** success element **set to** Ids of successfully created or updated entities **and** errors element **containing** information about the error for each of the entities that could not be created or updated	SUT → Client

	and updated resources **set to the** two valid entities **}**	

TP Id	TP/NGSI-LD/CI/Prov/BE/004_06	
Test objective	Check that you cannot upsert a batch of entities with an invalid request	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.8	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_8	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** an invalid Batch Entity Upsert Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entityOperations/upsert **and** Header: Content-Type **set to** application/ld+json **and** body **set to ${invalid_body}** **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request) **and** Response Body **containing** ProblemDetails element **containing** type element **set to ${problem_type} and** title element **containing** more information about the error **}**	SUT → Client

Permutation on TP Id	${invalid_body}	${problem_type}
TP/NGSI-LD/CI/Prov/BE/004_06_01	Is invalid JSON document	https://uri.etsi.org/ngsi-ld/errors/InvalidRequest
TP/NGSI-LD/CI/Prov/BE/004_06_02	Contains a null value in any of its items	https://uri.etsi.org/ngsi-ld/errors/BadRequestData

4.1.1.2.3 Update batch of Entities

TP Id	TP/NGSI-LD/CI/Prov/BE/005_01	
Test objective	Check that you can update a batch of entities	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.9	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_9	
Initial conditions	**with {** **the SUT being** in the "initial state" and **containing ${existing_entities}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Batch Entity Update Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entityOperations/update **and** Header: Content-Type **set to** application/ld+json **and** body **set to** array of **${entities}** **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 204 (No Content) **and** updated resources **set to ${existing_entities}** updated with attributes **in ${entities}** **}**	SUT → Client

Permutation on TP Id	${entities}
TP/NGSI-LD/CI/Prov/BE/005_01_01	A list of (entity having only properties)
TP/NGSI-LD/CI/Prov/BE/005_01_02	A list of (entity having only relationships)
TP/NGSI-LD/CI/Prov/BE/005_01_03	A list of (entity having properties and relationships)

TP Id	TP/NGSI-LD/CI/Prov/BE/005_02
Test objective	Check that you can update a batch of entities with noOverwrite option
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.9
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_5_6_9
Initial conditions	**with {** **the SUT being** in the "initial state" and **containing ${existing_entities}** **}**

Expected behaviour	Test events	Direction
	when { **the SUT receives** a valid Batch Entity Update Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entityOperations/update **and** Header: Content-Type **set to** application/ld+json **and** Query Parameter: options **set to** noOverwrite **and** body **set to** array of **${entities}** **}**	SUT ← Client
	then { **the SUT** updates the requested entities without overwriting existing attributes **and** **sends** a valid Response **containing** Response Status Code **set to** 204 (No Content) **and** updated resources **set to ${existing_entities} with** new attributes from **${entities}** appended **}**	SUT → Client

Permutation on TP Id	${entities}
TP/NGSI-LD/CI/Prov/BE/005_02_01	A list of (entity having only properties)
TP/NGSI-LD/CI/Prov/BE/005_02_02	A list of (entity having only relationships)
TP/NGSI-LD/CI/Prov/BE/005_02_03	A list of (entity having properties and relationships)

TP Id	TP/NGSI-LD/CI/Prov/BE/005_03	
Test objective	Check that you can update a batch of entities where some will succeed and others will fail	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.9	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_9	
Initial conditions	**with {** **the SUT being** in the "initial state" and **containing ${existing_entities}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Batch Entity Update Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entityOperations/update **and** Header: Content-Type **set to** application/ld+json **and** body **set to** array of **${new_existing_entities} and ${non_existing_entities}** **}**	SUT ← Client

	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 207 (Multi Status) **and** Response Body **containing** BatchOperationResult element **containing** success element **set to** Ids of successfully updated entities **and** errors element **containing** information about the error for each of the entities that could not be updated **and** updated resources **set to ${existing_entities}** updated with attributes **in ${new_existing_entities}** **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/Prov/BE/005_04	
Test objective	Check that you cannot update a batch of entities with an invalid request	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.9	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_9	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** an invalid Batch Entity Update Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entityOperations/update **and** Header: Content-Type **set to** application/ld+json **and** body **set to ${invalid_body}** **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request) **and** Response Body **containing** ProblemDetails element **containing** type element **set to ${problem_type} and**	SUT → Client

	title element **containing** more information about the error **}**	

Permutation on TP Id	${invalid_body}	${problem_type}
TP/NGSI-LD/CI/Prov/BE/005_04_01	Is invalid JSON document	https://uri.etsi.org/ngsi-ld/errors/InvalidRequest
TP/NGSI-LD/CI/Prov/BE/005_04_02	Is not syntactically correct according to the @context	https://uri.etsi.org/ngsi-ld/errors/BadRequestData

4.1.1.2.4 Delete batch of Entities

TP Id	TP/NGSI-LD/CI/Prov/BE/006_01	
Test objective	Check that you can delete a batch of entities	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.10	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_10	
Initial conditions	**with {** **the SUT being** in the "initial state" and **containing** existing entities **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Batch Entity Delete Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entityOperations/delete **and** Header: Content-Type **set to** application/ld+json **and** body **set to** JSON-LD Array of **${entities_ids}** to be deleted **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 204 (No Content) **and the SUT not containing** resources **with** id **in ${entities_ids}** **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/Prov/BE/006_02
Test objective	Check that you can delete a batch of entities where some will succeed and others will fail
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.10
Config Id	CF_01
Parent Release	V1.3.1

PICS Selection	PICS_5_6_10	
Initial conditions	**with {** **the SUT being** in the "initial state" and **containing** existing entities **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Batch Entity Delete Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entityOperations/delete **and** Header: Content-Type **set to** application/ld+json **and** body **set to** JSON-LD Array of ${**existing_entities_ids**} and ${**non-existing_entities_ids**} to be deleted **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 207 (Multi Status) **and** Response Body **containing** BatchOperationResult element **containing** success element **set to** Ids of successfully deleted entities **and** errors element **containing** information about the error for each of the entities that could not be deleted **and the SUT not containing** resources **with** id **in** **${existing_entities_ids}** **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/Prov/BE/006_03
Test objective	Check that you cannot delete a batch of entities with an invalid request
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.10
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_5_6_10
Initial conditions	**with {** **the SUT being** in the "initial state" **}**

Expected behaviour	Test events	Direction
	when { **the SUT receives** an invalid Batch Entity Delete Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entityOperations/delete **and** Header: Content-Type **set to** application/ld+json **and** body **set to ${invalid_body}** **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request) **and** Response Body **containing** ProblemDetails element **containing** type element **set to ${problem_type} and** title element **containing** more information about the error **}**	SUT → Client

Permutation on TP Id	${invalid_body}	${problem_type}
TP/NGSI-LD/CI/Prov/BE/006_03_01	Is invalid JSON document	https://uri.etsi.org/ngsi-ld/errors/InvalidRequest
TP/NGSI-LD/CI/Prov/BE/006_03_02	Is empty	https://uri.etsi.org/ngsi-ld/errors/BadRequestData

4.1.1.3 Temporal Entity

4.1.1.3.1 Create temporal representation of Entity

TP Id	TP/NGSI-LD/CI/PROV/TE/007_01
Test objective	Check that you can create a temporal representation of an entity
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.11
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_5_6_11
Initial conditions	**with {** **the SUT being** in the "initial state" **}**

Expected behaviour	Test events	Direction
	when { **the SUT receives** a valid Create Temporal Entity Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entities **and** method **set to** POST **and** Header: Content-Type **set to** ${contentType} **and** body **set to** ${entityTemporal} to be created **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 201 (CREATED) Created Entity **set to** ${entity} **}**	SUT → Client

Permutation on TP Id	${contentType}	${entityTemporal}
007_01_01	Application/json	entity with simplified temporal representation of an Entity (clause 4.5.9)
007_01_02	Application/json+ld	entity with simple temporal properties (arrays of (Property or Relationship) instances represented by JSON-LD objects) as defined in clauses 4.5.7 and 4.5.8
007_01_03	Application/json	No Context

TP Id	TP/NGSI-LD/CI/PROV/TE/007_02	
Test objective	Check that you cannot create a temporal entity with an invalid request	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.11	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_11	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Temporal Entity Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entities **and** method **set to** POST **and** Header: Content-Type **set to** ${contentType} **and** body **set to** ${invalid_body} to be created **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing**	SUT → Client

	Response Status Code **set to** 400 (Bad Request) **and** Response Body **containing** ProblemDetails element **containing** type element **set to ${problem_type} and** title element **containing** more information about the error **}**	

Permutation on TP Id	${invalid_body}	${problem_type}
007_02_01	Is not syntactically correct according to the @context	https://uri.etsi.org/ngsi-ld/errors/BadRequestData
007_02_02	Is invalid JSON document	https://uri.etsi.org/ngsi-ld/errors/InvalidRequest

4.1.1.3.2 Update temporal representation of Entity

TP Id	TP/NGSI-LD/CI/PROV/TE/008_01	
Test objective	Check that you can update a temporal representation of an entity with simple temporal properties (arrays of (Property or Relationship) instances represented by JSON-LD objects) as defined in clauses 4.5.7 and 4.5.8	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.11	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_11	
Initial conditions	**with {** **the SUT being** in the "initial state" with at least one entity with a temporal representation **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Update Temporal Entity Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entities **and** method **set to** POST **and** Header: Content-Type **set to** ${contentType} **and** body **set to** ${entityTemporal} to be created **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 204 Updated Entity **set to** ${entityTemporal} **}**	SUT → Client

Permutation on TP Id	${contentType}	${entityTemporal}
008_01_01	Application/json	entity with simplified temporal representation of an Entity (clause 4.5.9)
008_01_02	Application/json+ld	entity with simple temporal properties (arrays of (Property or Relationship) instances represented by JSON-LD objects) as defined in clauses 4.5.7 and 4.5.8

4.1.1.3.3 Add Attributes to Temporal Representation of an Entity

TP Id	TP/NGSI-LD/CI/PROV/TEA/014_01	
Test objective	Check that you can add a simple temporal attribute to a temporal representation of an entity	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.12	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_12	
Initial conditions	**with {** **the SUT being** in the "initial state" **containing** an initial Entity **${entity}** with an id **set to ${entityId}** an temporal attribute with an id **set to ${atrrId}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Add Temporal Attribute Request **from** the client **containing** URL **set to** /ngsi-ld/v1/ entities/${entityId}/attrs **and** method **set to** POST **and** Header: Content-Type **set to** ${contentType} **and** body **set to** ${entityTemporalFragment} to be created **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 204 **}**	SUT → Client

Permutation on TP Id	${entityId}	${contentType}	${entityTemporalFragment}
014_01_01	Existing Id	Application/json	Simplified temporal attribute (clause 4.5.9)
014_01_02	Existing Id	Application/json+ld	entity with simple temporal properties (arrays of (Property or Relationship) instances represented by JSON-LD objects) as defined in clauses 4.5.7 and 4.5.8
014_01_03	Existing Id	Application/json	No Context

TP Id	TP/NGSI-LD/CI/PROV/TEA/014_02	
Test objective	check that an error BadRequestData is raised if you add an attribute to a non-existing Entity (id not present)	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.12	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_12	
Initial conditions	**with {** **the SUT containing no** Entity with an id **set to ${entityId}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Add Temporal attribute to non-existing entity **from** the client **containing** URL **set to** /ngsi-ld/v1/ entities/${EntityId}/attrs **and** method **set to** POST **and** Header: Content-Type **set to** ${contentType} **and** body **set to** ${entityTemporalFragment} to be created **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request) and Response Body **containing** ProblemDetails element **containing** type element **set to ${problem_type} and** title element **containing** more information about the error **}**	SUT → Client

Permutation on TP Id	${EntityId)	${entityTemporalFragment} to	${problem_type}
014_02_01	Non Existing	valid	https://uri.etsi.org/ngsi-ld/errors/BadRequestData
014_02_02	invalid URI	valid	https://uri.etsi.org/ngsi-ld/errors/BadRequestData

TP Id	TP/NGSI-LD/CI/PROV/TEA/ 014_03_01
Test objective	check that an error **ResourceNotFound** is raised if you add an attribute with an endpoint that has no existing temporal representation of an Entity with the passed id
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.12
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_5_6_12

Initial conditions	**with {** **containing** an initial Entity **${entity}** with an id **set to ${entityId} and** no existing temporal attribute **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Add temporal attribute to a non-existing temporal representation of an existing entity **from** the client **containing** URL **set to** /ngsi-ld/v1/ entities/${EntityId}/attrs **and** method **set to** POST **and** Header: Content-Type **set to** ${contentType} **and** body **set to** ${entityTemporalFragment} to be created **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 404 (Not Found) and Response Body **containing** ProblemDetails element **containing** type element **set to ${problem_type} and** title element **containing** more information about the error **}**	SUT → Client

Permutation on TP Id	${EntityId)	${entityTemporalFragment} to	${problem_type}
014_03_01	Existing, no existing temporal representation	valid	https://uri.etsi.org/ngsi-ld/errors/ResourceNotFound

4.1.1.3.4 Delete Attribute from Temporal Representation of an Entity

TP Id	TP/NGSI-LD/CI/PROV/TEA/015_01	
Test objective	Check that you can delete an attribute of a temporal representation of an entity with simplified temporal representation of an Entity (clause 4.5.9) by id	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.13	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_13	
Initial conditions	**with {** **the SUT containing** an initial Entity **${entity}** with an id **set to ${entityId} and** an attribute with an id **set to ${atrrId}** **}**	
Expected behaviour	**Test events**	**Direction**
		SUT ← Client

	when { **the SUT receives** a valid Delete temporal attribute of an existing entity **from** the client **containing** URL **set to** /ngsi-ld/v1/ entities/${EntityId}/attrs/${attrId} **and** method **set to** DEL **and** Header: Content-Type **set to** ${contentType} **}**	
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 204 **}**	SUT → Client

Permutation on TP Id	${EntityId)	${attrId}	${deleteAll}	${contentType}
015_01_01	Existing	Existing	Not present	Application/json
015_01_02	Existing	Existing	Not present	Application/json
015_01_03	Existing	Existing	Not present	Application/json+ld

TP Id	TP/NGSI-LD/CI/PROV/TEA/015_02	
Test objective	Check that you can delete an attribute of a temporal representation of an entity with simplified temporal representation of an Entity (clause 4.5.9) by id. Check that if the deleteAll flag is set, all target attributes are deleted from the target temporal Entity	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.13	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_13	
Initial conditions	**with {** **the SUT containing** an initial Entity **${entity}** with an id **set to ${entityId} and** an attribute with an id **set to ${atrrId}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Delete temporal attribute of an existing entity **from** the client **containing** URL **set to** /ngsi-ld/v1/ entities/${EntityId}/attrs/${attrId} **and** method **set to** DEL **and** Header: Content-Type **set to** ${contentType} **and** Query Parameter: datasetId **set to** ${datasetId} **and** Query Parameter: deleteAll **set to** ${deleteAll} **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing**	SUT → Client

	Response Status Code **set to 2**04 **}**	

Permutation on TP Id	${EntityId)	${attrId}	${datasetID}	${deleteAll}
015_02_01	Existing	Existing	Not provided	True
015_02_02	Existing	Existing	Valid	Not set
015_02_03	Existing	Existing	Not provided	Not set

TP Id	TP/NGSI-LD/CI/PROV/TEA/015_04	
Test objective	Check that you an error **BadRequestData** is raised if you delete an attribute to a non-existing Entity (id not present)	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.13	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_13	
Initial conditions	**with {** **the SUT containing no** Entity with an id **set to ${entityId}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Delete Temporal attribute to non-existing entity **from** the client **containing** URL **set to** /ngsi-ld/v1/ entities/${EntityId}/attrs/${attrId} **and** method **set to** DEL **and** Header: Content-Type **set to** ${contentType} **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request) and Response Body **containing** ProblemDetails element **containing** type element **set to ${problem_type} and** title element **containing** more information about the error **}**	SUT → Client

Permutation on TP Id	${EntityId)	${attr	${problem_type}
015_04_01	Non Existing	valid	https://uri.etsi.org/ngsi-ld/errors/BadRequestData
015_04_02	invalid URI	valid	https://uri.etsi.org/ngsi-ld/errors/BadRequestData

TP Id	TP/NGSI-LD/CI/PROV/TEA/015_05	
Test objective	check that an error **BadRequestData** is raised if you delete an attribute with an invalid attribute Name of an temporal entity	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.13	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_13	
Initial conditions	**with {** **the SUT containing** Entity with an id **set to ${entityId}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Delete Temporal attribute to non-existing entity **from** the client **containing** URL **set to** /ngsi-ld/v1/ entities/${EntityId}/attrs/${attrId} **and** method **set to** DEL **and** Header: Content-Type **set to** ${contentType} **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request) and Response Body **containing** ProblemDetails element **containing** type element **set to ${problem_type} and** title element **containing** more information about the error **}**	SUT → Client

Permutation on TP Id	${EntityId}	${attrID}	${problem_type}
015_05	Non Existing	Invalid	https://uri.etsi.org/ngsi-ld/errors/BadRequestData

TP Id	TP/NGSI-LD/CI/PROV/TE/015_06
Test objective	check that an error **ResourceNotFound** is raised if you delete an attribute with an endpoint that has no existing temporal representation of an Entity with the passed id
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.13
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_5_6_13

Initial conditions	**with {** **containing** an initial Entity **${entity}** with an id **set to ${entityId} and** no existing temporal attribute **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Delete temporal attribute to a non-existing temporal representation of an existing entity **from** the client **containing** URL **set to** /ngsi-ld/v1/ entities/${EntityId}/attrs/${attrId} **and** method **set to** DEL **and** Header: Content-Type **set to** ${contentType} **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 404 (Not Found) and Response Body **containing** ProblemDetails element **containing** type element **set to ${problem_type} and** title element **containing** more information about the error **}**	SUT → Client

Permutation on TP Id	${EntityId)	${attrId}	${problem_type}
015_06_01	Existing	Non existing temporal representation	https://uri.etsi.org/ngsi-ld/errors/ResourceNotFound
015_06_02	Existing	Non existing	https://uri.etsi.org/ngsi-ld/errors/ResourceNotFound

4.1.1.3.5 Partial update Attribute instance in Temporal Representation of an Entity

TP Id	TP/NGSI-LD/CI/PROV/TEA/016_01
Test objective	Check that you can update an attribute instance by its instanceId of a temporal representation of an entity with simple temporal properties (arrays of (Property or Relationship) instances represented by JSON-LD objects) as defined in clauses 4.5.7 and 4.5.8
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.14
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_5_6_14
Initial conditions	**with {** **the SUT being** in the "initial state" **containing** an initial Entity **${entity}** with an id **set to ${entityId}** an temporal attribute with an id **set to ${atrrId}** and an existing **${instanceID}** **}**

Expected behaviour	Test events	Direction
	when { **the SUT receives** a valid Update Temporal Attribute Instance Request **from** the client **containing** URL **set to** /ngsi-ld/v1/ entities/${entityId}/attrs/${attrId}/${instanceId} **and** method **set to** POST **and** Header: Content-Type **set to** ${contentType} **and** body **set to** ${entityTemporalFragment} to be created **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 204 and Updated temporal entity fragment set to ${entityTemporalFragment} **}**	SUT → Client

Permutation on TP Id	${InstanceId}	${contentType}	${entityTemporalFragment}
016_01_01	Existing Id	Application/json	Simplified temporal attribute (clause 4.5.9)
016_01_02	Existing Id	Application/json+ld	entity with simple temporal properties (arrays of (Property or Relationship) instances represented by JSON-LD objects) as defined in clauses 4.5.7 and 4.5.8
016_01_03	Existing Id	Application/json	No Context

TP Id	TP/NGSI-LD/CI/PROV/TE/016_02
Test objective	Check that you an error **BadRequestData** is raised if you update an attribute instance by its instanceId of a non-existing temporal Entity (id not present)
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.14
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_5_6_14
Initial conditions	**with {** **the SUT being** in the "initial state" **containing** no Entity **${entity}** with an id **set to ${entityId}** **}**

Expected behaviour	Test events	Direction
	when { **the SUT receives** a valid Update Temporal Attribute Instance Request **from** the client **containing** URL **set to** /ngsi-ld/v1/ entities/${entityId}/attrs/${attrId}/${instanceId} **and** method **set to** POST **and** Header: Content-Type **set to** ${contentType} **and** body **set to** ${entityTemporalFragment} to be created **}**	SUT ← Client

	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request) and Response Body **containing** ProblemDetails element **containing** type element **set to ${problem_type} and** title element **containing** more information about the error **}**	SUT → Client

Permutation on TP Id	${EndityId}	${InstanceId}	${problem_type}
016_02_01	Not existing	n/a	https://uri.etsi.org/ngsi-ld/errors/BadRequestData
016_02_02	Invalid URI	n/a	https://uri.etsi.org/ngsi-ld/errors/BadRequestData
016_02_03	Invalid Name	n/a	https://uri.etsi.org/ngsi-ld/errors/BadRequestData

TP Id	TP/NGSI-LD/CI/PROV/TE/016_03	
Test objective	check that an error **BadRequestData** is raised if you update an attribute instance by its instanceId of a temporal Entity with an not valid URI for the instanceId	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.14	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_14	
Initial conditions	**with {** **the SUT being** in the "initial state" **containing** an initial Entity **${entity}** with an id **set to ${entityId}** an temporal attribute with an id **set to ${atrrId}** and an non existing **${instanceID}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Update Temporal Attribute Instance Request **from** the client **containing** URL **set to** /ngsi-ld/v1/ entities/${entityId}/attrs/${attrId}/${instanceId} **and** method **set to** POST **and** Header: Content-Type **set to** ${contentType} **and** body **set to** ${entityTemporalFragment} to be created **}**	SUT ← Client

	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request) and Response Body **containing** ProblemDetails element **containing** type element **set to ${problem_type} and** title element **containing** more information about the error **}**	SUT → Client

Permutation on TP Id	${EndityId}	${InstanceId}	${problem_type}
016_03_01	Existing	InstanceName not present	https://uri.etsi.org/ngsi-ld/errors/BadRequestData
016_03_02	Existing	Invalid URI	https://uri.etsi.org/ngsi-ld/errors/BadRequestData
016_03_03	Existing	Non Existing	https://uri.etsi.org/ngsi-ld/errors/BadRequestData

TP Id	TP/NGSI-LD/CI/PROV/TEA/016_05	
Test objective	check that the term expansion is applied when obtaining the target attribute to update an attribute instance by its instanceId of a temporal Entity	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.14	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_14	
Initial conditions	**with {** **the SUT being** in the "initial state" **containing** an initial Entity **${entity}** with an id **set to ${entityId}** an temporal attribute with an id **set to ${atrrId}** and an existing **${instanceID}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Update Temporal Attribute Instance Request **from** the client **containing** URL **set to** /ngsi-ld/v1/ entities/${entityId}/attrs/${attrId}/${instanceId} **and** method **set to** POST **and** Header: Content-Type **set to** ${contentType} **and** body **set to** ${entityTemporalFragment} to be created **}**	SUT ← Client

	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 204 **Check that response contains the complete term (applying term expansion)** **}**	SUT → Client
TP Id	TP/NGSI-LD/CI/PROV/TEA/016_04	
Test objective	check that an error **ResourceNotFound** is raised if you update an attribute instance by its instanceId of a temporal Entity with an endpoint that has no existing temporal representation of an Entity with the passed id	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.14	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_14	
Initial conditions	**with {** **the SUT being** in the "initial state" **containing** an initial Entity **${entity}** with an id **set to ${entityId} and** no temporal attribute with an id **set to ${atrrId}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Update Temporal Attribute Instance Request **from** the client **containing** URL **set to** /ngsi-ld/v1/ entities/${entityId}/attrs/${attrId}/${instanceId} **and** method **set to** POST **and** Header: Content-Type **set to** ${contentType} **and** body **set to** ${entityTemporalFragment} to be created **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request) and Response Body **containing** ProblemDetails element **containing** type element **set to ${problem_type} and** title element **containing** more information about the error **}**	SUT → Client

Permutation on TP Id	**${EnditytId}**	**${InstanceId}**	**${problem_type}**
016_04_01	Existing	Invalid URI	https://uri.etsi.org/ngsi-ld/errors/BadRequestData
004_04_02	Existing	Non existing	https://uri.etsi.org/ngsi-ld/errors/BadRequestData

4.1.1.3.6 Delete Attribute instance from Temporal Representation of an Entity

TP Id	TP/NGSI-LD/CI/PROV/TEA/017_01	
Test objective	Check that you can delete an attribute instance by its instanceId of a temporal representation of an entity with simple temporal properties (arrays of (Property or Relationship) instances represented by JSON-LD objects) as defined in clauses 4.5.7 and 4.5.8	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.15	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_15	
Initial conditions	**with {** **the SUT being** in the "initial state" **containing** an initial Entity **${entity}** with an id **set to ${entityId}** an temporal attribute with an id **set to ${atrrId}** and an existing **${instanceID}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Delete Temporal Attribute Instance Request **from** the client **containing** URL **set to** /ngsi-ld/v1/ entities/${entityId}/attrs/${attrId}/${instanceId} **and** method **set to** DEL **and** Header: Content-Type **set to** ${contentType} **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 204 **and** Temporal attribute is none **}**	SUT → Client

Permutation on TP Id	${InstanceId}	${contentType}
017_01_01	Existing Id	Application/json
017_01_02	Existing Id	Application/json+ld

TP Id	TP/NGSI-LD/CI/PROV/TEA/017_02
Test objective	Check that you an error **BadRequestData** is raised if you delete an attribute instance by its instanceId of a non-existing temporal Entity (id not present)
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.15
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_5_6_15
Initial conditions	**with {** **the SUT being** in the "initial state" **not containing** an initial Entity **${entity}** with an id **set to ${entityId}** **}**

Expected behaviour	Test events	Direction
	when { **the SUT receives** a valid Delete Temporal Attribute Instance Request **from** the client **containing** URL **set to** /ngsi-ld/v1/ entities/${entityId}/attrs/${attrId}/${instanceId} **and** method **set to** DEL **and** Header: Content-Type **set to** ${contentType} **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request) and Response Body **containing** ProblemDetails element **containing** type element **set to ${problem_type} and** title element **containing** more information about the error **}**	SUT → Client

Permutation on TP Id	${EndityId}	${InstanceId}	${problem_type}
017_02_01	Non Existing	n/a	https://uri.etsi.org/ngsi-ld/errors/BadRequestData
017_02_02	Invalid uri	n/a	https://uri.etsi.org/ngsi-ld/errors/BadRequestData
017_02_03	Invalid Name	n/a	https://uri.etsi.org/ngsi-ld/errors/BadRequestData

TP Id	TP/NGSI-LD/CI/PROV/TEA/017_03	
Test objective	Check that an error **BadRequestData** is raised if you delete an attribute instance by its instanceId of a temporal Entity with an instance name that is not present	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.15	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_15	
Initial conditions	**with {** **the SUT being** in the "initial state" **containing** an initial Entity **${entity}** with an id **set to ${entityId}** an temporal attribute with an id **set to ${atrrId} and** non-existing **${instanceID** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Delete Temporal Attribute Instance Request **from** the client **containing** URL **set to** /ngsi-ld/v1/ entities/${entityId}/attrs/${attrId}/${instanceId} **and** method **set to** DEL **and** Header: Content-Type **set to** ${contentType} **}**	SUT ← Client

	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request) and Response Body **containing** ProblemDetails element **containing** type element **set to ${problem_type} and** title element **containing** more information about the error **}**	SUT → Client

Permutation on TP Id	${EndityId}	${InstanceId}	${problem_type}
017_03_01	Valid	Instance name not present	https://uri.etsi.org/ngsi-ld/errors/BadRequestData
017_03_02	Valid	Invalid uri	https://uri.etsi.org/ngsi-ld/errors/BadRequestData
017_03_03	Not Valid	Not existing	https://uri.etsi.org/ngsi-ld/errors/BadRequestData

TP Id	TP/NGSI-LD/CI/PROV/TEA/017_05	
Test objective	Check that the term expansion is applied when obtaining the target attribute to delete an attribute instance by its instanceId of a temporal Entity	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.15	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_15	
Initial conditions	**with {** **the SUT being** in the "initial state" **containing** an initial Entity **${entity}** with an id **set to ${entityId}** an temporal attribute with an id **set to ${atrrId}** and an existing **${instanceID}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Delete Temporal Attribute Instance Request **from** the client **containing** URL **set to** /ngsi-ld/v1/ entities/${entityId}/attrs/${attrId}/${instanceId} **and** method **set to** DEL **and** Header: Content-Type **set to** ${contentType} **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 204 and Check that term expansion is applied **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/PROV/TE/017_04	
Test objective	Check that an error **ResourceNotFound** is raised if you delete an attribute instance by its instanceId of a temporal Entity with an endpoint that has no existing temporal representation of an Entity with the passed id	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.15	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_15	
Initial conditions	**with {** **the SUT being** in the "initial state" **containing** an initial Entity **${entity}** with an id **set to ${entityId} and** no temporal attribute with an id **set to ${atrrId}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Delete Temporal Attribute Instance Request **from** the client **containing** URL **set to** /ngsi-ld/v1/ entities/${entityId}/attrs/${attrId}/${instanceId} **and** method **set to** DEL **and** Header: Content-Type **set to** ${contentType} **and** **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 404 (Not Found) and Response Body **containing** ProblemDetails element **containing** type element **set to ${problem_type} and** title element **containing** more information about the error **}**	SUT → Client

Permutation on TP Id	${attrsId}	${InstanceId}	${problem_type}
017_04_01	No temporal representation	n/a	https://uri.etsi.org/ngsi-ld/errors/ResourceNotFound
017_04_02	Non existing id	n/a	https://uri.etsi.org/ngsi-ld/errors/ResourceNotFound
017_04_03	Valid	Non existing id	https://uri.etsi.org/ngsi-ld/errors/ResourceNotFound

4.1.1.3.7 Delete temporal representation of Entity

TP Id	TP/NGSI-LD/CI/PROV/TE/009_01	
Test objective	Check that you can delete a temporal representation of an entity with simple temporal properties (arrays of (Property or Relationship) instances represented by JSON-LD objects) as defined in clauses 4.5.7 and 4.5.8	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.16	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_16	
Initial conditions	**with {** **the SUT being** in the "initial state" **containing** an initial Entity **${entity}** with an id **set to ${entityId}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Delete Temporal representation Request **from** the client **containing** URL **set to** /ngsi-ld/v1/ entities/${entityId} **and** method **set to** DEL **and** Header: Content-Type **set to** ${contentType} **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 204 and **}**	SUT → Client

Permutation on TP Id	${InstanceId}	${contentType}
009_01_01	Existing Id	Application/json
009_01_02	Existing Id	Application/json+ld

TP Id	TP/NGSI-LD/CI/PROV/TE/009_02
Test objective	Check that you an error **BadRequestData** is raised if you delete a temporal entity with a non-existing EntityId (id not present)
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.16
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_5_6_16
Initial conditions	**with {** **the SUT being** in the "initial state" **not containing** an initial Entity **${entity}** with an id **set to ${entityId}** **}**

Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Delete Temporal Attribute Instance Request **from** the client **containing** URL **set to** /ngsi-ld/v1/ entities/${entityId} **and** method **set to** DEL **and** Header: Content-Type **set to** ${contentType} **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request) and Response Body **containing** ProblemDetails element **containing** type element **set to ${problem_type} and** title element **containing** more information about the error **}**	SUT → Client

Permutation on TP Id	${EnditytId}	${InstanceId}	${problem_type}
009_02_01	Non Existing	n/a	https://uri.etsi.org/ngsi-ld/errors/BadRequestData
009_02_02	Invalid uri	n/a	https://uri.etsi.org/ngsi-ld/errors/BadRequestData
009_02_03	Invalid Name	n/a	https://uri.etsi.org/ngsi-ld/errors/BadRequestData

TP Id	TP/NGSI-LD/CI/PROV/TE/009_03	
Test objective	check that an error **ResourceNotFound** is raised if you delete a temporal Entity with an endpoint that has no existing temporal representation of an Entity with the passed id	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.16	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_16	
Initial conditions	**with {** **the SUT being** in the "initial state" **not containing** an initial Entity **${entity}** with an id **set to ${entityId}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Delete Temporal Attribute Instance Request **from** the client **containing** URL **set to** /ngsi-ld/v1/ entities/${entityId} **and**	SUT ← Client

	method **set to** DEL **and** Header: Content-Type **set to** ${contentType} **and** **}**	
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 404 (Not Found) and Response Body **containing** ProblemDetails element **containing** type element **set to ${problem_type} and** title element **containing** more information about the error **}**	SUT → Client

Permutation on TP Id	${entityID}	${problem_type}
009_03	No entity	https://uri.etsi.org/ngsi-ld/errors/ResourceNotFound

4.1.1.4 Entity Attributes

4.1.1.4.1 Append Entity Attributes

TP Id	TP/NGSI-LD/CI/Prov/EA/010_01	
Test objective	Check that you can append entity attributes	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.3	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_3	
Initial conditions	**with {** **the SUT containing** an initial Entity **${entity}** with an id **set to ${entityId}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Append Attribute request **from** the client **containing** URL **set to** /ngsi-ld/v1/entities/${entityId}/attrs/ **and** method **set to** POST **and** Header: Content-Type **set to** application/ld+json **and** Query Parameter: options **set to** ${overwrite} request body **set to** a valid JSON-LD representing an NGSI-LD Entity Fragment **containing** an attribute (Attribute A) with **${datasetId} and**	SUT ← Client

	an attribute (Attribute B) with no **datasetId** **}**	
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** ${status_code} Response Body **containing** ${appended_attrs_list} **and contains** ${entity} with ${appended_attrs_list} **}**	SUT → Client

Permutation on TP Id	${overwrite}	${datasetId}	${status_code}	${appended_attrs_list}
TP/NGSI-LD/CI/ Prov/EA/010_01_01	empty	equal	204	Empty (Attribute A is overwritten Attribute B is added)
TP/NGSI-LD/CI/ Prov/EA/010_01_02	noOverwrite	equal	207	Attribute B (Attribute A fails to overwrite Attribute B is added)
TP/NGSI-LD/CI/ Prov/EA/010_01_03	empty	different	204	Empty (Attribute A is added Attribute B is added)
TP/NGSI-LD/CI/ Prov/EA/010_01_04	noOverwrite	different	204	Empty (Attribute A is added Attribute B is added)

TP Id	TP/NGSI-LD/CI/Prov/EA/010_02	
Test objective	Check that you cannot append entity attributes with invalid/missing id or invalid request body	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.3	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_3	
Initial conditions	**with {** **the SUT containing** an initial Entity **${entity}** with an id **set to ${entityId}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Append Attribute request **from** the client **containing** URL **set to** /ngsi-ld/v1/entities/${entityId}/attrs/ **and** method **set to** POST **and** Header: Content-Type **set to** application/ld+json **and** Query Parameter: options **set to** ${overwrite} request body **set to** ${entity_fragment} **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing**	SUT → Client

	Response Status Code **set to** 400 (Bad Request) **and** Response Body **containing** ProblemDetails element **containing** type element **set to ${problem_type} and** title element **containing** more information about the error**}**	

Permutation on TP Id	${entityId}	${entity_fragment}	${problem_type}
TP/NGSI-LD/CI/Prov/EA/010_02_01	empty	valid	https://uri.etsi.org/ngsi-ld/errors/BadRequestData
TP/NGSI-LD/CI/Prov/EA/010_02_02	invalid URI	valid	https://uri.etsi.org/ngsi-ld/errors/BadRequestData
TP/NGSI-LD/CI/Prov/EA/010_02_03	valid	invalid	https://uri.etsi.org/ngsi-ld/errors/BadRequestData

TP Id	TP/NGSI-LD/CI/Prov/EA/010_03	
Test objective	Check that you cannot append entity attributes if the entity id or attributes are not known to the system	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.3	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_3	
Initial conditions	**with {** **the SUT** in the initial conditions **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Append Attribute request **from** the client **containing** URL **set to** /ngsi-ld/v1/entities/${entityId_notFound}/attrs/ **and** method **set to** POST **and** Header: Content-Type **set to** application/ld+json **and** Query Parameter: options **set to** ${overwrite} request body **set to** ${entity_fragment} **}**	SUT ← Client

	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 404 (Not Found) **and** Response Body **containing** ProblemDetails element **containing** type element **set to** https://uri.etsi.org/ngsi-ld/errors/ResourceNotFound **and** title element **containing** more information about the error **}**	SUT → Client

4.1.1.4.2 Update Entity Attributes

TP Id	TP/NGSI-LD/CI/Prov/EA/011_01	
Test objective	Check that you can update entity attributes	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.2	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_2	
Initial conditions	**with {** **the SUT containing** an initial Entity **${entity}** with an id **set to ${entityId}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Append Attribute request **from** the client **containing** URL **set to** /ngsi-ld/v1/entities/${entityId}/attrs/ **and** method **set to** PATCH **and** Header: Content-Type **set to** application/ld+json **and** request body **set to** a valid JSON-LD representing an NGSI-LD Entity Fragment **containing** an attribute (Attribute A) with **${datasetId_A}** and **${type_A}** an attribute (Attribute B) with **${datasetId_B}** and **${type_B}** **}**	SUT ← Client

	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** ${status_code} Response Body **containing** ${updated_attrs_list} **and contains** ${entity} with ${updated_attrs_list} **}**	SUT → Client

Permutation on TP Id	${datasetId_A}	${type_A}	${datasetId_B}	${type_B}	${status_code}	${updated_attrs_list}
TP/NGSI-LD/CI/Prov/EA/011_01_01	empty	equal	empty	equal	204	Attribute A is updated Attribute B is updated
TP/NGSI-LD/CI/Prov/EA/011_01_02	equal	equal	equal	equal	204	Attribute A is updated Attribute B is updated
TP/NGSI-LD/CI/Prov/EA/011_01_03	equal	not equal	equal	equal	207	Attribute A is not updated Attribute B is updated
TP/NGSI-LD/CI/Prov/EA/011_01_04	equal	not equal	equal	not equal	204	Attribute A is not updated Attribute B is not updated

TP Id	TP/NGSI-LD/CI/Prov/EA/011_02	
Test objective	Check that you cannot update entity attributes with invalid/missing id or invalid request body	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.2	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_2	
Initial conditions	**with {** **the SUT containing** an initial Entity **${entity}** with an id **set to ${entityId}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Append Attribute request **from** the client **containing** URL **set to** /ngsi-ld/v1/entities/${entityId}/attrs/ **and** method **set to** PATCH **and** Header: Content-Type **set to** application/ld+json **and** request body **set to** ${entity_fragment} **}**	SUT □ Client

	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request) **and** Response Body **containing** ProblemDetails element **containing** type element **set to ${problem_type} and** title element **containing** more information about the error **}**	SUT □ Client

Permutation on TP Id	${entityId}	${entity_fragment}	${problem_type}
TP/NGSI-LD/CI/Prov/EA/011_02_01	empty	valid	https://uri.etsi.org/ngsi-ld/errors/BadRequestData
TP/NGSI-LD/CI/Prov/EA/011_02_02	invalid URI	valid	https://uri.etsi.org/ngsi-ld/errors/BadRequestData
TP/NGSI-LD/CI/Prov/EA/011_02_03	valid	invalid	https://uri.etsi.org/ngsi-ld/errors/BadRequestData

TP Id	TP/NGSI-LD/CI/Prov/EA/011_03	
Test objective	Check that you cannot update entity attributes if the entity id or attributes are not known to the system	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.2	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_2	
Initial conditions	**with {** **the SUT** in the initial conditions **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Append Attribute request **from** the client **containing** URL **set to** /ngsi-ld/v1/entities/${entityId_notFound}/attrs/ **and** method **set to** PATCH **and** Header: Content-Type **set to** application/ld+json **and** request body **set to** ${entity_fragment} **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 404 (Not Found) **and** Response Body **containing** ProblemDetails element **containing** type element **set to** https://uri.etsi.org/ngsi-ld/errors/ResourceNotFound **and**	SUT → Client

	title element **containing** more information about the error **}**	

4.1.1.4.3 Partial Update Entity Attributes

TP Id	TP/NGSI-LD/CI/Prov/EA/012_01	
Test objective	Check that you can perform a partial update on an entity attribute	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.4	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_4	
Initial conditions	**with {** **the SUT containing** an initial Entity **${entity}** with an id **set to ${entityId} and** an attribute with an id **set to** ${atrrId} **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Append Attribute request **from** the client **containing** URL **set to** /ngsi-ld/v1/entities/${entityId}/attrs/${attrId} **and** method **set to** PATCH **and** Header: Content-Type **set to** application/ld+json **and** request body **set to** a valid JSON-LD representing an NGSI-LD Entity Fragment **containing** **${elements_of_attr}** and **${datasetId}** and **${type}** **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to ${status_code}** Response Body **containing** **${message}** **and contains** ${entity} with ${attrId} with ${elements_of_attr} **}**	SUT → Client

Permutation on TP Id	${elements_of_attr}	${datasetId}	${type}	${status_code}	${message}
TP/NGSI-LD/CI/Prov/EA/012_01_01	valid	empty	equal	204	empty
TP/NGSI-LD/CI/Prov/EA/012_01_02	valid	equal	equal	204	empty

TP Id	TP/NGSI-LD/CI/Prov/EA/012_02	
Test objective	Check that you cannot perform a partial update on an entity attribute with invalid/missing ids or invalid request body	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.4	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_4	
Initial conditions	**with {** **the SUT containing** an initial Entity **${entity}** with an id **set to ${entityId} and** an attribute with an id **set to ${atrrId}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Append Attribute request **from** the client **containing** URL **set to** /ngsi-ld/v1/entities/${entityId}/attrs/${attrId} **and** method **set to** PATCH **and** Header: Content-Type **set to** application/ld+json **and** request body **set to** a valid JSON-LD representing an NGSI-LD **${entity_fragment} containing** **${elements_of_attr}** and **${datasetId}** and **${type}** **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request) **and** Response Body **containing** ProblemDetails element **containing** type element **set to ${problem_type} and** title element **containing** more information about the error **}**	SUT → Client

Permutation on TP Id	${entityId}	${attrId}	${entity_fragment}	${problem_type}
TP/NGSI-LD/CI/Prov/EA/012_02_01	invalid	valid	valid	https://uri.etsi.org/ngsi-ld/errors/BadRequestData
TP/NGSI-LD/CI/Prov/EA/012_02_02	empty	valid	valid	https://uri.etsi.org/ngsi-ld/errors/BadRequestData
TP/NGSI-LD/CI/Prov/EA/012_02_03	valid	valid	Attribute name missing	https://uri.etsi.org/ngsi-ld/errors/BadRequestData
TP/NGSI-LD/CI/Prov/EA/012_02_04	valid	invalid	valid	https://uri.etsi.org/ngsi-ld/errors/BadRequestData
TP/NGSI-LD/CI/Prov/EA/012_02_05	valid	valid	${type} different	https://uri.etsi.org/ngsi-ld/errors/BadRequestData
TP/NGSI-LD/CI/Prov/EA/012_02_06	valid	valid	empty	https://uri.etsi.org/ngsi-ld/errors/BadRequestData

TP Id	TP/NGSI-LD/CI/Prov/EA/012_03	
Test objective	Check that you cannot perform a partial update on an entity attribute if the entity id or attribute is not known to the system	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.4	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_4	
Initial conditions	**with {** **the SUT containing** an initial Entity **${entity}** with an id **set to ${entityId} and** an attribute with an id **set to ${atrrId}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Append Attribute request **from** the client **containing** URL **set to** /ngsi-ld/v1/entities/${entityId}/attrs/${attrId} **and** method **set to** PATCH **and** Header: Content-Type **set to** application/ld+json **and** request body **set to** a valid JSON-LD representing an NGSI-LD Entity Fragment **containing** **${elements_of_attr}** and **${datasetId}** and **${type}** **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 404 (Not Found) **and** Response Body **containing**	SUT → Client

	ProblemDetails element **containing** type element **set to** https://uri.etsi.org/ngsi-ld/errors/ResourceNotFound **and** title element **containing** more information about the error **}**	

Permutation on TP Id	${entityId}	${attrId}	${datasetId}
TP/NGSI-LD/CI/Prov/EA/012_03_01	Not found	Found	Found
TP/NGSI-LD/CI/Prov/EA/012_03_02	Found	Found	Empty
TP/NGSI-LD/CI/Prov/EA/012_03_03	Found	Found	Not Found
TP/NGSI-LD/CI/Prov/EA/012_03_04	Found	Not Found	Found

4.1.1.4.4 Delete Entity Attributes

TP Id	TP/NGSI-LD/CI/Prov/EA/013_01	
Test objective	Check that you can delete an attribute from an entity	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.5	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_5	
Initial conditions	**with {** **the SUT containing** an initial Entity **${entity}** with an id **set to ${entityId} and** an attribute with an id **set to ${atrrId}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Append Attribute request **from** the client **containing** URL **set to** /ngsi-ld/v1/entities/${entityId}/attrs/${attrId} **and** method **set to** DEL **and** Header: Content-Type **set to** application/ld+json **and** Query Parameter: datasetId **set to** ${datasetId} **and** Query Parameter: deleteAll **set to** ${deleteAll} **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 204 **and** ${entity} does not contain ${attrId} **}**	SUT → Client

Permutation on TP Id	${datasetId}	${deleteAll}
TP/NGSI-LD/CI/Prov/EA/013_01_01	empty	false
TP/NGSI-LD/CI/Prov/EA/013_01_02	valid	false
TP/NGSI-LD/CI/Prov/EA/013_01_03	valid	true

TP Id	TP/NGSI-LD/CI/Prov/EA/013_02	
Test objective	Check that you cannot delete an attribute from an entity with invalid/missing ids	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.5	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_5	
Initial conditions	**with {** **the SUT containing** an initial Entity **${entity}** with an id **set to ${entityId} and** an attribute with an id **set to ${atrrId}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Append Attribute request **from** the client **containing** URL **set to** /ngsi-ld/v1/entities/${entityId}/attrs/${attrId} **and** method **set to** DEL **and** Header: Content-Type **set to** application/ld+json **and** Query Parameter: datasetId **set to** empty **and** Query Parameter: deleteAll **set to** false **}**	SUT □ Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 **}**	SUT □ Client

Permutation on TP Id	**${entityId}**	**${attrId}**
TP/NGSI-LD/CI/Prov/EA/013_02_01	empty	valid
TP/NGSI-LD/CI/Prov/EA/013_02_02	invalid	valid
TP/NGSI-LD/CI/Prov/EA/013_02_03	valid	empty

TP Id	TP/NGSI-LD/CI/Prov/EA/013_03
Test objective	Check that you cannot delete an attribute from an entity if the entity id or attribute is not known to the system
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.5
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_5_6_5
Initial conditions	**with {** **the SUT containing** an initial Entity **${entity}** with an id **set to ${entityId} and** an attribute with an id **set to ${atrrId}** **}**

Expected behaviour	Test events	Direction
	when { **the SUT receives** a valid Append Attribute request **from** the client **containing** URL **set to** /ngsi-ld/v1/entities/${entityId}/attrs/${attrId} **and** method **set to** DEL **and** Header: Content-Type **set to** application/ld+json **and** Query Parameter: datasetId **set to** empty **and** Query Parameter: deleteAll **set to** false **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 404 (Not Found) **and** Response Body **containing** ProblemDetails element **containing** type element **set to** https://uri.etsi.org/ngsi-ld/errors/ResourceNotFound **and** title element **containing** more information about the error **}**	SUT → Client

Permutation on TP Id	${entityId}	${attrId}	${datasetId}
TP/NGSI-LD/CI/Prov/EA/013_03_01	Not found	Found	Specified
TP/NGSI-LD/CI/Prov/EA/013_03_02	Found	Not found	Not specified
TP/NGSI-LD/CI/Prov/EA/013_03_03	Found	Found	Not Found

4.1.2 Consumption

4.1.2.1 Entity

4.1.2.1.1 Retrieve Entity

TP Id	TP/NGSI-LD/CI/Cons/E/018_01
Test objective	Check that you can get an entity by id
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.7.1
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS 5_7_1
Initial conditions	**with {** **the SUT containing** an initial Entity **${entity}** with an id **set to ${entityId}** **}**

Expected behaviour	Test events	Direction
	when { **the SUT receives** a valid Get Entity Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entities/**${entityId} and** method **set to** GET **and** Query Parameter attrs **set to** ${**attrs} and** Query Parameter geometryProperty **set to ${geometryProperty}** **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 200 (OK) **and** Response Body **containing** ${entity_representation} **}**	SUT → Client

Permutation on TP Id	${entity}	${attrs}	${geometryProperty}	${entity_representation}
TP/NGSI-LD/CI/Cons/E/018_01_01	Simple properties	empty	empty	All entity properties are returned
TP/NGSI-LD/CI/Cons/E/018_01_02	Simple properties	"PropertyA", "PropertyB"	empty	Entity representation contains the selected attributes
TP/NGSI-LD/CI/Cons/E/018_01_03	With a location attribute	empty	"location"	GeoJSON Feature is returned

TP Id	TP/NGSI-LD/CI/Cons/E/018_02	
Test objective	Check that you cannot get an entity with invalid/missing id	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.7.1	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_7_1	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** an invalid Get Entity Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entities/**{entityId_invalid} and** method **set to** GET **}**	SUT ← Client

	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request) **and** Response Body **containing** ProblemDetails element **containing** type element **set to ${problem_type} and** title element **containing** more information about the error **}**	SUT → Client

Permutation on TP Id	${entityId_invalid}	${problem_type}
TP/NGSI-LD/CI/Cons/E/018_02_01	empty	https://uri.etsi.org/ngsi-ld/errors/BadRequestData
TP/NGSI-LD/CI/Cons/E/018_02_02	invalid URI	https://uri.etsi.org/ngsi-ld/errors/BadRequestData

TP Id	TP/NGSI-LD/CI/Cons/E/018_03	
Test objective	Check that you cannot get an entity if the entity id or attributes are not known to the system	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.6.6	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_6_6	
Initial conditions	**with {** **the SUT containing** an initial Entity **${entity}** with an id **set to ${entityId}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Delete Entity Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entities/**${entityId_notFound} and** method **set to** GET Query Parameter attrs **set to {attrs}** **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 404 (Not Found) **and** Response Body **containing** ProblemDetails element **containing** type element **set to ${problem_type} and** title element **containing** more information about the error **}**	SUT → Client

Permutation on TP Id	${entity}	${entityId_notFound}	${attrs}	${problem_type}
TP/NGSI-LD/CI/Cons/E/018_03_01	Simple properties	Random valid id that does not correspond to any entity	empty	https://uri.etsi.org/ngsi-ld/errors/ResourceNotFound
TP/NGSI-LD/CI/Cons/E/018_03_02	Simple properties	${entityId}	"Property_Not Found"	https://uri.etsi.org/ngsi-ld/errors/ResourceNotFound

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 representation	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 6.3.7	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_	
Initial conditions	**with {** **the SUT containing** an initial Entity **${entity}** with an id **set to ${entityId}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Get Entity Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entities/**${entityId} and** method **set to** GET **and** options **set to** "keyValues"	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 200 (OK) **and** Response Body **containing** ${entity_simplified} **}**	SUT → Client

TP Id	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 V1.3.1 [1], clause 6.3.7
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_6_3_7
Initial conditions	**with {** **the SUT containing** an initial Entity **${entity}** with an id **set to ${entityId}** **}**

Expected behaviour	Test events
	when { **the SUT receives** a valid Get Entity Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entities/**${entityId} and** method **set to** GET **and** options: "keyValues" **and** Accept-Header **set to** "application/geo+json" }
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 200 (OK) **and** Response Body **containing** ${entity_simplified_geojson} **}**

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 V1.3.1 [1], clause 6.3.5	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_6_3_5	
Initial conditions	**with {** **the SUT containing** an initial Entity **${entity}** with an id **set to ${entityId}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Get Entity Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entities/**${entityId} and** method **set to** GET **and** Header: Link **set to** ${jsonld_context} **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 200 (OK) **and** Response Body **containing** ${entity_representation} **}**	SUT → Client

Permutation on TP Id	${jsonld_context}	${entity_representation}
TP/NGSI-LD/CI/Cons/E/018_06_01	empty	Entity with attributes from the context provided at creation time not compacted
TP/NGSI-LD/CI/Cons/E/018_06_02	Context containing the terms used at entity creation	Entity representation contains the selected attributes

4.1.2.1.2 Query Entities

TP Id	TP/NGSI-LD/CI/Cons/E/019_01
Test objective	Check that you can query several entities based on query parameters
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.7.2
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_5_7_2
Initial conditions	**with {** **the SUT containing** an initial group of Entities **}**
Expected behaviour	**Test events**
	when { **the SUT receives** a valid Get Entities Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entities **and** method **set to** GET **and** Query Parameter **${parameter}** set to **${value}** **}**
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 200 (OK) **and** Response Body **containing** a list **containing** Entity elements **containing** **${value}** provided **}**

Permutation on TP Id	${parameter}	${value}
TP/NGSI-LD/CI/Cons/E/019_01_01	id	List of entity ids to be retrieved
TP/NGSI-LD/CI/Cons/E/019_01_02	type	List of entity types to be retrieved
TP/NGSI-LD/CI/Cons/E/019_01_03	idPattern	Regular expression that shall be matched by entity ids
TP/NGSI-LD/CI/Cons/E/019_01_04	attrs	List of Attributes to be matched by the Entities and included in the response
TP/NGSI-LD/CI/Cons/E/019_01_05	geometryProperty	Which GeoProperty to use for the geoquery

TP Id	TP/NGSI-LD/CI/Cons/E/019_02
Test objective	Check that you can query several entities via POST Interaction
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.7.2
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_5_7_2
Initial conditions	**with {** **the SUT containing** an initial group of Entities **}**
Expected behaviour	**Test events**
	when { **the SUT receives** a valid Get Entities Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entityOperations/query **and** method **set to** POST **and** request body **set to** ${query} **containing ${parameter}** set to **${value}** **}**
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 200 (OK) **and** Response Body **containing** a list **containing** Entity elements **containing** **${value}** provided **}**

Permutation on TP Id	${parameter}	${value}
TP/NGSI-LD/CI/Cons/E/019_02_01	id	List of entity ids to be retrieved
TP/NGSI-LD/CI/Cons/E/019_02_02	type	List of entity types to be retrieved
TP/NGSI-LD/CI/Cons/E/019_02_03	idPattern	Regular expression that shall be matched by entity ids
TP/NGSI-LD/CI/Cons/E/019_02_04	attrs	List of Attributes to be matched by the Entities and included in the response
TP/NGSI-LD/CI/Cons/E/019_02_05	geometryProperty	Which GeoProperty to use for the geoquery

TP Id	TP/NGSI-LD/CI/Cons/E/019_03
Test objective	Check that you cannot query entities if the request is incorrect
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.7.2
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_5_7_2
Initial conditions	**with {** **the SUT containing** an initial group of Entities **}**

Expected behaviour	Test events	Direction
	when { **the SUT receives** an invalid Get Entities Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entities **and** method **set to** GET **and** Query Parameter **${parameter}** set to **${value}** **}**	SUT → Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request) **and** Response Body **containing** ProblemDetails element **containing** type element **set to ${problem_type} and** title element **containing** more information about the error **}**	SUT → Client

Permutation on TP Id	${parameter}	${value}
TP/NGSI-LD/CI/Cons/E/019_03_01	id	Invalid URI
TP/NGSI-LD/CI/Cons/E/019_03_02	type	Invalid type
TP/NGSI-LD/CI/Cons/E/019_03_03	idPattern	Invalid regex
TP/NGSI-LD/CI/Cons/E/019_03_04	attrs	Invalid list of attributes
TP/NGSI-LD/CI/Cons/E/019_03_05	geometryProperty	Invalid property

TP Id	TP/NGSI-LD/CI/Cons/E/019_04	
Test objective	Check that the queried entities can be returned in a simplified representation	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 6.3.7	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_6_3_7	
Initial conditions	**with {** **the SUT containing** at least 3 Entities ${entities} with an attribute ${attributeA} **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Get Entities Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entities **and** method **set to** GET **and**	SUT ← Client

	Query Parameter attr **set to** ${attributeA} **and** options **set to** "keyValues" **}**	
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 200 (OK) **and** Response Body **containing** ${entity_simplified} **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/Cons/E/019_05
Test objective	Check that the queried entities can be returned in a geoJSON format
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 6.3.7
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_6_3_7
Initial conditions	**with {** **the SUT containing** at least 3 Entities ${entities} with an attribute ${attributeA} **}**
Expected behaviour	**Test events**
	when { **the SUT receives** a valid Get Entities Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entities **and** method **set to** GET **and** Query Parameter attr **set to** ${attributeA} **and** Accept-Header **set to** "application/geo+json" **}**
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 200 (OK) **and** Response Body **containing** ${entities_simplified_geojson} **}**

TP Id	TP/NGSI-LD/CI/Cons/E/019_06
Test objective	Check that you can query entities specifying a maximum number of results
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 6.3.10

Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_6_3_10	
Initial conditions	**with {** **the SUT containing** at least 3 Entities ${entities} with an attribute ${attributeA} **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Get Entities Request **from** the client **containing** URL **set to** /ngsi-ld/v1/entities **and** method **set to** GET **and** Query Parameter attr **set to** ${attributeA} **and** Query Parameter limit **set to** 2 **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 200 (OK) **and** Response Body **containing** Two entities with ${attributeA} **}**	SUT → Client

4.1.2.2 Temporal Entity

4.1.2.2.1 Retrieve temporal evolution of Entity

TP Id	TP/NGSI-LD/CI/Cons/TE/020_01
Test objective	Check that you can retrieve the temporal evolution of an entity
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.7.3
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_5_7_3
Initial conditions	**with {** **the SUT containing** an initial temporal Entity with an id **set to ${entityId}** and temporal evolution of that Entity. **}**

Expected behaviour	Test events	Direction
	when { **the SUT receives** a valid Retrieve temporal evolution of an Entity Request **from** the client **containing** URL **set to** /ngsi-ld/v1/temporal/entities/**${entityId}** **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 200 (OK) **and** Response Body **containing** EntityTemporal element **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/Cons/TE/020_02	
Test objective	Check that you can retrieve the temporal evolution of an entity using a context	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.7.3	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_7_3	
Initial conditions	**with {** **the SUT containing** an initial temporal Entity with an id **set to ${entityId}** and temporal evolution of that Entity. **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Retrieve temporal evolution of an Entity Request **from** the client **containing** URL **set to** /ngsi-ld/v1/temporal/entities/**${entityId} and** Header: Link **set to** the context to be used for term to URI expansion/reduction and following the naming convention **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 200 (OK) **and** Response Body **containing** EntityTemporal element **containing** attribute names compacted with the context provided **}**	SUT → Client
TP Id	TP/NGSI-LD/CI/Cons/TE/020_03	

Test objective	Check that you can retrieve the temporal evolution of certain attributes of an entity	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.7.3	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_7_3	
Initial conditions	**with {** **the SUT containing** an initial temporal Entity with an id **set to ${entityId}** and temporal evolution of that Entity. **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Retrieve temporal evolution of an Entity Request **from** the client **containing** URL **set to** /ngsi-ld/v1/temporal/entities/**${entityId} and** Query Parameter: attrs **set to** the list of attributes to be retrieved **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 200 (OK) **and** Response Body **containing** EntityTemporal element containing attributes **in** the list of attributes provided **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/Cons/TE/020_04
Test objective	Check that you can retrieve the temporal evolution of an entity matching the given NGSI-LD temporal query
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.7.3
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_5_7_3
Initial conditions	**with {** **the SUT containing** an initial temporal Entity with an id **set to ${entityId}** and temporal evolution of that Entity **containing** temporal attributes **containing** attribute instances **between** 2020-06-01 **and** 2020-10-01. **}**

Expected behaviour	Test events	Direction
	when { **the SUT receives** a valid Retrieve temporal evolution of an Entity Request **from** the client **containing** URL **set to** /ngsi-ld/v1/temporal/entities/**${entityId} and** Query Parameter: timerel **set to ${timerel} and** Query Parameter: timeAt **set to ${timeAt} and** Query Parameter: endTimeAt **set to ${endTimeAt}** **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 200 (OK) **and** Response Body **containing** EntityTemporal element **containing** attribute instances **in the time range specified by the NGSI-LD temporal query** **}**	SUT → Client

Permutation on TP Id	${timerel}	${timeAt}	${endTimeAt}
TP/NGSI-LD/CI/Cons/TE/020_04_01	after	2020-08-01T13:03:00Z	Not present
TP/NGSI-LD/CI/Cons/TE/020_04_02	between	2020-08-01T12:00:00Z	2020-08-01T13:00:00Z
TP/NGSI-LD/CI/Cons/TE/020_04_03	before	2020-08-01T12:05:00Z	Not present

TP Id	TP/NGSI-LD/CI/Cons/TE/020_05	
Test objective	Check that you can retrieve the temporal evolution of the last N instances of entity attributes	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.7.3	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_7_3	
Initial conditions	**with {** **the SUT containing** an initial temporal Entity with an id **set to ${entityId}** and temporal evolution of that Entity **containing** temporal attributes **containing** each 15 instances. **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Retrieve temporal evolution of an Entity Request **from** the client **containing** URL **set to** /ngsi-ld/v1/temporal/entities/**${entityId} and** Query Parameter: lastN **set to ${lastN}** **}**	SUT ← Client

	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 200 (OK) **and** Response Body **containing** EntityTemporal element **containing** attributes **containing** each at most **${lastN}** instances **}**	SUT → Client

Permutation on TP Id	${lastN}
TP/NGSI-LD/CI/Cons/TE/020_05_01	10
TP/NGSI-LD/CI/Cons/TE/020_05_02	20

TP Id	TP/NGSI-LD/CI/Cons/TE/020_06	
Test objective	Check that you cannot retrieve the temporal evolution of an entity with an invalid id (invalid URI)	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.7.3	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_7_3	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Retrieve temporal evolution of an Entity Request **from** the client **containing** URL **set to** /ngsi-ld/v1/temporal/entities/**${invalid_entityId}** **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request) **and** Response Body **containing** ProblemDetails element **containing** type element **set to** https://uri.etsi.org/ngsi-ld/errors/BadRequestData **and** title element **containing** more information about the error **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/Cons/TE/020_07	
Test objective	Check that you cannot retrieve the temporal evolution of a non-existing entity	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.7.3	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_7_3	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Retrieve temporal evolution of an Entity Request **from** the client **containing** URL **set to** /ngsi-ld/v1/temporal/entities/urn:ngsi-ld:Vehicle:unknowEntity **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 404 (Not Found) **and** Response Body **containing** ProblemDetails element **containing** type element **set to** https://uri.etsi.org/ngsild/errors/ResourceNotFound **and** title element **containing** more information about the error **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/Cons/TE/020_08
Test objective	Check that you cannot retrieve the temporal evolution of non-existing entity attributes
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.7.3
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_5_7_3
Initial conditions	**with {** **the SUT containing** an initial temporal Entity with an id **set to ${entityId}** and temporal evolution of that Entity. **}**

Expected behaviour	Test events	Direction
	when { **the SUT receives** a valid Retrieve temporal evolution of an Entity Request **from** the client **containing** URL **set to** /ngsi-ld/v1/temporal/entities/**${entityId} and** Query Parameter: attrs **set to** a list of unknown attributes **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 404 (Not Found) **and** Response Body **containing** ProblemDetails element **containing** type element **set to** https://uri.etsi.org/ngsild/errors/ResourceNotFound **and** title element **containing** more information about the error **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/Cons/TE/020_09	
Test objective	Check that you cannot retrieve the temporal evolution of an entity with an invalid request content	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.7.3	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_7_3	
Initial conditions	**with {** **the SUT containing** an initial temporal Entity with an id **set to ${entityId}** and temporal evolution of that Entity. **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Retrieve temporal evolution of an Entity Request **from** the client **containing** URL **set to** /ngsi-ld/v1/temporal/entities/**${entityId} and** Query Parameter: timerel **set to ${timerel} and** Query Parameter: timeAt **set to ${timeAt} and** Query Parameter: endTimeAt **set to ${endTimeAt}** **}**	SUT ← Client

	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request) **and** Response Body **containing** ProblemDetails element **containing** type element **set to** https://uri.etsi.org/ngsi-ld/errors/BadRequestData **and** title element **containing** more information about the error **}**	SUT → Client

Permutation on TP Id	${timerel}	${timeAt}	${endTimeAt}
TP/NGSI-LD/CI/Cons/TE/020_09_01	after	Not present	Not present
TP/NGSI-LD/CI/Cons/TE/020_09_02	between	2020-08-01T12:00:00Z	Not present
TP/NGSI-LD/CI/Cons/TE/020_09_03	before	Not present	Not present

TP Id	TP/NGSI-LD/CI/Cons/TE/020_10	
Test objective	Check that you can retrieve the temporal evolution of an entity with the simplified temporal representation	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.7.3	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_7_3	
Initial conditions	**with {** **the SUT containing** an initial temporal Entity with an id **set to ${entityId}** and temporal evolution of that Entity. **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Retrieve temporal evolution of an Entity Request **from** the client **containing** URL **set to** /ngsi-ld/v1/temporal/entities/**${entityId}** Query Parameter: attrs **set to** temporalValues **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 200 (OK) **and** Response Body **containing** EntityTemporal element containing simplified temporal representation of attributes **}**	SUT → Client

4.1.2.2.2 Query temporal evolution of Entities

TP Id	TP/NGSI-LD/CI/Cons/TE/021_01	
Test objective	Check that you can query the temporal evolution of entities	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.7.3	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_7_4	
Initial conditions	**with {** **the SUT containing** an initial three temporal Entities and temporal evolution of those entities. **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Query temporal evolution of Entities Request **from** the client **containing** URL **set to** /ngsi-ld/v1/temporal/entities **and** Query Parameter: timerel **set to ${timerel} and** Query Parameter: timeAt **set to ${timeAt} and** Query Parameter: type **set to ${entity_types}** **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 200 (OK) **and** Response Body **containing** a list **containing** EntityTemporal elements **containing** entity type **in ${entity_types} and** attribute instances **${timerel} ${timeAt}** **}**	SUT → Client

Permutation on TP Id	${timerel}	${timeAt}	${entity_types}
TP/NGSI-LD/CI/Cons/TE/021_01_01	after	2020-08-01T12:00:00Z	List of expanded entity types to be retrieved
TP/NGSI-LD/CI/Cons/TE/021_01_02	before	2020-09-01T13:05:00Z	List of expanded entity types to be retrieved

TP Id	TP/NGSI-LD/CI/Cons/TE/021_02
Test objective	Check that you can query the temporal evolution of certain attributes of entities
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.7.3
Config Id	CF_01
Parent Release	V1.3.1

PICS Selection	PICS_5_7_4	
Initial conditions	**with {** **the SUT containing** an initial two temporal Entities and temporal evolution of those entities. **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Query temporal evolution of Entities Request **from** the client **containing** URL **set to** /ngsi-ld/v1/temporal/entities **and** Query Parameter: timerel **set to** after **and** Query Parameter: timeAt **set to** 2020-07-01T12:05:00Z **and** Query Parameter: attrs **set to** List of attributes to be retrieved **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 200 (OK) **and** Response Body **containing** a list **containing** EntityTemporal elements **containing** attributes **in** the list of attributes provided **and** attribute instances **after** 2020-07-01T12:05:00Z **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/Cons/TE/021_03	
Test objective	Check that you can query the temporal evolution of the last N instances of entities attributes	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.7.3	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_7_4	
Initial conditions	**with {** **the SUT containing** an initial two temporal Entities and temporal evolution of those entities. **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Query temporal evolution of Entities Request **from** the client **containing** URL **set to** /ngsi-ld/v1/temporal/entities **and** Query Parameter: timerel **set to** after **and**	SUT ← Client

	Query Parameter: timeAt **set to** 2020-07-01T12:05:00Z **and** Query Parameter: type **set to** List of entity types to be retrieved **and** Query Parameter: lastN **set to** 14 **}**	
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 200 (OK) **and** Response Body **containing** a list **containing** EntityTemporal elements **containing** entity type **in** the list of entity types provided **and** attributes **containing** each last 14 instances **after** 2020-07-01T12:05:00Z **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/Cons/TE/021_04	
Test objective	Check that you can query the temporal evolution of entities using a context	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.7.3	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_7_4	
Initial conditions	**with {** **the SUT containing** an initial two temporal Entities and temporal evolution of those entities. **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Query temporal evolution of Entities Request **from** the client **containing** URL **set to** /ngsi-ld/v1/temporal/entities **and** Header: Link **set to** the context to be used for term to URI expansion/reduction and following the naming convention **and** Query Parameter: timerel **set to** after **and** Query Parameter: timeAt **set to** 2020-07-01T12:05:00Z **and** Query Parameter: type **set to** List of entity types to be retrieved **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 200 (OK) **and** Response Body **containing** a list **containing**	SUT → Client

	EntityTemporal elements **containing** entity type **in** the list of entity types provided **and** attribute instances **after** 2020-07-01T12:05:00Z **and** attribute names compacted with the context provided **}**	

TP Id	TP/NGSI-LD/CI/Cons/TE/021_05	
Test objective	Check that you can query the temporal evolution of entities matching the given type(s)	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.7.3	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_7_4	
Initial conditions	**with {** **the SUT containing** an initial two temporal Entities and temporal evolution of those entities. **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Query temporal evolution of Entities Request **from** the client **containing** URL **set to** /ngsi-ld/v1/temporal/entities **and** Query Parameter: timerel **set to** after **and** Query Parameter: timeAt **set to** 2020-07-01T12:05:00Z **and** Query Parameter: type **set to** List of entity types to be retrieved **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 200 (OK) **and** Response Body **containing** a list **containing** EntityTemporal elements **containing** entity type **in** the list of entity types provided **and** attribute instances **after** 2020-07-01T12:05:00Z **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/Cons/TE/021_06
Test objective	Check that you can query the temporal evolution of entities matching the given identifier(s)
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.7.3

Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_7_4	
Initial conditions	**with {** **the SUT containing** an initial two temporal Entities and temporal evolution of those entities. **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Query temporal evolution of Entities Request **from** the client **containing** URL **set to** /ngsi-ld/v1/temporal/entities **and** Query Parameter: timerel **set to** after **and** Query Parameter: timeAt **set to** 2020-07-01T12:05:00Z **and** Query Parameter: type **set to** List of entity types to be retrieved **and** Query Parameter: id **set to** List of entity ids to be retrieved **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 200 (OK) **and** Response Body **containing** a list **containing** EntityTemporal elements **containing** entity type **in** the list of entity types provided **and** entity id **in** the list of entity ids provided **and** attribute instances **after** 2020-07-01T12:05:00Z **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/Cons/TE/021_07	
Test objective	Check that you can query the temporal evolution of entities matching the given id pattern	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.7.3	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_7_4	
Initial conditions	**with {** **the SUT containing** an initial two temporal Entities and temporal evolution of those entities. **}**	
Expected behaviour	**Test events**	**Direction**
	when {	SUT ← Client

	the SUT receives a valid Query temporal evolution of Entities Request **from** the client **containing** URL **set to** /ngsi-ld/v1/temporal/entities **and** Query Parameter: timerel **set to** after **and** Query Parameter: timeAt **set to** 2020-07-01T12:05:00Z **and** Query Parameter: type **set to** List of entity types to be retrieved **and** Query Parameter: idPattern **set to** a regular expression that shall be matched by entity ids **}**	
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 200 (OK) **and** Response Body **containing** a list **containing** EntityTemporal elements **containing** entity type **in** the list of entity types provided **and** entity id matching id pattern provided **and** attribute instances **after** 2020-07-01T12:05:00Z **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/Cons/TE/021_08	
Test objective	Check that you can query the temporal evolution of entities matching the given NGSI-LD query	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.7.3	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_7_4	
Initial conditions	**with {** **the SUT containing** an initial two temporal Entities and temporal evolution of those entities. **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Query temporal evolution of Entities Request **from** the client **containing** URL **set to** /ngsi-ld/v1/temporal/entities **and** Query Parameter: timerel **set to** after **and** Query Parameter: timeAt **set to** 2020-07-01T12:05:00Z **and** Query Parameter: type **set to** List of entity types to be retrieved **and** Query Parameter: q **set to** an NGSI-LD query	SUT ← Client

	}	
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 200 (OK) **and** Response Body **containing** a list **containing** EntityTemporal elements **containing** entity type **in** the list of entity types provided **and** attribute instances **after** 2020-07-01T12:05:00Z **and** meet the matching conditions specified by the query **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/Cons/TE/021_09	
Test objective	Check that you can query the temporal evolution of entities matching the given NGSI-LD geo-query	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.7.3	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_7_4	
Initial conditions	**with {** **the SUT containing** an initial two temporal Entities and temporal evolution of those entities. **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Query temporal evolution of Entities Request **from** the client **containing** URL **set to** /ngsi-ld/v1/temporal/entities **and** Query Parameter: timerel **set to** after **and** Query Parameter: timeAt **set to** 2020-07-01T12:05:00Z **and** Query Parameter: type **set to** List of entity types to be retrieved **and** Query Parameter: georel **set to ${georel} and** Query Parameter: geometry **set to ${geometry} and** Query Parameter: coordinates **set to ${coordinates} and** Query Parameter: geoproperty **set to ${geoproperty} and** **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 200 (OK) **and**	SUT → Client

	Response Body **containing** a list **containing** EntityTemporal elements **containing** entity type **in** the list of entity types provided **and** attribute instances **after** 2020-07-01T12:05:00Z **and** GeoProperty instances that meet the geospatial restrictions imposed by the geo-query **}**	

Permutation on TP Id	${georel}	${geometry}	${coordinates}	${geoproperty}
TP/NGSI-LD/CI/Cons/TE/021_09_01	near;maxDistance==2000	Point	[-8.503,41.202]	Not present
TP/NGSI-LD/CI/Cons/TE/021_09_02	within	Polygon	[[-13.503,47.202], [6.541, 52.961], [20.37,44.653], [9.46,32.57], [-15.23,21.37]]	location

TP Id	TP/NGSI-LD/CI/Cons/TE/021_10	
Test objective	Check that you can query the temporal evolution of entities matching the given NGSI-LD Context Source filter	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.7.3	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_7_4	
Initial conditions	**with {** **the SUT containing** a context source registration **of** a context source (CS1) providing temporal information **of** two entities **of** type Building between 2020-08-01T22:07:00Z **and** 2021-08-01T21:07:00Z **and** CS1 **containing** two temporal entities **of** type Building **and** temporal evolution **of** those entities in **the** mentioned interval. **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Query temporal evolution of Entities Request **from** the client **containing** URL **set to** /ngsi-ld/v1/temporal/entities **and** Query Parameter: timerel **set to** after **and** Query Parameter: timeAt **set to** 2020-07-01T12:05:00Z **and** Query Parameter: type **set to** Building **and** Query Parameter: csf **set to** a context Source filter matching CS1 **}**	SUT ← Client

	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 200 (OK) **and** Response Body **containing** a list **containing** two EntityTemporal elements **from** the context sources discovered with the csf provided **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/Cons/TE/021_11	
Test objective	Check that you can query the temporal evolution of entities with a limit to the number of entities to be retrieved	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.7.3	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_7_4	
Initial conditions	**with {** **the SUT containing** an initial three temporal Entities and temporal evolution of those entities. **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Query temporal evolution of Entities Request **from** the client **containing** URL **set to** /ngsi-ld/v1/temporal/entities **and** Query Parameter: timerel **set to** after **and** Query Parameter: timeAt **set to** 2020-07-01T12:05:00Z **and** Query Parameter: type **set to** List of entity types to be retrieved **and** Query Parameter: limit **set to ${limit}** **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 200 (OK) **and** Response Body **containing** a list **containing** at most **${limit}** EntityTemporal elements **containing** entity type **in** the list of entity types provided **and** attribute instances **after** 2020-07-01T12:05:00Z **}**	SUT → Client

Permutation on TP Id	**${limit}**
TP/NGSI-LD/CI/Cons/TE/021_11_01	2
TP/NGSI-LD/CI/Cons/TE/021_11_02	20

TP Id	TP/NGSI-LD/CI/Cons/TE/021_12	
Test objective	Check that you cannot query the temporal evolution of entities with an invalid request or invalid request content	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.7.3	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_7_4	
Initial conditions	**with {** **the SUT containing** an initial two temporal Entities and temporal evolution of those entities. **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Query temporal evolution of Entities Request **from** the client **containing** URL **set to** /ngsi-ld/v1/temporal/entities **and** Query Parameter: timerel **set to** after **and** Query Parameter: timeAt **set to** 2020-07-01T12:05:00Z **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request) **and** Response Body **containing** ProblemDetails element **containing** type element **set to** https://uri.etsi.org/ngsi-ld/errors/BadRequestData **and** title element **containing** more information about the error **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/Cons/TE/021_13
Test objective	Check that you can query the temporal evolution of entities using the entityOperations method
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.7.3
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_5_7_4
Initial conditions	**with {** **the SUT containing** an initial two temporal Entities and temporal evolution of those entities. **}**

Expected behaviour	Test events	Direction
	when { **the SUT receives** a valid Entity Operations Request to Query temporal evolution of Entities **from** the client **containing** URL **set to** /ngsi-ld/v1/temporal/entityOperations/query **and** Body **containing** a NGSI-LD temporal query **containing ${timerel}** and **${timeAt}** attributes **and** an entity type **set to ${entity_types}** **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 200 (OK) **and** Response Body **containing** a list **containing** EntityTemporal elements **containing** entity type **in ${entity_types} and** attribute instances **${timerel} ${timeAt}** **}**	SUT → Client

Permutation on TP Id	${timerel}	${timeAt}	${entity_types}
TP/NGSI-LD/CI/Cons/TE/021_13_01	after	2020-08-02T12:05:00Z	List of expanded entity types to be retrieved
TP/NGSI-LD/CI/Cons/TE/021_13_02	before	2020-08-02T12:05:00Z	List of expanded entity types to be retrieved

4.1.2.3 Discovery

4.1.2.3.1 Retrieve Available Entity Types

TP Id	TP/NGSI-LD/CI/CONS/DISC/022_01	
Test objective	Check that you can retrieve a list of NGSI-LD entity types	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.7.5	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_7_5	
Initial conditions	**with {** **the SUT containing** an initial state **}**	
Expected behaviour	Test events	Direction
	when {	SUT ← Client

	the SUT receives a valid Entity type query **from** the client **containing** URL **set to** /ngsi-ld/v1/entityTypeList and **${context}**	
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 200 (OK) **and** Response Body **containing ${EntityTypeList}** **}**	SUT → Client

Permutation on TP Id	${context}	${EntityTypeList} }
TP/NGSI-LD/CI/CONS/DISC/022_01_01		Json object with list of entity types
TP/NGSI-LD/CI/CONS/DISC/022_01_02	Json-ld context	Json object with list of entity types

4.1.2.3.2 Retrieve Details of Available Entity Types

TP Id	TP/NGSI-LD/CI/CONS/DISC/023_01	
Test objective	Check that you can retrieve a list with a detailed representation of NGSI-LD entity types	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.7.6	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_7_6	
Initial conditions	**with {** **the SUT containing** an initial state **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Entity type query **from** the client **containing** URL **set to** /ngsi-ld/v1/entityType and **${context}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 200 (OK) **and** Response Body **containing** a list **containing** All available entity types **}**	SUT → Client

Permutation on TP Id	${context}
23_01_01	No context
23_01_02	Json-ld context

4.1.2.3.3 Retrieve Available Entity Type Information

TP Id	TP/NGSI-LD/CI/CONS/DISC/024_01	
Test objective	Check that you can retrieve a detailed representation of a specified NGSI-LD entity type	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.7.7	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_7_7	
Initial conditions	**with {** **the SUT containing** an initial state **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Entity type query **from** the client **containing** URL **set to** /ngsi-ld/v1/entityType?type=${typeName} and **${context}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 200 (OK) **and** Response Body **containing** EntityTypeInfo **}**	SUT → Client

Permutation on TP Id	${context}
24_01_01	No Context
24_01_02	Json-ld context

4.1.2.3.4 Retrieve Available Entity Type Information

TP Id	TP/NGSI-LD/CI/CONS/DISC/025_0
Test objective	Check that you can retrieve a list of NGSI-LD attributes
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.7.8
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_5_7_8
Initial conditions	**with {** **the SUT containing** an initial state **}**

Expected behaviour	Test events	Direction
	when { **the SUT receives** a valid Entity type query **from** the client **containing** URL **set to** /ngsi-ld/v1/AttributeList and **${context}** **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 200 (OK) **and** Response Body **containing** attributeList **}**	SUT → Client

Permutation on TP Id	${context}
25_01_01	No context
25_01_02	Json-ld context

4.1.2.3.5 Retrieve Details of Available Attributes

TP Id	TP/NGSI-LD/CI/CONS/DISC/026_01	
Test objective	Check that you can retrieve a list with a detailed representation of NGSI-LD attributes	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.7.9	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_7_9	
Initial conditions	**with {** **the SUT containing** an initial state **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Entity type query **from** the client **containing** URL **set to** /ngsi-ld/v1/Attribute and **${context}** **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 200 (OK) **and** Response Body **containing** an array of Attributes **}**	SUT → Client

Permutation on TP Id	${context}
26_01_01	No Context
26_01_02	Json-ld context

4.1.2.3.6 Retrieve Details of Available Attributes

TP Id	TP/NGSI-LD/CI/CONS/DISC/027_01	
Test objective	Check that you can retrieve a list with a detailed representation of NGSI-LD attributes	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.7.10	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_7_10	
Initial conditions	**with {** **the SUT containing** an initial state **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Entity type query **from** the client **containing** URL **set to** /ngsi-ld/v1/Attribute?attributeName=${attributeName} and **${context}** **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 200 (OK) **and** Response Body **containing** an attribute **}**	SUT → Client

Permutation on TP Id	${context}
27_01_01	No Context
27_01_02	Json-ld context

4.1.3 Subscription

4.1.3.1 Create Subscription

TP Id	TP/NGSI-LD/CI/SUB/028_01
Test objective	Check that you cannot create a subscription: if data types, cardinalities and restrictions are not met, then an error of type BadRequestData is be raised.
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.8.1
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_5_8_1

Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Subscription Request **from** the client **containing** URL **set to** /ngsi-ld/v1/subscriptions **and** method **set to** POST **and** Header: Content-Type **set to** Application/ld+json **and** body **set to {invalid_subscription}** **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request) **and** Response Body **containing** ProblemDetails element **containing** type element **set to ${problem_type} and** title element **containing** more information about the error **}**	SUT → Client

Permutation on TP Id	${invalid_subscription}	${problem_type}
28_01_01	Subscription containing invalid data type	https://uri.etsi.org/ngsi-ld/errors/BadRequestData
28_01_02	Subscription containing invalid cardinality	https://uri.etsi.org/ngsi-ld/errors/BadRequestData
28_01_03	Subscription containing invalid restriction	https://uri.etsi.org/ngsi-ld/errors/BadRequestData

TP Id	TP/NGSI-LD/CI/SUB/028_02	
Test objective	Check that you cannot create a subscription: if the NGSI-LD endpoint already knows about this Subscription, as there is an existing Subscription whose id (URI) is equivalent, then an error of type AlreadyExists is be raised.	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.8.1	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_8_1	
Initial conditions	**with {** **the SUT being** in the "initial state", the SUT has a Subscription ${subscription) created with the id **set to** ${subscriptionId} **}**	
	Test events	**Direction**

Expected behaviour	**when {** **the SUT receives** a valid Create Subscription Request **from** the client **containing** URL **set to** /ngsi-ld/v1/subscriptions **and** method **set to** POST **and** Header: Content-Type **set to** Application/ld+json **and** body **set to** ${subscription) and the id **set to** ${subscriptionId} **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 409 (Already Exists) **and** Response Body **containing** ProblemDetails element **containing** type element **set to** https://uri.etsi.org/ngsi-ld/errors/AlreadyExists **and** title element **containing** more information about the error **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/SUB/028_03	
Test objective	Check that you can create a subscription: if the subscription document does not include a Subscription identifier, then a new identifier (URI) is be automatically generated by the implementation.	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.8.1	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_8_1	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Subscription Request **from** the client **containing** URL **set to** /ngsi-ld/v1/subscriptions **and** method **set to** POST **and** Header: Content-Type **set to** Application/ld+json **and** body **containing** the subscription to be created **not containing** an id **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 201 (Created) **and**	SUT → Client

	Location Header **containing** the URI of the created {subscription} **}**	

TP Id	TP/NGSI-LD/CI/SUB/028_04	
Test objective	Check that you can create a subscription: the subscription expiration date is equal to the value of the expiresAt member.	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.8.1	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_8_1	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Subscription Request **from** the client **containing** URL **set to** /ngsi-ld/v1/subscriptions **and** method **set to** POST **and** Header: Content-Type **set to** Application/ld+json **and** body **set to** subscription to be created **and** expiration date **set to** expiresAt **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Status Code **set to** 201 (CREATED) **and** Body **set to set to** Subscription **containing** expiration date **set to** the value of expiresAt **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/SUB/028_05
Test objective	Check that you cannot create a subscription: if the expiration timestamp provided represents a moment before the current date and time, then an error of type BadRequestData is raised
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.8.1
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_5_8_1

Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Subscription Request **from** the client **containing** URL **set to** /ngsi-ld/v1/subscriptions **and** method **set to** POST **and** Header: Content-Type **set to** Application/ld+json **and** body **set to s**ubscription to be created **containing** expiration timestamp **set to** a moment before the current date and time **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request) **and** Response Body **containing** ProblemDetails element **containing** type element **set to** https://uri.etsi.org/ngsi-ld/errors/BadRequestData **and** title element **containing** more information about the error **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/SUB/028_06	
Test objective	Check that you can create a subscription: If there is no expiresAt member, then the Subscription is considered as perpetual.	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.8.1	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_8_1	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Subscription Request **from** the client **containing** URL **set to** /ngsi-ld/v1/subscriptions **and** method **set to** POST **and** Header: Content-Type **set to** Application/ld+json **and**	SUT ← Client

	body **containing** the subscription to be created **not containing** expiresAt member **}**	
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 201 (CREATED) **and** Location header **set to** the resource URI of the subscription created **and** the subscription created **not containing** expiresAt member **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/SUB/028_07	
Test objective	Check that you can create a subscription: If the value of the isActive field is not included or is true then the initial status of the Subscription is set to "active"	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.8.1	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_8_1	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Subscription Request **from** the client **containing** URL **set to** /ngsi-ld/v1/subscriptions **and** method **set to** POST **and** Header: Content-Type **set to** Application/ld+json **and** body **set to** Subscription to be created **containing** isActive **set to** {value} **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Status Code **set to** 201 (CREATED) **and** Body **set to** Subscription **containing** Status **set to** active **}**	SUT → Client

Permutation on TP Id	${value}
TP/NGSI-LD/CI/SUB/028_07_01	not included
TP/NGSI-LD/CI/SUB/028_07_02	true

TP Id	TP/NGSI-LD/CI/SUB/028_08

Test objective	Check that you can create a subscription: If the value of the isActive field is false, then the initial status of the Subscription is set to "paused".	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.8.1	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_8_1	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Subscription Request **from** the client **containing** URL **set to** /ngsi-ld/v1/subscriptions **and** method **set to** POST **and** Header: Content-Type **set to** Application/ld+json **and** body **set to** Subscription to be created **containing** isActive **set to** false **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Status Code **set to** 201 (CREATED) **and** Body **set to** Subscription **containing** Status **set to** paused **}**	SUT → Client

4.1.3.2 Update Subscription

TP Id	TP/NGSI-LD/CI/SUB/029_01
Test objective	Check that you cannot update a subscription: if the Subscription id is not present or it is not a valid URI, then an error of type BadRequestData shall be raised.
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.8.2
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_5_8_2
Initial conditions	**with {** **the SUT being** in the "initial state" **}**

Expected behaviour	Test events	Direction
	when { **the SUT receives** an Update Subscription Request **from** the client **containing** URL **set to** /ngsi-ld/v1/subscriptions/${subscriptionId} **and** method **set to** PATCH **and** Header: Content-Type **set to** application/ld+json **and** body **set to** a subscription update fragment **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request) **and** Response Body **containing** ProblemDetails element **containing** type element **set to ${problem_type} and** title element **containing** more information about the error **}**	SUT → Client

Permutation on TP Id	${subscriptionId}
TP/NGSI-LD/CI/SUB/029_01_01	null
TP/NGSI-LD/CI/SUB/029_01_02	Not a valid URI

TP Id	TP/NGSI-LD/CI/SUB/029_02	
Test objective	Check that you cannot update a subscription: If the NGSI-LD System does not know about the target Subscription, because there is no existing Subscription whose id (URI) is equivalent, an error of type ResourceNotFound shall be raised.	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.8.2	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_8_2	
Initial conditions	**with {** **the SUT not containing** a subscription with id **set to ${subscriptionId}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** an Update Subscription Request **from** the client **containing** URL **set to** /ngsi-ld/v1/subscriptions/**${subscriptionId} and** method **set to** PATCH **and** Header: Content-Type **set to** application/ld+json **and** body **set to** a subscription update fragment	SUT ← Client

	}	
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 404 (ResourceNotFound) **and** Response Body **containing** ProblemDetails element **containing** type element **set to ${problem_type} and** title element **containing** more information about the error **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/SUB/029_03	
Test objective	Check that you cannot update a subscription: If the data types and restriction are not met by the Subscription Fragment, then an error of type BadRequestData shall be raised	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.8.2	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_8_2	
Initial conditions	**with {** **the SUT containing** a subscription with id **set to ${subscriptionId}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** an Update Subscription Request **from** the client **containing** URL **set to** /ngsi-ld/v1/subscriptions/**${subscriptionId} and** method **set to** PATCH **and** Header: Content-Type **set to** application/ld+json **and** body **set to** a subscription update fragment **containing** invalid data types and restrictions **}**	SUT ← Client

	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request) **and** Response Body **containing** ProblemDetails element **containing** type element **set to ${problem_type} and** title element **containing** more information about the error **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/SUB/029_04	
Test objective	Check that you cannot update a subscription: Any attempt to remove (by setting them to null in the Fragment) mandatory properties of a Subscription (clause 5.2.12) shall result in an error of type BadRequestData	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.8.2	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_8_2	
Initial conditions	**with {** **the SUT containing** a subscription with id **set to ${subscriptionId}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** an Update Subscription Request **from** the client **containing** URL **set to** /ngsi-ld/v1/subscriptions/**${subscriptionId} and** method **set to** PATCH **and** Header: Content-Type **set to** application/ld+json **and** body **set to** a subscription update fragment **containing** mandatory property **set to** null **}**	SUT ← Client

	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request) **and** Response Body **containing** ProblemDetails element **containing** type element **set to ${problem_type} and** title element **containing** more information about the error **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/SUB/029_05	
Test objective	Check that you can update a subscription: Term to URI expansion of Attribute names shall be observed.	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.8.2	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_8_2	
Initial conditions	**with {** **the SUT containing** a subscription with id **set to ${subscriptionId} and** matching entities of type T **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** an Update Subscription Request **from** the client **containing** URL **set to** /ngsi-ld/v1/subscriptions/**${subscriptionId} and** method **set to** PATCH **and** Header: Content-Type **set to** application/ld+json **and** body **set to ${update_fragment} containing** entities member **and** @context member providing new expanded format for the entity type T	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 204 (No Content) **and** updated resource **set to the** subscription updated **with ${update_fragment} when** retrieved **with the** new provided @context **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/SUB/029_06
Test objective	Check that you can update a subscription: The implementation shall modify the target Subscription.
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.8.2

Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_8_2	
Initial conditions	**with {** **the SUT containing** a subscription with id **set to ${subscriptionId}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** an Update Subscription Request **from** the client **containing** URL **set to** /ngsi-ld/v1/subscriptions/**${subscriptionId} and** method **set to** PATCH **and** Header: Content-Type **set to** application/ld+json **and** body **set to** subscription update fragment **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 204 (No Content) **and** updated resource **set to the** subscription updated **with ${update_fragment}** **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/SUB/029_07	
Test objective	Check that you can update a subscription: If isActive is equal to true or null and expiresAt is not present, then status shall be updated to "active", if and only if, the previous value of status was different than "expired".	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.8.2	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_8_2	
Initial conditions	**with {** **the SUT containing** a subscription with id **set to ${subscriptionId} and** status member different than "expired". **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** an Update Subscription Request **from** the client **containing** URL **set to** /ngsi-ld/v1/subscriptions/**${subscriptionId} and** method **set to** PATCH **and** Header: Content-Type **set to** application/ld+json **and**	SUT ← Client

	body **set to** subscription update fragment **containing** isActive **set to ${isActive} and** **not containing** expiresAt member **}**	
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 204 (No Content) **and** the status of **${subscriptionId}** is **set to** "active" **}**	SUT → Client

Permutation on TP Id	${isActive}
TP/NGSI-LD/CI/SUB/029_07_01	true
TP/NGSI-LD/CI/SUB/029_07_02	null

TP Id	TP/NGSI-LD/CI/SUB/029_08	
Test objective	Check that you can update a subscription: If isActive is equal to true or null and expiresAt is null or corresponds to a DateTime in the future, then status shall be updated to "active".	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.8.2	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_8_2	
Initial conditions	**with {** **the SUT containing** a subscription with id **set to ${subscriptionId}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** an Update Subscription Request **from** the client **containing** URL **set to** /ngsi-ld/v1/subscriptions/**${subscriptionId} and** method **set to** PATCH **and** Header: Content-Type **set to** application/ld+json **and** body **set to** subscription update fragment **containing** isActive **set to ${isActive} and** expiresAt **set to ${expiresAt}** **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 204 (No Content) **and** the status of **${subscriptionId}** is **set to** "active" **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/SUB/029_09

Test objective	Check that you can update a subscription: If isActive is equal to false and expiresAt is not present, then status shall be updated to "paused", if and only if, the previous value of status was different than "expired".	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.8.2	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_8_2	
Initial conditions	**with {** **the SUT containing** a subscription with id **set to ${subscriptionId} and** status member different than "expired" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** an Update Subscription Request **from** the client **containing** URL **set to** /ngsi-ld/v1/subscriptions/**${subscriptionId} and** method **set to** PATCH **and** Header: Content-Type **set to** application/ld+json **and** body **set to** subscription update fragment **containing** isActive member **set to** false **and** **not containing** expiresAt member **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 204 (No Content) **and** the status of **${subscriptionId}** is **set to** "paused" **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/SUB/029_10	
Test objective	Check that you can update a subscription: If only expiresAt is included and refers to a DateTime in the future or is null, then status shall be updated to "active", if and only if the previous value of status was "expired".	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.8.2	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_8_2	
Initial conditions	**with {** **the SUT containing** a subscription with id **set to ${subscriptionId} and** status member **set to** "expired". **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** an Update Subscription Request **from** the client **containing**	SUT ← Client

	URL **set to** /ngsi-ld/v1/subscriptions/**${subscriptionId} and** method **set to** PATCH **and** Header: Content-Type **set to** application/ld+json **and** body **set to** subscription update fragment **containing** expiresAt member **set to** ${expiresAt} **}**	
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 204 (No Content) **and** the status of **${subscriptionId}** is **set to** "active" **}**	SUT → Client

Permutation on TP Id	${expiresAt}
TP/NGSI-LD/CI/SUB/029_10_01	refers to a DateTime in the future
TP/NGSI-LD/CI/SUB/020_10_02	null

TP Id	TP/NGSI-LD/CI/SUB/029_11	
Test objective	Check that you cannot update a subscription: If expiresAt is included but referring to a DateTime in the past, then a BadRequestData error shall be raised	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.8.2	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_8_2	
Initial conditions	**with {** **the SUT containing** a subscription with id **set to ${subscriptionId}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** an Update Subscription Request **from** the client **containing** URL **set to** /ngsi-ld/v1/subscriptions/**${subscriptionId} and** method **set to** PATCH **and** Header: Content-Type **set to** application/ld+json **and** body **set to** subscription update fragment **containing** expiresAt member **set to** a date referring to a DateTime in the past **}**	SUT ← Client
	then { **the SUT sends** a Response **containing** Response Status Code **set to** 400 (Bad Request) **and** Response Body **containing** ProblemDetails element **containing** type element **set to ${problem_type} and**	SUT → Client

	title element **containing** more information about the error **}**	

4.1.3.3 Retrieve Subscription

TP Id	TP/NGSI-LD/CI/SUB/030_01	
Test objective	Check that you cannot retrieve a subscription: If the subscription Id is not present or it is not a valid URI, then an error of type BadRequestData shall be raised.	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.8.3	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_8_3	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** an Subscription Retrieve a Request **from** the client **containing** URL **set to** /ngsi-ld/v1/subscriptions/**${subscriptionId} and** method **set to** GET **}**	SUT ← Client
	then { **the SUT sends** a Response **containing** Response Status Code **set to** 400 (Bad Request) **and** Response Body **containing** ProblemDetails element **containing** type element **set to ${problem_type} and** title element **containing** more information about the error **}**	SUT → Client

Permutation on TP Id	${subscriptionId}
TP/NGSI-LD/CI/SUB/030_01_01	Not present
TP/NGSI-LD/CI/SUB/030_01_02	Not a valid URI

TP Id	TP/NGSI-LD/CI/SUB/030_02
Test objective	Check that you cannot retrieve a subscription: If the identifier provided does not correspond to any existing subscription in the system then an error of type ResourceNotFound shall be raised.
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.8.3

Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_8_3	
Initial conditions	**with {** **the SUT not containing** a subscription with id **set to ${subscriptionId}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** an Subscription Retrieve a Request **from** the client **containing** URL **set to** /ngsi-ld/v1/subscriptions/**${subscriptionId} and** method **set to** GET **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 404 (ResourceNotFound) **and** Response Body **containing** ProblemDetails element **containing** type element **set to ${problem_type} and** title element **containing** more information about the error **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/SUB/030_03	
Test objective	Check that you can retrieve a subscription	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.8.3	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_8_3	
Initial conditions	**with {** **the SUT containing** a subscription with id **set to ${subscriptionId}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** an Subscription Retrieve a Request **from** the client **containing** URL **set to** /ngsi-ld/v1/subscriptions/**${subscriptionId} and** method **set to** GET **}**	SUT ← Client

	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 200 (Ok) **and** Response body **set to** Subscription **}**	SUT → Client

4.1.3.4 Query Subscriptions

TP Id	TP/NGSI-LD/CI/SUB/031_01	
Test objective	Check that you can query a list of subscriptions	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.8.4	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_8_4	
Initial conditions	**with {** **the SUT containing** three Subscriptions **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Query Subscriptions Request **from** the client **containing** URL **set to** /ngsi-ld/v1/subscriptions **and** method **set to** GET **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** Ok **and** **body set to** list **containing** three subscriptions **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/SUB/031_02
Test objective	Check that you can query a list of subscriptions: Pagination logic shall be in place
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.8.4
Config Id	CF_05
Parent Release	V1.3.1
PICS Selection	PICS_5_8_4
Initial conditions	**with {** **the SUT containing** three Subscriptions **}**

Expected behaviour	Test events	Direction
	when { **the SUT receives** a valid Query Subscriptions Request **from** the client **containing** URL **set to** /ngsi-ld/v1/subscriptions **and** method **set to** GET **and** Query Parameter limit **set to ${limit} and** Query Parameter page **set to ${page}** **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 200 (Ok) **and** Response Body **containing** a list **of** subscriptions respecting the pagination logic **}**	SUT → Client

Permutation on TP Id	${limit}	${page}
TP/NGSI-LD/CS/REGSUB/031_02_01	1	2
TP/NGSI-LD/CS/REGSUB/031_02_02	2	2
TP/NGSI-LD/CS/REGSUB/031_02_03	15	1

4.1.3.5 Delete Subscription

TP Id	TP/NGSI-LD/CI/SUB/032_01	
Test objective	Check that you cannot delete a subscription: If the subscription Id is not present or it is not a valid URI, then an error of type BadRequestData shall be raised.	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.8.5	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_8_5	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** an Delete Subscription Request **from** the client **containing** URL **set to** /ngsi-ld/v1/subscriptions/**${subscriptionId} and** method **set to** Delete **}**	SUT ← Client
	then { **the SUT sends** a Response **containing** Response Status Code **set to** 400 (Bad Request) **and**	SUT → Client

	Response Body **containing** ProblemDetails element **containing** type element **set to ${problem_type} and** title element **containing** more information about the error **}**	

Permutation on TP Id	${subscriptionId}
TP/NGSI-LD/CI/SUB/032_01_01	Not present
TP/NGSI-LD/CI/SUB/032_01_02	Not a valid URI

TP Id	TP/NGSI-LD/CI/SUB/032_02	
Test objective	Check that you cannot delete a subscription: If the subscription id provided does not correspond to any existing subscription in the system then an error of type ResourceNotFound shall be raised.	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.8.5	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_8_5	
Initial conditions	**with {** **the SUT not containing** a subscription with id **set to ${subscriptionId}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** an Delete Subscription Request **from** the client **containing** URL **set to** /ngsi-ld/v1/subscriptions/**${subscriptionId} and** method **set to** Delete }	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 404 (ResourceNotFound) **and** Response Body **containing** ProblemDetails element **containing** type element **set to ${problem_type} and** title element **containing** more information about the error **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/SUB/032_03
Test objective	Check that you can delete a subscription

Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.8.5	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_8_5	
Initial conditions	**with {** **the SUT containing** a subscription with id **set to ${subscriptionId}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** an Delete Subscription Request **from** the client **containing** URL **set to** /ngsi-ld/v1/subscriptions/**${subscriptionId} and** method **set to** Delete **}**	SUT ← Client
	then { **the SUT sends** a Response **containing** Response Status Code **set to** 204 (No Content) **and the SUT not containing** resource **with** id **set to ${subscriptionId}** **}**	SUT → Client

4.1.3.6 Notification Behaviour

TP Id	TP/NGSI-LD/CI/SUB/046_01
Test objective	Notifications shall only be sent if and only if the status of the corresponding subscription ("subscription.status") is active, i.e. not paused nor expired.
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.8.6
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_5_8_6
Initial conditions	**with {** **the SUT contains** **an entity** equals **${entity}** with a **property** equals **${property}** and with **value** equals **${property.value1}** and a **subscription with id set to ${subscriptionId}** and **status** equals "**active"** and **watchedAttributes** is **empty**

	and with **subscription.entity** including an **entityInfo** object with the **id** equal to **${entity.id}** **}**	
Expected behaviour	**Test events**	**Direction**
	When{ an **update of the value of ${entity}** happens and updates **value** to **${property.value2}** **the SUT needs to send out** a notification to the client **sends** a notification to the client	SUT → Client
	then { **the client at ${endpoint} receives** a valid Notification **containing** a **subscriptionId** equals to **${subscriptionId}** and a **data** containing **${entity}** with the **entity.property** equal to **${property}** and **property.value** equal to **${property.value2}** **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/SUB/046_02
Test objective	If a Subscription defines a timeInterval member, a Notification shall be sent periodically, when the time interval (in seconds) specified in such value field is reached, regardless of Attribute changes. The notification message shall include all the subscribed Entities that match the query and geoquery conditions.
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.8.6
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_5_8_6
Initial conditions	**with {** **an entity** equals **${entity}** with a **property** equals **${property}** and with **value** equals **${property.value1}** and the entity fulfills the **${query}** conditions defined in **q** and a **subscription with id set to ${subscriptionId}** and **status** equals "**active"** and **timeInterval** is set to **${timeInterval}** and **watchedAttributes** is **empty** and **q** equals **${query}**

	and **geoQ** equals **${geoQuery}** and with **subscription.entity** including an **entityInfo** object with the **id** equal to **${entity.id}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **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 { **the client at ${endpoint} receives** a valid Notification **containing** a **subscriptionId** equals to **${subscriptionId}** and a **data** containing **${entity}** with the **entity.property** equal to **${property}** and **property.value** equal to **${property.value1}** and all **${Entity}** that **matches ${query} and ${geoQ}** **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/SUB/046_03
Test objective	If a Subscription defines a timeInterval member, a Notification shall be sent periodically, when the time interval (in seconds) specified in such value field is reached, regardless of Attribute changes. The notification message shall include all the subscribed Entities if no query or geoquery are not defined
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.8.6
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_5_8_6
Initial conditions	**with {** **an entity** equals **${entity}** with a **property** equals **${property}** and with **value** equals **${property.value1}** and a **subscription with id set to ${subscriptionId}** and **status** equals "**active"** and **timeInterval** is set to **${timeInterval}** and **watchedAttributes** is **empty** and **q not defined**

	and **geoQ not defined** and with **subscription.entity** including an **entityInfo** object with the **id** equal to **${entity.id}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **When the timeinterval** is reached at **${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 { **the client at ${endpoint} receives** a valid Notification **containing** a **subscriptionId** equals to **${subscriptionId}** and a **data** containing **${entity}** with the **entity.property** equal to **${property}** and **property.value** equal to **${property.value1}** **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/SUB/046_04
Test objective	If a Subscription does not define a timeInterval term, the notification shall be sent whenever there is a change in the watched Attributes. The notification message shall include all the subscribed Entities that changed and that match (as mandated by clauses 4.9 and4.10) the query and geoquery conditions."
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.8.6
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_5_8_6
Initial conditions	**with {** **the SUT contains** **an entity** equals **${entity}** with a **property** equals **${property}** and with **value** equals **${property.value1}** and the entity fulfills the **${query}** conditions defined in **q and geoQ** and a **subscription with id set to ${subscriptionId}** and **status** equals "**active"** and **watchedAttributes** is set to **${property.name}** and **timeInterval** not defined and **q** equals **${query}**

	and **geoQ** equals **${geoQuery}** and with **subscription.entity** including an **entityInfo** object with the **id** equal to **${entity.id}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { an **update of the value of ${entity}** happens and updates **value** to **${property.value2}** **the SUT needs to send out** a notification to the client **sends** a notification to the client	SUT → Client
	then { **the client at ${endpoint} receives** a valid Notification **containing** a **subscriptionId** equals to **${subscriptionId}** and a **data** containing **${entity}** with the **entity.property** equal to **${property}** and **property.value** equal to **${property.value2}** and all **${Entity}** that **matches ${query} and ${geoQ}** **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/SUB/046_05
Test objective	If a Subscription does not define a timeInterval term, the notification shall be sent whenever there is a change in the watched Attributes. The notification message shall include all the subscribed Entities that changed and query or geoquery are not defined
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.8.6
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_5_8_6
Initial conditions	**with {** **the SUT contains** **an entity** equals **${entity}** with a **property** equals **${property}** and with **value** equals **${property.value1}** and a **subscription with id set to ${subscriptionId}** and **status** equals "**active"** and **timeInterval** not defined and **watchedAttributes** is set to **${property.name}** and **q** not defined

	and **geoQ** not defined and with **subscription.entity** including an **entityInfo** object with the **id** equal to **${entity.id}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { an **update of the value of ${entity}** happens and updates **value** to **${property.value2}** **the SUT needs to send out** a notification to the client **sends** a notification to the client	SUT → Client
	then { **the client at ${endpoint} receives** a valid Notification **containing** a **subscriptionId** equals to **${subscriptionId}** and a **data** containing **${entity}** with the **entity.property** equal to **${property}** and **property.value** equal to **${property.value2}** **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/SUB/046_06
Test objective	If a Subscription does not define a timeInterval term, the notification shall be sent whenever there is a change in the watched Attributes. If a Context Source filter is defined, then only the subscribed Entities whose origin Context Source matches the referred filter shall be included.
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.8.6
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_5_8_6
Initial conditions	**with {** **the SUT contains** **an entity** equals **${entity}** with a **property** equals **${property}** and with **value** equals **${property.value1}** and en**tity** is located at **${contextSource}** and a **subscription with id set to ${subscriptionId}** and **status** equals "**active"** and **timeInterval** not defined and **watchedAttributes** is set to **${property.name}** and **q** not defined

	and **geoQ** not defined **and csf** is **${csfList} containing only ${contextSource}** and with **subscription.entity** including an **entityInfo** object with the **id** equal to **${entity.id}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { an **update of the value of ${entity}** happens and updates **value** to **${property.value2}** **the SUT needs to send out** a notification to the client **sends** a notification to the client	SUT → Client
	then { **the client at ${endpoint} receives** a valid Notification **containing** a **subscriptionId** equals to **${subscriptionId}** and a **data** containing **${entity}** with the **entity.property** equal to **${property}** and **property.value** equal to **${property.value2}** and all entities in data are located in **${contextSource}** **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/SUB/046_07
Test objective	The structure of the notification message shall be as mandated by clause 5.3.1.
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.8.6
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_5_8_6
Initial conditions	**with {** **the SUT contains** **an entity** equals **${entity}** and a **subscription with id set to ${subscriptionId}** and **status** equals "**active"** and with **subscription.entity** including an **entityInfo** object with the **id** equal to **${entity.id}** **}**

Expected behaviour	Test events	Direction
	when { an **creation of the property of ${entity}** happens and creates **value** to **${property.value1}** **the SUT needs to send out** a notification to the client **sends** a notification to the client at **${timestamp}**	SUT → Client
	then { **the client at ${endpoint} receives** a valid Notification **containing** **id** generated and **type** equals "**notification"** **and subscriptionId** equals **${subscriptionId}** **and notifiedAt** contains a **timestamp ISO-8601 compliant** and **data** contains **${entity}** **}**	SUT → Client

Permutation on TP Id	Behaviour
046_07_01	Valid notification with attributes as stated above.
046_07_02	The Entity Attributes included (Properties or Relationships) shall be those specified by the notification.attributes member in the Subscription data type (clause 5.2.12).
046_07_03	URI expansion shall be observed (clause 5.5.7).

TP Id	TP/NGSI-LD/CI/SUB/046_08
Test objective	The structure of the notification message shall be as mandated by clause 5.3.1. The absence of the notification.attributes member of a Subscription means that all Entity Attributes shall be included
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.8.6
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_5_8_6
Initial conditions	**with {** **the SUT contains** **an entity** equals **${entity}** with a **property** equal to **${property1}** and with a **property** equal to **${property2}** and a **subscription with id set to ${subscriptionId}** and **status** equals "**active"** and **notification.attributes** is not **defined** and with **subscription.entity** including an **entityInfo** object with the **id** equal to **${entity.id}** **}**

Expected behaviour	Test events	Direction
	when { an **update of the property1 of ${entity}** happens and creates **value** to **${property1.value1}** **the SUT needs to send out** a notification to the client **sends** a notification to the client at **${timestamp}**	SUT → Client
	then { **the client at ${endpoint} receives** a valid Notification **containing** the entities containing all attributes **${property1}** and **${property2}** **}**	SUT → Client

Permutation on TP Id	behaviour
46_08_01	All attributes are included
46_08_02	If the notification.format member value is "keyValues" then a simplified representation of the entities (as mandated by clause 4.5.3) shall be provided.

TP Id	TP/NGSI-LD/CI/SUB/046_09	
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 the endpoint.uri member of the notification structure defined by clause 5.2.14.	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.8.6	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_8_6	
Initial conditions	**with {** **the SUT contains** **an entity** equals **${entity}** and a **subscription with id set to ${subscriptionId}** and **status** equals "**active"** and an **endpoint** defined as **${endpoint}** and **notification.attributes** is not **defined** and with **subscription.entity** including an **entityInfo** object with the **id** equal to **${entity.id}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { an **creates of the property1 of ${entity}** happens and creates **value** to **${property1.value1}**	SUT → Client

	the SUT needs to send out a notification to the client **sends** a notification to the client at **${timestamp}**	
	then { **the client at ${endpoint} receives** a valid Notification **at** the endpoint specified at **${endpoint.uri}** **}**	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 V1.3.1 [1], clause 5.8.6	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_8_6	
Initial conditions	**with {** **the SUT contains** **an entity** equals **${entity}** and a **subscription with id set to ${subscriptionId}** and **status** equals "**active"** and an **endpoint** defined as **${endpoint}** and **notification.attributes** is not **defined** and with **subscription.entity** including an **entityInfo** object with the **id** equal to **${entity.id}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { an **creates of the property1 of ${entity}** happens and creates **value** to **${property1.value1}** **the SUT needs to send out** a notification to the client **sends** a notification to the client at **${timestamp}**	SUT → Client
	then { **the client at ${endpoint} receives** a valid Notification **which is** in **json** **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/SUB/046_11	
Test objective	The notification.timesSent member shall be incremented by one.	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.8.6	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_8_6	
Initial conditions	**with {** **the SUT contains** **an entity** equals **${entity}** and a **subscription with id set to ${subscriptionId}** and **status** equals "**active"** and an **endpoint** defined as **${endpoint}** and **notification.attributes** is not **defined** and **notification.timeSent** equals **${timeSent}** and with **subscription.entity** including an **entityInfo** object with the **id** equal to **${entity.id}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { an **creates of the property1 of ${entity}** happens and creates **value** to **${property1.value1}** **the SUT needs to send out** a notification to the client **sends** a notification to the client at **${timestamp}**	SUT → Client
	then { **the client at ${endpoint} receives** a valid Notification **and** and **notification.timesSent** equals **${timeSent} + 1** **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/SUB/046_12
Test objective	The notification.lastNotification member shall be updated with a timestamp representing the current date and time. This test will check the format.
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.8.6
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_5_8_6

Initial conditions	**with {** **the SUT contains** **an entity** equals **${entity}** and a **subscription with id set to ${subscriptionId}** and **status** equals "**active"** and an **endpoint** defined as **${endpoint}** and **notification.attributes** is not **defined** and **notification.timeSent** equals **${timeSent}** and with **subscription.entity** including an **entityInfo** object with the **id** equal to **${entity.id}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { an **creates of the property1 of ${entity}** happens and creates **value** to **${property1.value1}** **the SUT needs to send out** a notification to the client **sends** a notification to the client at **${timestamp}**	SUT → Client
	then { **the client at ${endpoint} receives** a valid Notification and **notification.lastnotification** contains a **timestamp compliant to the ISO 8601 [**3**] format** **}**	SUT → Client

TP Id	TP/NGSI-LD/CI/SUB/046_13	
Test objective	If the response to the notification request is 200 OK then implementations shall: - Update notification.lastSuccess with a timestamp representing the current date and time. - Update notification.status to "ok".	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.8.6	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_8_6	
Initial conditions	**with {** **the SUT containing** a subscription with id **set to ${subscriptionId}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the client receives a notification** **The clients sends an 200 response to the SUT**	SUT← Client

	then { **the SUT receives the 200 notification** and **notification.lastSuccess** contains a **timestamp compliant to the ISO-8601 format** and **notification.status** equals **OK** **}**	SUT← Client

TP Id	TP/NGSI-LD/CI/SUB/046_14	
Test objective	If the response to the notification request is different than 200 OK then implementations shall: - Update notification.lastFailure with a timestamp representing the current date and time. - Update notification.status to "failed".	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.8.6	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_8_6	
Initial conditions	**with {** **the SUT containing** a subscription with id **set to ${subscriptionId}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the client receives a notification** **with {** **the SUT containing** a subscription with id **set to ${subscriptionId}** **and notification is not ok** **}** **The clients sends another than 200 response to the SUT**	SUT← Client
	then { **the SUT receives the 200 notification** **and notification.lastFailure** contains a **timestamp compliant to the ISO-8601 format** and **notification.status** equals to **failed** **}**	SUT← Client

4.2 Context Source

4.2.1 Registration

4.2.1.1 Register Context Source

TP Id	TP/NGSI-LD/CS/REG/033_01
Test objective	Check that you can create a context source registration

Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.9.2	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_5_9_2	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Context Source Registration Request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceRegistrations **and** Header: Content-Type **set to** application/ld+json **and** method **set to** POST **and** body **containing** the ${csourceRegistration} to be created **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 201 (CREATED) **and** Location header **set to** the resource URI of the created context source registration resource **and contains** the ${csourceRegistration} **}**	SUT → Client

Permutation on TP Id	${csourceRegistration}
TP/NGSI-LD/CS/REG/033_01_01	Specific id with expiration date
TP/NGSI-LD/CS/REG/033_01_02	Never expires (expiresAt not defined)
TP/NGSI-LD/CS/REG/033_01_03	Invalid specified id (a valid one is assigned)

TP Id	TP/NGSI-LD/CS/REG/033_02
Test objective	Check that you cannot create a context source with invalid content
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.9.2
Config Id	CF_05
Parent Release	V1.3.1
PICS Selection	PICS_5_9_2
Initial conditions	**with {** **the SUT being** in the "initial state" **}**

Expected behaviour	Test events	Direction
	when { **the SUT receives** a valid Create Context Source Registration Request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceRegistrations **and** Header: Content-Type **set to** application/ld+json **and** method **set to** POST **and** body **containing** the ${invalid_body} to be created **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request) **and** Response Body **containing** ProblemDetails element **containing** type element **set to ${problem_type} and** title element **containing** more information about the error **}**	SUT → Client

Permutation on TP Id	${invalid_body}	${problem_type}
TP/NGSI-LD/CS/REG/033_02_01	invalid JSON document	https://uri.etsi.org/ngsi-ld/errors/InvalidRequest
TP/NGSI-LD/CS/REG/033_02_02	Different data structure than CsourceRegistration	https://uri.etsi.org/ngsi-ld/errors/BadRequestData
TP/NGSI-LD/CS/REG/033_02_03	Date in the past	https://uri.etsi.org/ngsi-ld/errors/BadRequestData

TP Id	TP/NGSI-LD/CS/REG/033_03	
Test objective	Check that you cannot create a context source registration that already exists	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.9.2	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_9_2	
Initial conditions	**with {** **the SUT being** in the "initial state" and **containing** an initial Context Source Registration with an id **set to** ${registrationId} **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Context Source Registration Request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceRegistrations **and** Header: Content-Type **set to** application/ld+json **and**	SUT ← Client

	method **set to** POST **and** body **containing** ${csourceRegistration} with the id **set to** ${registrationId} **}**	
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 409 (Already Exists) **and** Response Body **containing** ProblemDetails element **containing** type element **set to** https://uri.etsi.org/ngsi-ld/errors/AlreadyExists **and** title element **containing** more information about the error **}**	SUT → Client

TP Id	TP/NGSI-LD/CS/REG/033_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 V1.3.1 [1], clause 6.3.5	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_6_3_5	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Context Source Registration Request **from** the client **containing** URL **set to** /ngsi-ld/v1/cSourceRegistrations **and** Header: Content-Type **set to** application/json **and** Header: Link **set to** a @context **containing** terms used by the context source registration to create body **set to** context source registration to be created **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 201 (Created) **and** Persisted Context Source Registration contains attributes expanded as per the supplied @context **}**	SUT → Client

TP Id	TP/NGSI-LD/CS/REG/033_05	
Test objective	Check that the default @context is used if the Content-Type header is "application/json" and the Link header does not contain a JSON-LD @context	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 6.3.5	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_6_3_5	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Context Source Registration Request **from** the client **containing** URL **set to** /ngsi-ld/v1/cSourceRegistrations **and** Header: Content-Type **set to** application/json **and** body **set to** context source registration to be created **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 201 (Created) **and** Persisted Context Source Registration contains attributes expanded as per the default @context **}**	SUT → Client

TP Id	TP/NGSI-LD/CS/REG/033_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 V1.3.1 [1], clause 6.3.5	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_6_3_5	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Context Source Registration Request **from** the client **containing**	SUT ← Client

	URL **set to** /ngsi-ld/v1/cSourceRegistrations **and** Header: Content-Type **set to** application/json **and** body **set to** context source registration **containing** a @context term **}**	
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request) **and** Response Body **containing** ProblemDetails element **containing** type element **set to** https://uri.etsi.org/ngsi-ld/errors/BadRequestData **and** title element **containing** more information about the error **}**	SUT → Client

TP Id	TP/NGSI-LD/CS/REG/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 V1.3.1 [1], clause 6.3.5	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_6_3_5	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Context Source Registration Request **from** the client **containing** URL **set to** /ngsi-ld/v1/cSourceRegistrations **and** Header: Content-Type **set to** application/ld+json **and** body **set to** context source registration **containing** a @context term **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 201 (Created) **and** Persisted Context Source Registration contains attributes expanded as per the supplied @context **}**	SUT → Client

TP Id	TP/NGSI-LD/CS/REG/033_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 V1.3.1 [1], clause 6.3.5	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_6_3_5	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Context Source Registration Request **from** the client **containing** URL **set to** /ngsi-ld/v1/cSourceRegistrations **and** Header: Content-Type **set to** application/ld+json **and** body **set to** context source registration **not containing** a @context term **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request) **and** Response Body **containing** ProblemDetails element **containing** type element **set to** https://uri.etsi.org/ngsi-ld/errors/BadRequestData **and** title element **containing** more information about the error **}**	SUT → Client

TP Id	TP/NGSI-LD/CS/REG/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 V1.3.1 [1], clause 6.3.5
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_6_3_5

Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Context Source Registration Request **from** the client **containing** URL **set to** /ngsi-ld/v1/cSourceRegistrations **and** Header: Content-Type **set to** application/ld+json **and** Header: Link **set to** a @context **containing** terms used by the context source registration to create body **set to** entity to be created **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request) **and** Response Body **containing** ProblemDetails element **containing** type element **set to** https://uri.etsi.org/ngsi-ld/errors/BadRequestData **and** title element **containing** more information about the error **}**	SUT → Client

4.2.1.2 Update Context Source Registration

TP Id	TP/NGSI-LD/CS/REG/034_01	
Test objective	Check that you can update a context source registration by id	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.9.3	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_9_3	
Initial conditions	**with {** **the SUT being** in the "initial state" and **containing** an initial Context Source Registration with an id **set to** ${registrationId} **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Context Source Registration Request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceRegistrations/${registrationId} **and**	SUT ← Client

	Header: Content-Type **set to** application/ld+json **and** method **set to** PATCH **and** body **containing** ${csourceRegistration} **}**	
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 204 (No Content) **and contains** the updated ${csourceRegistration} **}**	SUT → Client

Permutation on TP Id	${csourceRegistration}
TP/NGSI-LD/CS/REG/034_01_01	Expiration date
TP/NGSI-LD/CS/REG/034_01_02	Never expires (expiresAt not defined)

TP Id	TP/NGSI-LD/CS/REG/034_02	
Test objective	Check that you cannot update a context source registration under some conditions	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.9.3	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_9_3	
Initial conditions	**with {** **the SUT being** in the "initial state" and **containing** an initial Context Source Registration with an id **set to** ${registrationId} **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Context Source Registration Request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceRegistrations/${registrationId} **and** Header: Content-Type **set to** application/ld+json **and** method **set to** PATCH **and** body **containing** ${csourceRegistration} **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request) **and** Response Body **containing** ProblemDetails element **containing** type element **set to ${problem_type} and** title element **containing** more information about the error	SUT → Client

	}	

Permutation on TP Id	${csourceRegistration}	${registrationId}	${problem_type}
TP/NGSI-LD/CS/REG/034_02_01	valid	empty	https://uri.etsi.org/ngsi-ld/errors/BadRequestData
TP/NGSI-LD/CS/REG/034_02_02	valid	invalid URI	https://uri.etsi.org/ngsi-ld/errors/BadRequestData
TP/NGSI-LD/CS/REG/034_02_03	different data type	valid	https://uri.etsi.org/ngsi-ld/errors/BadRequestData
TP/NGSI-LD/CS/REG/034_02_04	without mandatory property	valid	https://uri.etsi.org/ngsi-ld/errors/BadRequestData
TP/NGSI-LD/CS/REG/034_02_05	invalid json	valid	https://uri.etsi.org/ngsi-ld/errors/InvalidRequest

TP Id	TP/NGSI-LD/CS/REG/034_03	
Test objective	Check that you cannot update a context source registration by id if the id is not known to the system	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.9.3	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_9_3	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Context Source Registration Request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceRegistrations/${registrationId} **and** Header: Content-Type **set to** application/ld+json **and** method **set to** PATCH **and** body **containing** ${csourceRegistration} **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 404 (Not Found) **and** Response Body **containing** ProblemDetails element **containing** type element **set to** https://uri.etsi.org/ngsi-ld/errors/ResourceNotFound **and** title element **containing** more information about the error **}**	SUT → Client

4.2.1.3 Delete Context Source Registration

TP Id	TP/NGSI-LD/CS/REG/035_01	
Test objective	Check that you can delete a context source registration by id	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.9.4	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_9_4	
Initial conditions	**with {** **the SUT being** in the "initial state" and **containing** an initial Context Source Registration with an id **set to** ${registrationId} **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Context Source Registration Request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceRegistrations/${registrationId} **and** Header: Content-Type **set to** application/ld+json **and** method **set to** DEL **and** body **containing** ${csourceRegistration} **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 204 (No Content) **}**	SUT → Client

TP Id	TP/NGSI-LD/CS/REG/035_02	
Test objective	Check that you cannot delete a context source registration under some conditions	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.9.3	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_9_4	
Initial conditions	**with {** **the SUT being** in the "initial state" and **containing** an initial Context Source Registration with an id **set to** ${registrationId} **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Context Source Registration Request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceRegistrations/${registrationId} **and**	SUT ← Client

	Header: Content-Type **set to** application/ld+json **and** method **set to** DEL **and** body **containing** ${csourceRegistration} **}**	
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request) **and** Response Body **containing** ProblemDetails element **containing** type element **set to** https://uri.etsi.org/ngsi-ld/errors/BadRequestData **and** title element **containing** more information about the error **}**	SUT → Client

Permutation on TP Id	${registrationId}
TP/NGSI-LD/CS/REG/035_02_01	empty
TP/NGSI-LD/CS/REG/035_02_02	invalid URI

TP Id	TP/NGSI-LD/CS/REG/035_03	
Test objective	Check that you cannot delete a context source registration by id if the id is not known to the system	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.9.3	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_5_9_3	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Context Source Registration Request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceRegistrations/${registrationId} **and** Header: Content-Type **set to** application/ld+json **and** method **set to** DEL **and** body **containing** ${csourceRegistration} **}**	SUT ← Client

	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 404 (Not Found) **and** Response Body **containing** ProblemDetails element **containing** type element **set to** https://uri.etsi.org/ngsi-ld/errors/ResourceNotFound **and** title element **containing** more information about the error **}**	SUT → Client

4.2.2 Registration Subscription

4.2.2.1 Create Context Source Registration Subscription

TP Id	TP/NGSI-LD/CS/REGSUB/038_01	
Test objective	Check that you can create a minimal context source registration subscription	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.11.2	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_5_11_2	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Context Source Registration Subscription Request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceSubscriptions **and** Header: Content-Type **set to** application/ld+json **and** body **containing ${subscription}** **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 201 (CREATED) **and** Location header **set to** the resource URI of the subscription created **and** created resource **set to ${subscription}** **}**	SUT → Client

TP Id	TP/NGSI-LD/CS/REGSUB/038_02	
Test objective	Check that you can create a context source registration subscription without providing an id and it will be automatically generated	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.11.2	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_5_11_2	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Context Source Registration Subscription Request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceSubscriptions **and** Header: Content-Type **set to** application/ld+json **and** body **containing ${subscription} not containing** an id **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 201 (CREATED) **and** Location header **set to** the resource URI of the subscription created **and** created resource **set to ${subscription} with** auto generated id **}**	SUT → Client

TP Id	TP/NGSI-LD/CS/REGSUB/038_03	
Test objective	Check that you can create a context source registration subscription without providing isActive member and will be active by default	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.11.2	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_5_11_2	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Context Source Registration Subscription Request **from** the client **containing**	SUT ← Client

	URL **set to** /ngsi-ld/v1/csourceSubscriptions **and** Header: Content-Type **set to** application/ld+json **and** body **containing** the subscription to be created **not containing** isActive member **}**	
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 201 (CREATED) **and** Location header **set to** the resource URI of the subscription created **and** the subscription isActive member **set to** true **}**	SUT → Client

TP Id	TP/NGSI-LD/CS/REGSUB/038_04	
Test objective	Check that you can create a context source registration subscription with isActive member set to false and its initial status will be set to "paused"	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.11.2	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_5_11_2	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Context Source Registration Subscription Request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceSubscriptions **and** Header: Content-Type **set to** application/ld+json **and** body **containing** the subscription to be created **containing** isActive member **set to** false **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 201 (CREATED) **and** Location header **set to** the resource URI of the subscription created **and** the subscription isActive member is **set to** false **}**	SUT → Client

TP Id	TP/NGSI-LD/CS/REGSUB/038_05	
Test objective	Check that you can create a context source registration subscription with an expiresAt member and when it is due the status of the subscription changes to expired	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.11.2	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_5_11_2	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Context Source Registration Subscription Request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceSubscriptions **and** Header: Content-Type **set to** application/ld+json **and** body **containing** the subscription to be created **containing** expiresAt member **set to** CurrentDateTime + 5 seconds **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 201 (CREATED) **and** Location header **set to** the resource URI of the subscription created **and after** 10 seconds the subscription isActive member **set to** false **}**	SUT → Client

TP Id	TP/NGSI-LD/CS/REGSUB/038_06
Test objective	Check that you can create a context source registration subscription without an expiresAt member and it will be considered as perpetual
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.11.2
Config Id	CF_05
Parent Release	V1.3.1
PICS Selection	PICS_5_11_2
Initial conditions	**with {** **the SUT being** in the "initial state" **}**

Expected behaviour	Test events	Direction
	when { **the SUT receives** a valid Create Context Source Registration Subscription Request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceSubscriptions **and** Header: Content-Type **set to** application/ld+json **and** body **containing** the subscription to be created **not containing** expiresAt member **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 201 (CREATED) **and** Location header **set to** the resource URI of the subscription created **and** the subscription status is always **set to** active **}**	SUT → Client

TP Id	TP/NGSI-LD/CS/REGSUB/038_07	
Test objective	Check that you cannot create a context source registration subscription where another context source registration subscription whose id is equivalent exists, an error of type AlreadyExists shall be raised	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.11.2	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_5_11_2	
Initial conditions	**with {** **the SUT containing** a Context Source Registration Subscription (CSRS1) **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Context Source Registration Subscription Request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceSubscriptions **and** Header: Content-Type **set to** application/ld+json **and** body **containing** the subscription to be created **containing** the same id **of** CSRS1 **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 409 (Conflict) **and** Response Body **containing**	SUT → Client

	ProblemDetails element **containing** type element **set to** https://uri.etsi.org/ngsi-ld/errors/AlreadyExists **and** title element **containing** more information about the error **}**	

TP Id	TP/NGSI-LD/CS/REGSUB/038_08	
Test objective	Check that you cannot create a context source registration subscription If the data types, cardinalities and restrictions expressed by clause 5.2.12 are not met, an error of type BadRequestData shall be raised	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.11.2	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_5_11_2	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Context Source Registration Subscription Request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceSubscriptions **and** Header: Content-Type **set to** application/ld+json **and** body **set to ${subscription}** **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request) **and** Response Body **containing** ProblemDetails element **containing** type element **set to** https://uri.etsi.org/ngsi-ld/errors/BadRequestData **and** title element **containing** more information about the error **}**	SUT → Client

Permutation on TP Id	${subscription}
TP/NGSI-LD/CS/REGSUB/038_08_01	subscription **not containing** notification member
TP/NGSI-LD/CS/REGSUB/038_08_02	subscription **containing** invalid type member
TP/NGSI-LD/CS/REGSUB/038_08_03	subscription **containing** invalid q member
TP/NGSI-LD/CS/REGSUB/038_08_04	subscription **containing** empty watchedAttributes member

TP Id	TP/NGSI-LD/CS/REGSUB/038_09	
Test objective	Check that you cannot create a context source registration subscription with an expiration timestamp representing a moment before the current date and time, an error of type BadRequestData shall be raised	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.11.2	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_5_11_2	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Context Source Registration Subscription Request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceSubscriptions **and** Header: Content-Type **set to** application/ld+json **and** body **containing** the subscription to be created **containing** expiresAt member **before** CurrentDateTime **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request) **and** Response Body **containing** ProblemDetails element **containing** type element **set to** https://uri.etsi.org/ngsi-ld/errors/BadRequestData **and** title element **containing** more information about the error **}**	SUT → Client

4.2.2.2 Update Context Source Registration Subscription

TP Id	TP/NGSI-LD/CS/REGSUB/039_01
Test objective	Check that you can update a context source registration subscription
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.11.3
Config Id	CF_05
Parent Release	V1.3.1
PICS Selection	PICS_5_11_3

Initial conditions	**with {** **the SUT containing** a Context Source Registration Subscription (CSRS1) **with** an id **set to ${subscriptionId}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Update Context Source Registration Subscription Request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceSubscriptions/**${subscriptionId} and** Header: Content-Type **set to** application/json **and** body **containing ${update_fragment}** **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 204 (No Content) **and** updated resource **set to the** subscription updated **with ${update_fragment}** **}**	SUT → Client

TP Id	TP/NGSI-LD/CS/REGSUB/039_02	
Test objective	Check that you cannot update a context source registration subscription with an invalid URI, an error of type BadRequestData shall be raised	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.11.3	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_5_11_3	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Update Context Source Registration Subscription Request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceSubscriptions/**${invalidId} and** Header: Content-Type **set to** application/json **and** body **containing** the subscription fragment to be updated **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request) **and**	SUT → Client

	Response Body **containing** ProblemDetails element **containing** type element **set to** https://uri.etsi.org/ngsi-ld/errors/BadRequestData **and** title element **containing** more information about the error **}**	

TP Id	TP/NGSI-LD/CS/REGSUB/039_03	
Test objective	Check that you cannot update an unknown context source registration subscription, an error of type ResourceNotFound shall be raised	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.11.3	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_5_11_3	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Update Context Source Registration Subscription Request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceSubscriptions/**${unknownUri} and** Header: Content-Type **set to** application/json **and** body **containing** the subscription fragment to be updated **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 404 (Not Found) **and** Response Body **containing** ProblemDetails element **containing** type element **set to** https://uri.etsi.org/ngsi-ld/errors/ResourceNotFound **and** title element **containing** more information about the error **}**	SUT → Client

TP Id	TP/NGSI-LD/CS/REGSUB/039_04

Test objective	Check that you cannot update a context source registration subscription with a fragment that does not meet the data types and restrictions expressed by clause 5.2.12, an error of type BadRequestData shall be raised	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.11.3	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_5_11_3	
Initial conditions	**with {** **the SUT containing** a Context Source Registration Subscription (CSRS1) **with** an id **set to ${subscriptionId}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Update Context Source Registration Subscription Request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceSubscriptions/**${subscriptionId} and** Header: Content-Type **set to** application/json **and** body **set to ${subscription_fragment}** **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request Data) **and** Response Body **containing** ProblemDetails element **containing** type element **set to** https://uri.etsi.org/ngsi-ld/errors/BadRequestData **and** title element **containing** more information about the error **}**	SUT → Client

Permutation on TP Id	**${subscription_fragment}**
TP/NGSI-LD/CS/REGSUB/039_04_01	subscription fragment **containing** type member **set to** null
TP/NGSI-LD/CS/REGSUB/039_04_02	subscription fragment **containing** notification member **not containing** an endpoint

TP Id	TP/NGSI-LD/CS/REGSUB/039_05
Test objective	Check that you cannot update a context source registration subscription with an invalid request body (invalid JSON document), an error of type InvalidRequest shall be raised
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.11.3
Config Id	CF_05
Parent Release	V1.3.1
PICS Selection	PICS_5_11_3

Initial conditions	**with {** **the SUT containing** a Context Source Registration Subscription (CSRS1) **with** an id **set to** **${subscriptionId}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Update Context Source Registration Subscription Request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceSubscriptions/**${subscriptionId} and** Header: Content-Type **set to** application/json **and** body **set to** invalid json **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request Data) **and** Response Body **containing** ProblemDetails element **containing** type element **set to** https://uri.etsi.org/ngsi-ld/errors/InvalidRequest **and** title element **containing** more information about the error **}**	SUT → Client

4.2.2.3 Retrieve Context Source Registration Subscription

TP Id	TP/NGSI-LD/CS/REGSUB/040_01	
Test objective	Check that you can retrieve a context source registration subscription	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.11.4	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_5_11_4	
Initial conditions	**with {** **the SUT containing** a Context Source Registration Subscription (CSRS1) **with** an id **set to** **${subscriptionId}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Retrieve Context Source Registration Subscription Request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceSubscriptions/**${subscriptionId}** **}**	SUT ← Client

	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 200 (Ok) **and** Response Body **containing** the representation of CSRS1 **}**	SUT → Client

TP Id	TP/NGSI-LD/CS/REGSUB/040_02	
Test objective	Check that you cannot retrieve a context source registration subscription with an invalid URI, an error of type BadRequestData shall be raised	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.11.4	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_5_11_4	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Retrieve Context Source Registration Subscription Request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceSubscriptions /**${invalidId}** **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request Data) **and** Response Body **containing** ProblemDetails element **containing** type element **set to** https://uri.etsi.org/ngsi-ld/errors/BadRequestData **and** title element **containing** more information about the error **}**	SUT → Client

TP Id	TP/NGSI-LD/CS/REGSUB/040_03
Test objective	Check that you cannot retrieve an unknown context source registration subscription, an error of type ResourceNotFound shall be raised
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.11.4
Config Id	CF_05
Parent Release	V1.3.1

PICS Selection	PICS_5_11_4	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Retrieve Context Source Registration Subscription Request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceSubscriptions /**${unknownUri}** **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 404 (Not Found) **and** Response Body **containing** ProblemDetails element **containing** type element **set to** https://uri.etsi.org/ngsi-ld/errors/ResourceNotFound **and** title element **containing** more information about the error **}**	SUT → Client

4.2.2.4 Query Context Source Registration Subscriptions

TP Id	TP/NGSI-LD/CS/REGSUB/041_01
Test objective	Check that you can query context source registration subscriptions
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.11.5
Config Id	CF_05
Parent Release	V1.3.1
PICS Selection	PICS_5_11_5
Initial conditions	**with {** **the SUT containing** a Context Source Registration Subscription (CSRS1) **with** an id **set to ${subscriptionId1}** **and** a Context Source Registration Subscription (CSRS2) **with** an id **set to ${subscriptionId2}** **}**

Expected behaviour	Test events	Direction
	when { **the SUT receives** a valid Query Context Source Registration Subscriptions Request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceSubscriptions **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 200 (Ok) **and** Response Body **containing** a list **of** two representations of CSRS1 **and** CSRS2 **}**	SUT → Client

TP Id	TP/NGSI-LD/CS/REGSUB/041_02	
Test objective	Check that you 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 V1.3.1 [1], clause 5.11.5	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_5_11_5	
Initial conditions	**with {** **the SUT containing** a Context Source Registration Subscription (CSRS1) **with** an id **set to ${subscriptionId1}** **and** a Context Source Registration Subscription (CSRS2) **with** an id **set to ${subscriptionId2}** **and** a Context Source Registration Subscription (CSRS3) **with** an id **set to ${subscriptionId3}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Query Context Source Registration Subscriptions Request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceSubscriptions **and** Query Parameter limit **set to ${limit}** **}**	SUT ← Client

	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 200 (Ok) **and** Response Body **containing** a list **of ${number}** context source registration subscriptions **}**	SUT → Client

Permutation on TP Id	${limit}	${number}
TP/NGSI-LD/CS/REGSUB/041_02_01	1	1
TP/NGSI-LD/CS/REGSUB/041_02_02	2	2
TP/NGSI-LD/CS/REGSUB/041_02_03	15	3

TP Id	TP/NGSI-LD/CS/REGSUB/041_03	
Test objective	Check that you can query context source registration subscriptions with providing page and limit parameters for pagination, pagination logic shall be in place as mandated by clause 5.5.9.	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.11.5	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_5_11_5	
Initial conditions	**with {** **the SUT containing** a Context Source Registration Subscription (CSRS1) **with** an id **set to ${subscriptionId1}** **and** a Context Source Registration Subscription (CSRS2) **with** an id **set to ${subscriptionId2}** **and** a Context Source Registration Subscription (CSRS3) **with** an id **set to ${subscriptionId3}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Query Context Source Registration Subscriptions Request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceSubscriptions **and** Query Parameter limit **set to ${limit} and** Query Parameter page **set to ${page}** **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 200 (Ok) **and** Response Body **containing** a list **of** context source registration subscriptions respecting the pagination logic **}**	SUT → Client

Permutation on TP Id	${limit}	${page}
TP/NGSI-LD/CS/REGSUB/041_03_01	1	2
TP/NGSI-LD/CS/REGSUB/041_03_02	2	2
TP/NGSI-LD/CS/REGSUB/041_03_03	15	1

TP Id	TP/NGSI-LD/CS/REGSUB/041_04	
Test objective	Check that you cannot query context source registration subscriptions with invalid page and limit parameters	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.11.5	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_5_11_5	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Query Context Source Registration Subscriptions Request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceSubscriptions **and** Query Parameter limit **set to ${limit} and** Query Parameter page **set to ${page}** **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request Data) **and** Response Body **containing** ProblemDetails element **containing** type element **set to** https://uri.etsi.org/ngsi-ld/errors/BadRequestData **and** title element **containing** more information about the error **}**	SUT → Client

Permutation on TP Id	${limit}	${page}
TP/NGSI-LD/CS/REGSUB/041_04_01	-5	2
TP/NGSI-LD/CS/REGSUB/041_04_02	2	-3
TP/NGSI-LD/CS/REGSUB/041_04_03	0	0

4.2.2.5 Delete Context Source Registration Subscription

TP Id	TP/NGSI-LD/CS/REGSUB/042_01
Test objective	Check that you can delete a context source registration subscription

Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.11.6	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_5_11_6	
Initial conditions	**with {** **the SUT containing** a Context Source Registration Subscription (CSRS1) **with** an id **set to** **${subscriptionId}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Delete Context Source Registration Subscription Request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceSubscriptions/**${subscriptionId}** **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 204 (No Content) **and the SUT not containing** resource **with** id **set to** **${subscriptionId}** **}**	SUT → Client

TP Id	TP/NGSI-LD/CS/REGSUB/042_02	
Test objective	Check that you cannot delete a context source registration subscription with an invalid URI, an error of type BadRequestData shall be raised	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.11.6	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_5_11_6	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Delete Context Source Registration Subscription Request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceSubscriptions /**${invalidId}** **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 400 (Bad Request Data) **and**	SUT → Client

	Response Body **containing** ProblemDetails element **containing** type element **set to** https://uri.etsi.org/ngsi-ld/errors/BadRequestData **and** title element **containing** more information about the error **}**	

TP Id	TP/NGSI-LD/CS/REGSUB/042_03	
Test objective	Check that you cannot delete an unknown context source registration subscription, an error of type ResourceNotFound shall be raised	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.11.6	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_5_11_6	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Delete Context Source Registration Subscription Request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceSubscriptions /**${unknownUri}** **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 404 (Not Found) **and** Response Body **containing** ProblemDetails element **containing** type element **set to** https://uri.etsi.org/ngsi-ld/errors/ResourceNotFound **and** title element **containing** more information about the error **}**	SUT → Client

4.2.2.6 Context Source Registration Subscription Notification Behaviour

TP Id	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, independent of any changes to the set of Context Source registrations	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.11.7	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_5_11_7	
Initial conditions	**with {** **the SUT containing** a Context Source Registration (CSR1) providing latest information about some entities **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Context Source Registration Subscription Request **from** the client **with** entities member matching CSR1 **and** timeInterval member **set to** 10 **}**	SUT ← Client
	then { **the SUT sends to** the endpoint URI mentioned in CSRS1, a CsourceNotification **at** the subscription creation **and** every 10 seconds **}**	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 a timeInterval member, a cSourceNotification, with the appropriate trigger reason in the "triggerReason" member, will be sent initially on subscription and whenever there is a change of a matching Context Source Registration	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.11.7	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_5_11_7	
Initial conditions	**with {** **the SUT containing** a Context Source Registration (CSR1) providing latest information about some entities **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Context Source Registration Subscription (CSRS1) Request **from** the client **with** entities member matching CSR1 **and** without timeInterval member **}**	SUT ← Client

	then { **the SUT sends to** the endpoint URI mentioned in CSRS1, a CsourceNotification **at** subscription creation **containing** type **set to "**ContextSource Notification" **and** subscriptionId **set to** id **of** CSRS1 **and** notifiedAt **a** timestamp **and** data **set to a** list **containing** CSR1 **and** triggerReason **set to** "newlyMatching" **and when** CSR1 endpoint parameter is updated **the SUT sends a** CsourceNotification **containing** type **set to "**ContextSource Notification" **and** subscriptionId **set to** id **of** CSRS1 **and** notifiedAt **a** timestamp **and** data **set to a** list **containing** CSR1 **and** triggerReason element **set to** "updated" **}**	SUT → Client

TP Id	TP/NGSI-LD/CS/REGSUB/047_03	
Test objective	Check that instead of providing the original context source registration which may contain a lot of irrelevant information, implementations should return filtered context source registrations, which only contain context source registration information relevant for the subscription, in particular only matching RegistrationInfo elements	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.11.7	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_5_11_7	
Initial conditions	**with {** **the SUT containing** a Context Source Registration Subscription (CSRS1) **with** entities member matching entities **of** type X **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Context Source Registration Request **from** the client providing information about entities **of** type X **and** Y **}**	SUT ← Client

	then { **the SUT sends to** the endpoint URI mentioned in CSRS1, a CsourceNotification **containing** type **set to "**ContextSource Notification" **and** subscriptionId **set to** id **of** CSRS1 **and** notifiedAt **a** timestamp **and** data **set to a** list **containing** CSR1 **containing** information about entities **of** type X **and** triggerReason **set to** "newlyMatching" **}**	SUT → Client

TP Id	TP/NGSI-LD/CS/REGSUB/047_04	
Test objective	The structure of the csource notification message shall be as mandated by clause 5.3.2	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.11.7	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_5_11_7	
Initial conditions	**with {** **the SUT containing** a Context Source Registration Subscription (CSRS1) **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Context Source Registration Request **from** the client matching CSRS1 **}**	SUT ← Client
	then { **the SUT sends to** the endpoint URI mentioned in CSRS1, a CsourceNotification **containing** type **set to "**ContextSource Notification" **and** subscriptionId **set to** id **of** CSRS1 **and** notifiedAt **a** timestamp **and** data **set to a** list **containing** CSR1 **and** triggerReason **set to** "newlyMatching" **}**	SUT → Client

TP Id	TP/NGSI-LD/CS/REGSUB/047_05
Test objective	Check that if a cSourceNotification is sent successfully to the "endpoint" member, the "notification.timesSent" member shall be incremented by one and the "notification.lastSuccess" and "notification.lastNotification" members shall be updated with the

	current timestamp and the status of the context source registration subscription shall be updated to "ok"	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.11.7	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_5_11_7	
Initial conditions	**with {** **the SUT containing** a Context Source Registration Subscription (CSRS1) **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Context Source Registration (CSR1) Request **from** the client matching CSRS1 **}**	SUT ← Client
	then { **the SUT sends to** the endpoint URI mentioned in CSRS1, a CsourceNotification **containing** type **set to "**ContextSource Notification" **and** subscriptionId **set to** id **of** CSRS1 **and** notifiedAt **a** timestamp **and** data **set to a** list **containing** CSR1 **and** triggerReason **set to** "newlyMatching" **and** increments by one the notification.timesSent member **of** CSRS1 **and** updates the notification.lastNotification **of** CSRS1 **with** the current timestamps **}**	SUT → Client

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 V1.3.1 [1], clause 5.11.7
Config Id	CF_05
Parent Release	V1.3.1
PICS Selection	PICS_5_11_7

Initial conditions	**with {** **the SUT containing** a Context Source Registration Subscription (CSRS1) **with** notification member **containing** an unreachable endpoint **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Context Source Registration Request **from** the client matching CSRS1 **}**	SUT ← Client
	then { **the SUT** fails **in sending** the CsourceNotification **and** updates the notification.lastFailure **of** CSRS1 **with** the current timestamps **and** updates the notification.status **of** CSRS1 **with "**failed" **}**	SUT → Client

TP Id	TP/NGSI-LD/CS/REGSUB/047_07
Test objective	Check that a cSourceNotification shall only be sent if and only if the status of the corresponding subscription ("subscription.status") is active not paused nor expired
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.11.7
Config Id	CF_05
Parent Release	V1.3.1
PICS Selection	PICS_5_11_7
Initial conditions	**with {** **the SUT containing** **${state}** Context Source Registration Subscription (CSRS1) **}**

Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Context Source Registration Request **from** the client matching CSRS1 **}**	SUT ← Client
	then { **the SUT** will not **send** a CsourceNotification **}**	SUT → Client

Permutation on TP Id	**${state}**
TP/NGSI-LD/CS/REGSUB/047_07_01	paused
TP/NGSI-LD/CS/REGSUB/047_07_02	expired

TP Id	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 V1.3.1 [1], clause 5.11.7	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_5_11_7	
Initial conditions	**with {** **the SUT containing** a Context Source Registration (CSR1) providing latest information about some entities **and** a Context Source Registration Subscription (CSRS1) without temporalQ member **and with** entities member matching entities **of** CSR1 **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Update Context Source Registration Request **from** the client **to** update the endpoint member **of** CSR1 **and** CSR1 still matches CSRS1 **}**	SUT ← Client
	then { **the SUT sends to** the endpoint URI mentioned in CSRS1, a CsourceNotification **containing** type **set to "**ContextSource Notification" **and** subscriptionId **set to** id **of** CSRS1 **and** notifiedAt **a** timestamp **and** data **set to a** list **containing** CSR1 **and** triggerReason **set to "**updated" **}**	SUT → Client

TP Id	TP/NGSI-LD/CS/REGSUB/047_09
Test objective	Check if a context source registration subscription defines an entities member, a CsourceNotification will be triggered from context source registrations with information member matching the described entities
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.11.7
Config Id	CF_05
Parent Release	V1.3.1
PICS Selection	PICS_5_11_7
Initial conditions	**with {** **the SUT containing**

	a Context Source Registration (CSR1) providing latest information about some entities **and** a Context Source Registration Subscription (CSRS1) **with** entities member matching entities **of** CSR1 **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Update Context Source Registration Request **from** the client **to** update information member **of** CSR1 **to** no longer match CSRS1 **}**	SUT ← Client
	then { **the SUT sends to** the endpoint URI mentioned in CSRS1, a CsourceNotification **containing** type **set to "**ContextSource Notification" **and** subscriptionId **set to** id **of** CSRS1 **and** notifiedAt **a** timestamp **and** data **set to a** list **containing** CSR1 **and** triggerReason **set to "**noLongerMatching" **}**	SUT → Client

TP Id	TP/NGSI-LD/CS/REGSUB/047_10
Test objective	Check if a context source registration subscription defines temporalQ member with timeproperty observedAt, the temporal query is matched against the observationInterval of matching context source registrations
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.11.7
Config Id	CF_05
Parent Release	V1.3.1
PICS Selection	PICS_5_11_7
Initial conditions	**with {** **the SUT containing** a Context Source Registration Subscription (CSRS1) **with** temporalQ member **containing** timeproperty **set to** observedAt **}**

Expected behaviour	Test events	Direction
	when { **the SUT receives** a valid Create Context Source Registration Request **from** the client **with** information **and** observationInterval members matching CSRS1 **}**	SUT ← Client
	then { **the SUT sends to** the endpoint URI mentioned in CSRS1, a CsourceNotification **containing** type **set to "**ContextSource Notification" **and** subscriptionId **set to** id **of** CSRS1 **and** notifiedAt **a** timestamp **and** data **set to a** list **containing** CSR1 **and** triggerReason **set to** "newlyMatching" **}**	SUT → Client

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 managementInterval of matching context source registrations	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.11.7	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_5_11_7	
Initial conditions	**with {** **the SUT containing** a Context Source Registration Subscription (CSRS1) **with** temporalQ member **containing** timeproperty **set to ${timeproperty}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Context Source Registration Request **from** the client **with** information **and** managementInterval members matching CSRS1 **}**	SUT ← Client
	then { **the SUT sends to** the endpoint URI mentioned in CSRS1, a CsourceNotification **containing** type **set to "**ContextSource Notification" **and** subscriptionId **set to** id **of** CSRS1 **and** notifiedAt **a** timestamp **and** data **set to a** list **containing** CSR1 **and**	SUT → Client

	triggerReason **set to "**newlyMatching" **}**	

Permutation on TP Id	${timeproperty}
TP/NGSI-LD/CS/REGSUB/047_11_01	createdAt
TP/NGSI-LD/CS/REGSUB/047_11_02	modifiedAt

TP Id	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 V1.3.1 [1], clause 5.11.7	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_5_11_7	
Initial conditions	**with {** **the SUT containing** a Context Source Registration Subscription (CSRS1) **with** entities **and** watchedAttributes members **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Context Source Registration Request **from** the client **with** information member matching CSRS1 **}**	SUT ← Client
	then { **the SUT sends to** the endpoint URI mentioned in CSRS1, a CsourceNotification **containing** type **set to "**ContextSource Notification" **and** subscriptionId **set to** id **of** CSRS1 **and** notifiedAt **a** timestamp **and** data **set to a** list **containing** CSR1 **and** triggerReason **set to "**newlyMatching" **}**	SUT → Client

TP Id	TP/NGSI-LD/CS/REGSUB/047_13
Test objective	Check if a context source registrations subscription does not define watchedAttributes member, a CsourceNotification will be triggered from context source registrations with information member matching all attributes of the described entities
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.11.7
Config Id	CF_05
Parent Release	V1.3.1

PICS Selection	PICS_5_11_7	
Initial conditions	**with {** **the SUT containing** a Context Source Registration Subscription (CSRS1) **with** entities member **and** without watchedAttributes member **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Context Source Registration Request **from** the client **with** information member **containing** entities member matching CSRS1 **and** propertyNames **and** relationshipNames members **}**	SUT ← Client
	then { **the SUT sends to** the endpoint URI mentioned in CSRS1, a CsourceNotification **containing** type **set to "**ContextSource Notification" **and** subscriptionId **set to** id **of** CSRS1 **and** notifiedAt **a** timestamp **and** data **set to a** list **containing** CSR1 **and** triggerReason **set to "**newlyMatching" **}**	SUT → Client

TP Id	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 V1.3.1 [1], clause 5.11.7	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_5_11_7	
Initial conditions	**with {** **the SUT containing** a Context Source Registration Subscription (CSRS1) **with** entities member **and** geoQ member **containing** geoproperty member **set to** location **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Context Source Registration Request **from** the client **with** information **and** location members matching CSRS1 **}**	SUT ← Client

	then { **the SUT sends to** the endpoint URI mentioned in CSRS1, a CsourceNotification **containing** type **set to "**ContextSource Notification" **and** subscriptionId **set to** id **of** CSRS1 **and** notifiedAt **a** timestamp **and** data **set to a** list **containing** CSR1 **and** triggerReason **set to "**newlyMatching" **}**	SUT → Client

TP Id	TP/NGSI-LD/CS/REGSUB/047_15	
Test objective	Check if a context source registrations subscription does not define a geoproperty in the geoQ member, a CsourceNotification will be triggered from matching context source registrations with a matching location member	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.11.7	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_5_11_7	
Initial conditions	**with {** **the SUT containing** a Context Source Registration Subscription (CSRS1) **with** entities member **and** geoQ member **not containing** geoproperty member **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Create Context Source Registration Request **from** the client **with** information **and** location members matching CSRS1 **}**	SUT ← Client
	then { **the SUT sends to** the endpoint URI mentioned in CSRS1, a CsourceNotification **containing** type **set to "**ContextSource Notification" **and** subscriptionId **set to** id **of** CSRS1 **and** notifiedAt **a** timestamp **and** data **set to a** list **containing** CSR1 **and** triggerReason **set to "**newlyMatching" **}**	SUT → Client

TP Id	TP/NGSI-LD/CS/REGSUB/047_16	
Test objective	Check if you update a context source registration subscription, a CsourceNotification will be sent with all currently matching context source registrations	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.11.7	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_5_11_7	
Initial conditions	**with {** **the SUT containing** Context Source Registrations (CSR1) **and** (CSR2) providing respectively latest information about entities of type T **and** U a Context Source Registration Subscription (CSRS1) **with** entities members matching entities **of** type Z **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Update Context Source Registration Subscription Request **from** the client **to** update entities member **of** CSRS1 **to** match entities of type **${type}** **}**	SUT ← Client
	then { **the SUT sends to** the endpoint URI mentioned in CSRS1, a CsourceNotification **containing** type **set to "**ContextSource Notification" **and** subscriptionId **set to** id **of** CSRS1 **and** notifiedAt **a** timestamp **and** data **set to a** list **containing ${CsourceRegistrations} and** triggerReason **set to** "newlyMatching" **}**	SUT → Client

Permutation on TP Id	${type}	${CsourceRegistrations}
TP/NGSI-LD/CB/REGSUB/047_16_01	T	CSR1
TP/NGSI-LD/CB/REGSUB/047_16_02	U	CSR2
TP/NGSI-LD/CB/REGSUB/047_16_03	T and U	CSR1 and CSR2

4.2.3 Discovery

4.2.3.1 Retrieve Context Source Registration

TP Id	TP/NGSI-LD/CS/DISC/036_01
Test objective	Check that you cannot a retrieve Context Source Registration, if the context source registration id is not present or it is not a valid URI.

Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.10.1	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_5_10_1	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a retrieve Context Source Registration request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceRegistration/${id} **and** method **set to** GET **}**	SUT ← Client
	then { **the SUT sends** a Response **containing** Response Status Code **set to** BadRequestData **}**	SUT → Client

Permutation on TP Id	**${id}**
TP/NGSI-LD/CS/DISC/036_01_01	Not present
TP/NGSI-LD/CS/DISC/036_01_02	Is not a valid URI

TP Id	TP/NGSI-LD/CS/DISC/036_02	
Test objective	Check that you 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.	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.10.1	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_5_10_1	
Initial conditions	**with {** **the SUT** being in the "initial state" and **the SUT** does not contain a Context Source Registration with id equal to ${id} }	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a retrieve Context Source Registration request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceRegistration/${id} **and** method **set to** GET **}**	SUT ← Client

	then { **the SUT sends** a Response **containing** Response Status Code **set to** ResourceNotFound **}**	SUT → Client

TP Id	TP/NGSI-LD/CS/DISC/036_03	
Test objective	Check that you can retrieve a Context Source Registration. Term to URI expansion of Attribute names shall be observed.	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.10.1	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_5_10_1	
Initial conditions	**with {** **the SUT** being in the "initial state" and **the SUT** contain a Context Source Registration with id equal to ${id} **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a retrieve Context Source Registration request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceRegistration/${id} **and** method **set to** GET **}**	SUT ← Client
	then { **the SUT sends** a Response **containing** Response Status Code **set to** 200 (OK) **and** **body set to** Context Source Registration **}**	SUT → Client

TP Id	TP/NGSI-LD/CS/DISC/036_04
Test objective	Check that you can retrieve a Context Source Registration
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.10.1
Config Id	CF_05
Parent Release	V1.3.1
PICS Selection	PICS_5_10_1
Initial conditions	**with {** **the SUT** being in the "initial state" and **the SUT** contains a Context Source Registration with id equald to ${id} **}**

Expected behaviour	Test events	Direction
	when { **the SUT receives** a retrieve Context Source Registration request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceRegistration/${id} **and** method **set to** GET **}**	SUT ← Client
	then { **the SUT sends** a Response **containing** body **set** to Context Source Registration **}**	SUT → Client

TP Id	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 V1.3.1 [1], clause 6.3.5	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_6_3_5	
Initial conditions	**with {** **the SUT** being in the "initial state" and **the SUT** containing a Context Source Registration with id equal to ${csrId} **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a valid Retrieve Context Source Registration request **from** the client **containing** URL **set to** /ngsi-ld/v1/cSourceRegistration/**${csrId} and** method **set to** GET **and** Header: Link **set to** ${jsonld_context} **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 200 (OK) **and** Response Body **containing** ${csr_representation} **}**	SUT → Client

Permutation on TP Id	${jsonld_context}	${csr_representation}
TP/NGSI-LD/CS/DISC/036_05_01	empty	Context source registration with attributes from the context provided at creation time not compacted
TP/NGSI-LD/CS/DISC/036_05_02	Context containing the terms used at context source registration creation	Context source registration with attributes from the context provided at creation time compacted

4.2.3.2 Query context source registrations

TP Id	TP/NGSI-LD/CS/DISC/037_01	
Test objective	Check that you 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 V1.3.1 [1], clause 5.10.2	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_5_10_2	
Initial conditions	**with {** **the SUT** being in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a query Context Source Registration request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceRegistrations **and** method **set to** Get **and** query params **set to** Context Source Registration Query **containing** at least one of ${list} **}**	SUT ← Client
	then { **the SUT sends** a Response **containing** Response Status Code **set to** Ok **and** **body set to** list of all matching Context Source Registrations **}**	SUT → Client

Permutation on TP Id	${list}
TP/NGSI-LD/CS/DISC/037_01_01	List of Entity Types
TP/NGSI-LD/CS/DISC/037_01_02	List of Attributes names

TP Id	TP/NGSI-LD/CS/DISC/037_02
Test objective	Check that you cannot query context source registrations, if neither Entity types nor Attribute names are provided, an error of type.
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.10.2
Config Id	CF_05
Parent Release	V1.3.1

PICS Selection	PICS_5_10_2	
Initial conditions	**with {** **the SUT** being in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a query Context Source Registration request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceRegistrations **and** method **set to** Get **and** query params **set to** Context Source Registration Query **not containing** neither Entity types nor Attribute **}**	SUT ← Client
	then { **the SUT sends** a Response **containing** Response Status Code **set to** BadRequestData **}**	SUT → Client

TP Id	TP/NGSI-LD/CS/DISC/037_03	
Test objective	Check that you 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 V1.3.1 [1], clause 5.10.2	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_5_10_2	
Initial conditions	**with {** **the SUT** being in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a query Context Source Registration request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceRegistrations **and** method **set to** Get **and** query params **set to** Context Source Registration Query **not containing** list of Entity identifiers **containing** ${value} **}**	SUT ← Client
	then { **the SUT sends** a Response **containing** Response Status Code **set to** BadRequestData	SUT → Client

	}	

Permutation on TP Id	**${value}**
TP/NGSI-LD/CS/DISC/037_03_01	Not valid URI
TP/NGSI-LD/CS/DISC/037_03_02	Not syntactically valid query
TP/NGSI-LD/CS/DISC/037_03_03	Not syntactically valid geo-query
TP/NGSI-LD/CS/DISC/037_03_04	Not syntactically valid temporal query

TP Id	TP/NGSI-LD/CS/DISC/037_04	
Test objective	Check that you can query context source registrations. If a JSON-LD context is not provided, then all the query terms shall be resolved against the default JSON-LD @context.	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.10.2	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_5_10_2	
Initial conditions	**with {** **the SUT** being in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a query Context Source Registration request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceRegistrations **and** method **set to** Get **and** query params **set to** Context Source Registration Query **not containing** JSON-LD context **}**	SUT ← Client
	then { **the SUT sends** a Response **containing** Response Status Code **set to** 200 (OK) **and** **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_05
Test objective	Check that you can query context source registrations. If present, the entity specification in the query consisting of a combination of entity type and entity id/entity id pattern matches an EntityInfo specified in a RegistrationInfo of the information property in a context source registration. If there is no EntityInfo specified in the RegistrationInfo, the entity specification is considered matching. If there is no EntityInfo specified in the RegistrationInfo, the entity specification is considered matching.
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.10.2
Config Id	CF_05
Parent Release	V1.3.1
PICS Selection	PICS_5_10_2

Initial conditions	**with {** **the SUT** being in the "initial state", **the SUT containing** a Context Source Registration ${csourceRegistration} **containing** information property **containing** Registration Info **containing** EntityInfo **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a query Context Source Registration request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceRegistrations **and** method **set to** Get and query parameters **set to** Context Source Registration Query **containing** entity specification **set to** ${entitySpec} **}**	SUT ← Client
	then { **the SUT sends** a Response **containing** Response Status Code **set to** 200 (OK) **and** **body set to** list **containing** ${csourceRegistration} **}**	SUT → Client

Permutation on TP Id	${entityInfo}
TP/NGSI-LD/CS/DISC/037_05_01	EntityInfo **matching** Entity Specification ${entitySpec}
TP/NGSI-LD/CS/DISC/037_05_02	empty

TP Id	TP/NGSI-LD/CS/DISC/037_06	
Test objective	Check that you can query context source registrations. If present, at least one Attribute name specified in the query matches one Property or Relationship in the RegistrationInfo element of the information property in a context source registration. If no Properties or Relationships are specified in the RegistrationInfo, the Attribute names are considered matching.	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.10.2	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_5_10_2	
Initial conditions	**with {** **the SUT** being in the "initial state", **the SUT containing** a Context Source Registration ${csourceRegistration} **containing** **RegistrationInfo** ${regInfo} containing Properties or Relationships ${proprel} **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a query Context Source Registration request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceRegistrations **and** method **set to** Get and query params **set to** Context Source Registration Query **containing** Attribute name **set to** ${attName} **}**	SUT ← Client

	then { **the SUT sends** a Response **containing** Response Status Code **set to** 200 (OK) **and** **body set to** list **containing** ${csourceRegistration} **}**	SUT → Client

Permutation on TP Id	${attName}
TP/NGSI-LD/CS/DISC/037_06_01	Properties or Relationships matching ${attName}
TP/NGSI-LD/CS/DISC/037_06_02	Empty

TP Id	TP/NGSI-LD/CS/DISC/037_07	
Test objective	Check that you can query context source registrations. If present, the geoquery is matched against the GeoProperty identified in the geoquery. The geoquery matches the GeoProperty specified in the Context Source Registration, if the location directly matches or if the location possibly contains locations that would match the geoquery	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.10.2	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_5_10_2	
Initial conditions	**with {** **the SUT** being in the "initial state", **the SUT containing** a Context Source Registration ${csourceRegistration} **containing** GeoProperty **containing** georel **set to** ${georel} **and** geometry **set to** ${geometry} **and** coordinates **set to** ${coordinates} **and** geoproperty **set to** ${geoproperty} **and** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a query Context Source Registration request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceRegistrations **and** method **set to** Get **and** query params **set to** Context Source Registration geoquery **containing** Query Parameter: georel **set to ${georel} and** Query Parameter: geometry **set to ${geometry} and** Query Parameter: coordinates **set to ${coordinates} and** Query Parameter: geoproperty **set to ${geoproperty}** **}**	SUT ← Client
	then { **the SUT sends** a Response **containing** Response Status Code **set to** 200 (OK) **and** **body set to** list **containing** ${csourceRegistration} **}**	SUT → Client

Permutation on TP Id	${georel}	${geometry}	${coordinates}	${geoproperty}
TP/NGSI-LD/CS/DISC/037_07_01	near;maxDistance==2000	Point	[-8.503,41.202]	Not present
TP/NGSI-LD/CS/DISC/037_07_02	within	Polygon	[[-13.503,47.202], [6.541, 52.961], [20.37,44.653], [9.46,32.57], [-15.23,21.37]]	location

TP Id	TP/NGSI-LD/CS/DISC/037_08	
Test objective	Check that you 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 V1.3.1 [1], clause 5.10.2	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_5_10_2	
Initial conditions	**with {** **the SUT** being in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a query Context Source Registration request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceRegistrations **and** method **set to** Get **and** query params **set to** Context Source Registration temporal query **not containing** temporal Query **}**	SUT ← Client
	then { **the SUT sends** a Response **containing** Response Status Code **set to** 200 (OK) **and** **body set to** list **containing** matching Context Source Registrations for Context Sources providing latest information **}**	SUT → Client

TP Id	TP/NGSI-LD/CS/DISC/037_09
Test objective	Check that you can query context source registrations. If a temporal query is present, only Context Source Registrations with specified time intervals, i.e. observationInterval or managementInterval are considered. If the timeproperty is observedAt or no timeproperty is specified in the temporal query (default: observedAt), the temporal query is matched against the observationInterval (if present). If the timeproperty is createdAt or modifiedAt, the temporal query is matched against the managementInterval (if present).If the relevant interval is not present, there is no match.

Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.10.2	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_5_10_2	
Initial conditions	**with {** **the SUT** being in the "initial state" **and** **the SUT** containing a Context Source Registration **having** ${timeprop} before **${time}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a query Context Source Registration request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceRegistrations **and** method **set to** Get **and** query params **set to** Context Source Registration temporal query **containing** Query Parameter**: timeproperty set to ${timeprop}** Query Parameter: timerel **set to ${timerel} and** Query Parameter: timeAt **set to ${time}** **}**	SUT ← Client
	then { **the SUT sends** a Response **containing** Response Status Code **set to** 200 (OK) **and** **body set to** list **containing** Context Source Registrations matched against **${matchingProp}** **}**	SUT → Client

Permutation on TP Id	${timeprop}	${timerel}	${time}	${matchingprop}
TP/NGSI-LD/CS/DISC/037_09_01	ObservedAt	before	2017-12-13T14:20:00Z	observationInterval
TP/NGSI-LD/CS/DICS/037_09_02	Not present	before	2017-12-13T14:20:00Z	observationInterval
TP/NGSI-LD/CS/DISC/037_09_03	createdAt	before	2017-12-13T14:20:00Z	managementInterval
TP/NGSI-LD/CS/DISC/037_09_04	modifiedAt	before	2017-12-13T14:20:00Z	managementInterval

TP Id	TP/NGSI-LD/CS/DISC/037_10
Test objective	Check that you 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 V1.3.1 [1], clause 5.10.2
Config Id	CF_05
Parent Release	V1.3.1
PICS Selection	PICS_5_10_2
Initial conditions	**with {** **the SUT containing** three Context Source Registrations **}**

Expected behaviour	Test events	Direction
	when { **the SUT receives** a query Context Source Registration request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceRegistrations **and** Method **set to** Get **and** Query Parameter: **${param} set to ${value}** **}**	SUT ← Client
	then { **the SUT sends** a Response **containing** Response Status Code **set to** 200 (OK) **and** **body set to** list **containing** matching Context Source Registrations **}**	SUT → Client

Permutation on TP Id	${param}	${value}
TP/NGSI-LD/CS/DISC/037_10_01	id	List of Entity Ids
TP/NGSI-LD/CS/DISC/037_10_02	q	NGSI-LD Query
TP/NGSI-LD/CS/DISC/037_10_03	csf	Context Source filter

TP Id	TP/NGSI-LD/CS/DISC/037_11	
Test objective	Check that you can query context source registrations with providing page and limit parameters, pagination logic shall be in place as mandated by clause 5.5.9.	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 5.10.2	
Config Id	CF_05	
Parent Release	V1.3.1	
PICS Selection	PICS_5_10_2	
Initial conditions	**with {** **the SUT** being in the "initial state" and **containing** a Context Source Registration (CSR1) **with** an id **set to ${id1}** **and** a Context Source Registration (CSR2) **with** an id **set to ${id2}** **and** a Context Source Registration (CSR3) **with** an id **set to ${id3}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a query Context Source Registration request **from** the client **containing** URL **set to** /ngsi-ld/v1/csourceRegistrations **and** method **set to** Get **and** query params **set to** Context Source Registration query **containing** Query Parameter limit **set to ${limit} and** Query Parameter page **set to ${page}** **}**	SUT ← Client

	then { **the SUT sends** a Response **containing** Response Status Code **set to** 200 (OK) **and** **body set to** list **containing** matching Context Source Registrations respecting Pagination logic **}**	SUT → Client

Permutation on TP Id	${limit}	${page}
TP/NGSI-LD/CS/DISC/037_11_01	1	2
TP/NGSI-LD/CS/DISc/037_11_02	2	2
TP/NGSI-LD/CS/DISC/037_11_03	15	1

4.2.4 Common Behaviours

4.2.4.1 NGSI-LD API common behaviours

TP Id	TP/NGSI-LD/CB/043	
Test objective	Verify throwing 503 – LDContextNotAvaliable error if remote JSON-LD @context cannot be retrieved	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 6.3.2	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_6_3_2	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a Request **from** the client **containing** URL **set to** /ngsi-ld/v1/**${endpoint} and** method **set to** POST **and** Header: Content-Type **set to** application/ld+json **and** body **set to** JSON-LD object **where** @context is not available	SUT ← Client

	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 503 **and** Response Body **containing** ProblemDetails element **containing** type element **set to** https://uri.etsi.org/ngsi-ld/errors/LDContextNotAvailable **and** title element **containing** more information about the error **}**	SUT → Client

Permutation on TP Id	${endpoint}
TP/NGSI-LD/CB/043_01	/entities/
TP/NGSI-LD/CB/043_02	/csourceSubscriptions/
TP/NGSI-LD/CB/043_03	/temporal/entities/
TP/NGSI-LD/CB/043_04	/entityOperations/create
TP/NGSI-LD/CB/043_05	/csourceRegistrations/

4.2.4.2 API HTTP binding common behaviours

4.2.4.2.1 HTTP request pre-conditions

TP Id	TP/NGSI-LD/HTTP/044_01	
Test objective	Verify that PATCH HTTP requests can be done with "application/merge-patch+json" as Content-Type	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 6.3.4	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_6_3_4	
Initial conditions	**with {** **the SUT being** in the "initial state" and an **${object}** already exists **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a Request **from** the client **containing** URL **set to** /ngsi-ld/v1/**${endpoint} and** method **set to** PATCH **and** Header: Content-Type **set to** application/merge-patch+json **and** body **set to** JSON object	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 204 (No Content) **}**	SUT → Client

Permutation on TP Id	${object}	${endpoint}
TP/NGSI-LD/CB/044_01_01	entity	/entities/{entityId}/attrs/{attrId}
TP/NGSI-LD/CB/044_01_02	subscription	/subscriptions/{subscriptionId}

TP Id	TP/NGSI-LD/HTTP/044_02	
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 V1.3.1 [1], clause 6.3.4	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_6_3_4	
Initial conditions	**with {** **the SUT being** in the "initial state" and an **${object}** already exists **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a Request **from** the client **containing** URL **set to** /ngsi-ld/v1/**${endpoint} and** method **set to** GET	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 200 **and** Response Body **containing** queried ${object} in json format **}**	SUT → Client

Permutation on TP Id	${object}	${endpoint}
TP/NGSI-LD/CB/044_02_01	entity	/entities/{entityId}
TP/NGSI-LD/CB/044_02_02	subscription	/subscriptions/{subscriptionId}
TP/NGSI-LD/CB/044_02_03	csource registration	/csourceRegistrations/
TP/NGSI-LD/CB/044_02_04	temporal entity	/temporal/entities

TP Id	TP/NGSI-LD/HTTP/044_03
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 V1.3.1 [1], clause 6.3.4
Config Id	CF_01
Parent Release	V1.3.1
PICS Selection	PICS_6_3_4
Initial conditions	**with {** **the SUT being** in the "initial state" **}**

Expected behaviour	Test events	Direction
	when { **the SUT receives** a Request **from** the client **containing** URL **set to** /ngsi-ld/v1/**${endpoint} and** method **set to ${method} and** Header: Content-Type **set to** application/xml **and** body **set to** Ld+json object	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 415 **and** Response Body **containing** ProblemDetails element **containing** type element **set to** Unsupported Media Type **and** title element **containing** more information about the error **}**	SUT → Client

Permutation on TP Id	${method}	${endpoint}
TP/NGSI-LD/CB/044_03_01	PATCH	/entities/{entityId}/attrs/{attrId}
TP/NGSI-LD/CB/044_03_02	PATCH	/subscriptions/{subscriptionId}
TP/NGSI-LD/CB/044_03_03	POST	/entities/
TP/NGSI-LD/CB/044_03_04	POST	/entities/{entityId}/attrs/
TP/NGSI-LD/CB/044_03_05	POST	/subscriptions/
TP/NGSI-LD/CB/044_03_06	POST	/entityOperations/create

TP Id	TP/NGSI-LD/HTTP/044_04	
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 V1.3.1 [1], clause 6.3.4	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_ 6_3_4	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a Request **from** the client **containing** URL **set to** /ngsi-ld/v1/**${endpoint} and** method **set to** GET **and** Header: Accept **set to** application/xml	SUT ← Client

	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 415 **and** Response Body **containing** List of the available representations of the resources **}**	SUT → Client

Permutation on TP Id	${endpoint}
TP/NGSI-LD/CB/044_04_01	/entities/{entityId}
TP/NGSI-LD/CB/044_04_02	/subscriptions/{subscriptionId}
TP/NGSI-LD/CB/044_04_03	/csourceRegistrations/
TP/NGSI-LD/CB/044_04_04	/csourceSubscriptions/
TP/NGSI-LD/CB/044_04_05	/temporal/entities

TP Id	TP/NGSI-LD/HTTP/044_05	
Test objective	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".	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 6.3.4	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_6_3_4	
Initial conditions	**with {** **the SUT being** in the "initial state" **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT receives** a Request **from** the client **containing** URL **set to** /ngsi-ld/v1/**${endpoint} and** method **set to** POST **and** Header: Accept **set to** application/geo+json	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 406 **and** Response Body **containing** List of the available representations of the resources **}**	SUT → Client

Permutation on TP Id	${endpoint}
TP/NGSI-LD/CB/044_05_01	/entities/
TP/NGSI-LD/CB/044_05_02	/entities/{entityId}/attrs/{attrId}
TP/NGSI-LD/CB/044_05_03	/subscriptions/
TP/NGSI-LD/CB/044_05_04	/subscriptions/{subscriptionId}
TP/NGSI-LD/CB/044_05_05	/temporal/entities/
TP/NGSI-LD/CB/044_05_06	/entityOperations/create
TP/NGSI-LD/CB/044_05_07	/csourceRegistrations/

4.2.4.2.2 JSON-LD @context resolution

TP Id	TP/NGSI-LD/CB/HTTP/045_01	
Test objective	If the request verb is GET or DELETE, then the associated JSON-LD "@context" shall be obtained from a Link header as mandated by JSON-LD.	
Reference	ETSI GS CIM 009 V1.3.1 [1], clause 6.3.5	
Config Id	CF_01	
Parent Release	V1.3.1	
PICS Selection	PICS_6_3_5	
Initial conditions	**with {** **the SUT being** in the "initial state" with an already created **${resource}** **}**	
Expected behaviour	**Test events**	**Direction**
	when { **the SUT sends** a valid **${get_or_delete_operation} from** the client **containing** Header: Link **containing** a valid link to a JSON-LD @context **}**	SUT ← Client
	then { **the SUT sends** a valid Response **containing** Response Status Code **set to** 201 (CREATED) **and** A cSourceNotification shall be sent with the matching Context Source Registration **}**	SUT → Client

Permutation on TP Id	${resource}	${get_or_delete_operation}	${coordinates}	${geoproperty}
TP/NGSI-LD/CB/HTTP/045_01_01	near;maxDistance==2000	Point	[8,40]	Not present
TP/NGSI-LD/CB/HTTP/045_01_02	within	Polygon	[[100.0,0.0], [101.0,0.0], [101.0,1.0], [100.0,1.0], [100.0,0.0]]	location

Annex A (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, 23rd 2021	V1.0.2	
April 28th 2021	V1.1.1	Last Technical Officer review for Publication pre-processing with ETSI EditHelp

History

Document history		
V1.1.1	May 2021	Publication