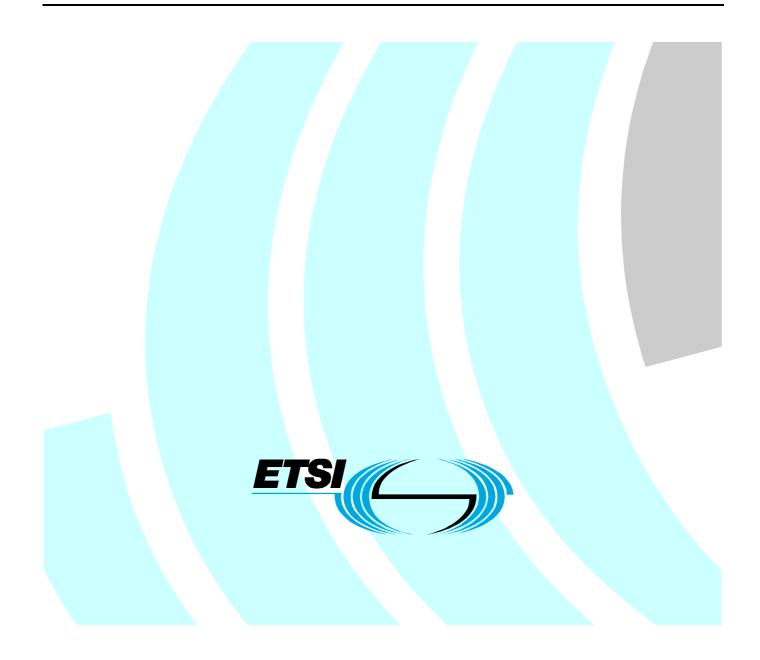
## Final draft ETSI ES 202 196-6 V1.2.1 (2004-12)

ETSI Standard

Open Service Access (OSA); Application Programming Interface (API); Test Suite Structure and Test Purposes (TSS&TP); Part 6: Mobility SCF (Parlay 3)



Reference RES/TISPAN-06005-06-OSA

2

Keywords API, OSA, TSS&TP

#### 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 - NAF 742 C Association à but non lucratif enregistrée à la Sous-Préfecture de Grasse (06) N° 7803/88

#### Important notice

Individual copies of the present document can be downloaded from: http://www.etsi.org

The present document may be made available in more than one electronic version or in print. In any case of existing or perceived difference in contents between such versions, the reference version is the Portable Document Format (PDF). In case of dispute, the reference shall be the printing on ETSI printers of the PDF version kept on a specific network drive within ETSI Secretariat.

Users of the present document should be aware that the document may be subject to revision or change of status. Information on the current status of this and other ETSI documents is available at <a href="http://portal.etsi.org/tb/status/status.asp">http://portal.etsi.org/tb/status/status.asp</a>

If you find errors in the present document, please send your comment to one of the following services: http://portal.etsi.org/chaircor/ETSI\_support.asp

#### **Copyright Notification**

No part may be reproduced except as authorized by written permission. The copyright and the foregoing restriction extend to reproduction in all media.

> © European Telecommunications Standards Institute 2004. All rights reserved.

**DECT**<sup>TM</sup>, **PLUGTESTS**<sup>TM</sup> and **UMTS**<sup>TM</sup> are Trade Marks of ETSI registered for the benefit of its Members. **TIPHON**<sup>TM</sup> and the **TIPHON logo** are Trade Marks currently being registered by ETSI for the benefit of its Members. **3GPP**<sup>TM</sup> is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners.

## Contents

Intelle	ectual Property Rights	4
Forew	vord	4
1	Scope	5
2	References	5
3 3.1 3.2	Definitions and abbreviations Definitions Abbreviations	5
4	Test Suite Structure (TSS)	6
5	Test Purposes (TP)	
5.1 5.1.1	Introduction TP naming convention	7
5.1.2 5.1.3	Source of TP definition Test strategy	
5.2 5.2.1	TPs for the Mobility SCF User Location	
5.2.1.1	I IpUserLocation interface	8
5.2.1.2 5.2.2	2 IpTrigerredUserLocation interface User Location Camel	
5.2.3 5.2.4	User Location Emergency User status	51
5.3 5.3.1	TPs for the application using the Mobility SCF User Location	
5.3.1.1	I IpAppUserLocation interface	56
5.3.1.2 5.3.2	2 IpAppTrigerredUserLocation interface User Location Camel	
5.3.3 5.3.4	User Location Emergency User Status	
Histor	ry	90

3

## Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is 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 (http://webapp.etsi.org/IPR/home.asp).

4

Pursuant to the ETSI IPR Policy, no investigation, 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.

## Foreword

This ETSI Standard (ES) has been produced by ETSI Technical Committee Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN), and is now submitted for the ETSI standards Membership Approval Procedure.

The present document is part 6 of a multi-part deliverable covering Open Service Access (OSA); Application Programming Interface (API); Test Suite Structure and Test Purposes (TSS&TP) for Parlay 3, as identified below:

- Part 1: "Overview";
- Part 2: "Common data definitions";
- Part 3: "Framework";
- Part 4: "Call control SCF";
- Part 5: "User interaction SCF";
- Part 6: "Mobility SCF";
- Part 7: "Terminal capabilities SCF";
- Part 8: "Data session control SCF";
- Part 9: "Generic messaging SCF";
- Part 10: "Connectivity manager SCF";
- Part 11: "Account management SCF";
- Part 12: "Charging SCF".

To evaluate conformance of a particular implementation, it is necessary to have a set of test purposes to evaluate the dynamic behaviour of the Implementation Under Test (IUT). The specification containing those test purposes is called a Test Suite Structure and Test Purposes (TSS&TP) specification.

## 1 Scope

The present document provides the Test Suite Structure and Test Purposes (TSS&TP) specification for the Mobility SCF of the Application Programming Interface (API) for Open Service Access (OSA) defined in ES 201 915-6 [1] in compliance with the relevant requirements, and in accordance with the relevant guidance given in ISO/IEC 9646-2 [4] and ETS 300 406 [5].

## 2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication and/or edition number or version number) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies.

Referenced documents which are not found to be publicly available in the expected location might be found at <a href="http://docbox.etsi.org/Reference">http://docbox.etsi.org/Reference</a>.

- [1] ETSI ES 201 915-6: "Open Service Access (OSA); Application Programming Interface (API); Part 6: Mobility SCF (Parlay 3)".
- [2] ETSI ES 202 170: "Open Service Access (OSA); Application Programming Interface (API); Implementation Conformance Statement (ICS) proforma specification (Parlay 3)".
- [3] ISO/IEC 9646-1: "Information technology Open Systems Interconnection Conformance testing methodology and framework Part 1: General concepts".
- [4] ISO/IEC 9646-2: "Information technology Open Systems Interconnection Conformance testing methodology and framework Part 2: Abstract Test Suite specification".
- [5] ETSI ETS 300 406: "Methods for testing and Specification (MTS); Protocol and profile conformance testing specifications; Standardization methodology".

## 3 Definitions and abbreviations

## 3.1 Definitions

For the purposes of the present document, the terms and definitions given in ES 201 915-6 [1], ISO/IEC 9646-1 [3], ISO/IEC 9646-2 [4] and the following apply:

abstract test case: Refer to ISO/IEC 9646-1 [3].

Abstract Test Method (ATM): Refer to ISO/IEC 9646-1 [3].

Abstract Test Suite (ATS): Refer to ISO/IEC 9646-1 [3].

ICS proforma: Refer to ISO/IEC 9646-1 [3].

Implementation Conformance Statement (ICS): Refer to ISO/IEC 9646-1 [3].

Implementation eXtra Information for Testing (IXIT): Refer to ISO/IEC 9646-1 [3].

Implementation Under Test (IUT): Refer to ISO/IEC 9646-1 [3].

IXIT proforma: Refer to ISO/IEC 9646-1 [3].

Lower Tester (LT): Refer to ISO/IEC 9646-1 [3].

Test Purpose (TP): Refer to ISO/IEC 9646-1 [3].

## 3.2 Abbreviations

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

API	Application Programming Interface
ATM	Abstract Test Method
ATS	Abstract Test Suite
ICS	Implementation Conformance Statement
IUT	Implementation Under Test
IXIT	Implementation eXtra Information for Testing
LT	Lower Tester
М	Mobility
OSA	Open Service Access
TP	Test Purpose
TSS	Test Suite Structure
ULC	User Location Camel
ULE	User Location Emergency
US	User Status

## 4 Test Suite Structure (TSS)

Mobility (M):

- User Location:
  - IpUserLocation interface (UL) (01);
  - IpTrigerredUserLocation interface (TUL) (02).
- User Location Camel (ULC) (03).
- User Location Emergency (ULE) (04).
- User Status (US) (05).

6

## 5 Test Purposes (TP)

### 5.1 Introduction

For each test requirement a TP is defined.

#### 5.1.1 TP naming convention

TPs are numbered, starting at 01, within each group. Groups are organized according to the TSS. Additional references are added to identify the actual test suite (see table 1).

7

#### Table 1: TP identifier naming convention scheme

```
      Identifier:
      <suite_id>_<group>_<nnn>

      <suite_id>
      = SCG name:
      "M" for Mobility part of Mobility SCF

      <group>
      = group number:
      two character field representing the group reference according to TSS

      <nn>
      = sequential number:
      (01 to 99)
```

#### 5.1.2 Source of TP definition

The TPs are based on ES 201 915-6 [1].

#### 5.1.3 Test strategy

As the base standard ES 201 915-6 [1] contains no explicit requirements for testing, the TPs were generated as a result of an analysis of the base standard and the PICS specification ES 202 170 [2].

The TPs are only based on conformance requirements related to the externally observable behaviour of the IUT and are limited to conceivable situations to which a real implementation is likely to be faced (see ETS 300 406 [5]).

## 5.2 TPs for the Mobility SCF

All PICS items referred to in this clause are as specified in ES 202 170 [2] unless indicated otherwise by another numbered reference.

All parameters specified in method calls are valid unless specified.

The procedures to trigger the SCF to call methods in the application are dependent on the underlying network architecture and are out of the scope of this test specification. Those method calls are preceded by the words "Triggered action".

#### 5.2.1 User Location

#### 5.2.1.1 IpUserLocation interface

#### Test M\_UL\_01

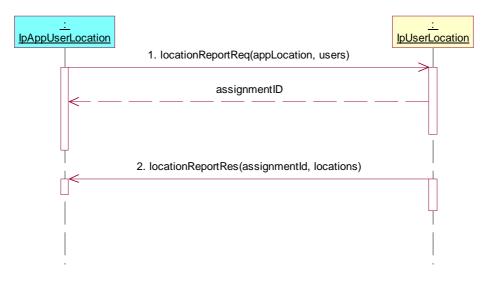
Summary: all methods, successful

Reference: ES 201 915-6 [1], clause 8.1

Selection: locationReportReq method supported - PICS item: [2] UL1

Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpUserLocation interface through selecting that service and signing the required service agreement.

- 1. Method call **locationReportReq**() Parameters: appLocation, users Check: valid value of TpSessionID is returned
- Triggered action: cause IUT to call locationReportRes() method on the tester's (Application) IpAppUserLocation interface.
   Parameters: assignmentId, locations



Summary:	locationReportReq, locationReportErr
Reference:	ES 201 915-6 [1], clause 8.1
Selection:	locationReportReq method supported - PICS item: [2] UL1
Preamble:	Registration of the IUT (User Location SCF) and the tester (appli- must have obtained a reference to an instance of the IpUserLocati

reamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpUserLocation interface through selecting that service and signing the required service agreement.

Test Sequence:

- 1. Method call **locationReportReq(**) Parameters: appLocation, users with unknown or absent subscriber Check: valid value of TpSessionID is returned
- Triggered action: cause IUT to call locationReportErr() method on the tester's (Application) IpAppUserLocation interface.
   Parameters: assignmentId, cause indicating P M UNKNOWN SUBSCRIBER or

arameters: assignmentId, cause indicating P\_M\_UNKNOWN\_SUBSCRIBER or P\_ABSENT\_SUBSCRIBER, diagnostic

<u>:</u> IpAppUserLocation	<u>:</u> IpUserLocation
1. locationReportReq(appLocation, unknown or absent users)	
assignmentID	
2. locationReportErr(assignmentId, P_M_UNKNOWN_SUBSCRIBER or P_ABSENT_SUB	BSCRIBER, diagnostic)

Summary:	all methods, successful
Reference:	ES 201 915-6 [1], clause 8.1
Selection:	extendedLocationReportReq method supported - PICS item: [2] UL2
Preamble:	Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpUserLocation interface through selecting that service and signing the required service agreement.

- 1. Method call **extendedLocationReportReq**() Parameters: appLocation, users, request Check: valid value of TpSessionID is returned
- Triggered action: cause IUT to call extendedLocationReportRes() method on the tester's (Application) IpAppUserLocation interface.
   Parameters: assignmentId, locations

<u>IpAppUse</u>	:_ erLocation		<u>IpUserL</u>	:_ .ocation
	1.	extendedLocationReportReq(appLocation, users, request)		
	<	assignmentID		
L F		2. extendedLocationReportRes(assignmentId, locations)		

Summary:	extendedLocationReportReq, extendedLocationReportErr
Reference:	ES 201 915-6 [1], clause 8.1
Selection:	extendedLocationReportReq method supported - PICS item: [2] UL2
Preamble:	Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpUserLocation interface through selecting that service and signing the required service agreement.
Test Sequence	x

11

1. Method call **extendedLocationReportReq**() Parameters: appLocation, users with unknown or absent subscriber, request Check: valid value of TpSessionID is returned

 Triggered action: cause IUT to call extendedLocationReportErr() method on the tester's (Application) IpAppUserLocation interface.
 Parameters: assignmentId, cause indicating P\_M\_UNKNOWN\_SUBSCRIBER or

arameters: assignmentId, cause indicating P\_M\_UNKNOWN\_SUBSCRIBER o P\_ABSENT\_SUBSCRIBER, diagnostic

<u>:</u> IpAppUserLocation		<u>:</u> pUserLocation
	1. extendedLocationReportReq(appLocation, unknown or absent users, request)	
	assignmentID	
€ —		
		Ť
2. e	extendedLocationReportErr(assignmentId, P_M_UNKNOWN_SUBSCRIBER or P_ABSENT_SUBSCRIBER, diagnostic)	)
		Ţ

## Summary:extendedLocationReportReq, P\_REQUESTED\_ACCURACY\_CANNOT\_BE\_DELIVEREDReference:ES 201 915-6 [1], clause 8.1

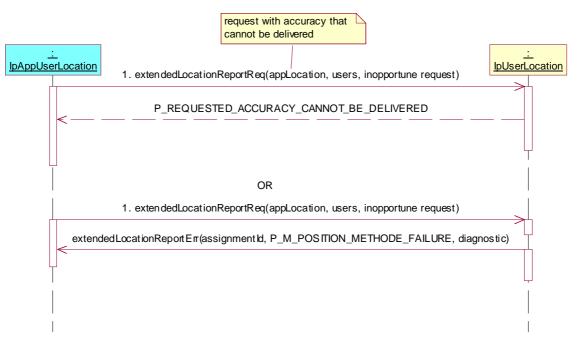
Selection: extendedLocationReportReq method supported - PICS item: [2] UL2

Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpUserLocation interface through selecting that service and signing the required service agreement.

Test Sequence:

Test M\_UL\_05

# 1. Method call extendedLocationReportReq() Parameters: appLocation, users, request with accuracy that cannot be delivered Check: P\_REQUESTED\_ACCURACY\_CANNOT\_BE\_DELIVERED is returned, or extendedLocationReportErr() with P\_M\_POSITION\_METHOD\_FAILURE value of TpMobilityError.



## Summary: extendedLocationReportReq, P\_REQUESTED\_RESPONSE\_TIME\_CANNOT\_BE\_DELIVERED

13

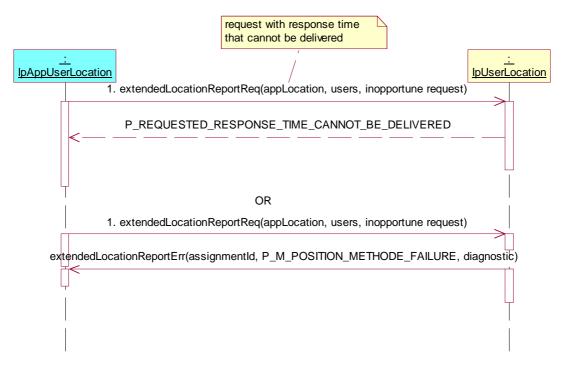
Reference: ES 201 915-6 [1], clause 8.1

Selection: extendedLocationReportReq method supported - PICS item: [2] UL2

Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpUserLocation interface through selecting that service and signing the required service agreement.

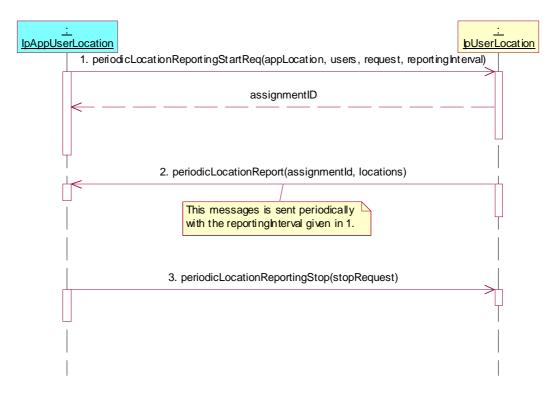
Test Sequence:

#### 1. Method call extendedLocationReportReq() Parameters: appLocation, users, request with response time that cannot be delivered Check: P\_REQUESTED\_RESPONSE\_TIME\_CANNOT\_BE\_DELIVERED is returned, or extendedLocationReportErr() with P\_M\_POSITION\_METHOD\_FAILURE value of TpMobilityError.



Summary:	all methods, successful
Reference:	ES 201 915-6 [1], clause 8.1
Selection:	periodicLocationReportingStartReq method supported - PICS item: [2] UL3
Preamble:	Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpUserLocation interface through selecting that service and signing the required service agreement.

- 1. Method call **periodicLocationReportingStartReq**() Parameters: appLocation, users, request, reportingInterval Check: valid value of TpSessionID is returned
- Triggered action: periodically cause IUT to call periodicLocationReport() method on the tester's (Application) IpAppUserLocation interface.
   Parameters: assignmentId, locations
   Check: These messages are sent with correct reporting Interval given in 1.
- 3. Method call **periodicLocationReportingStop()** Parameters: stopRequest Check: No further periodicLocationReport is sent.



Summary:	all methods, unknown or absent subscriber
Reference:	ES 201 915-6 [1], clause 8.1
Selection:	periodocLocationReportingStartReq method supported - PICS item: [2] UL3
Preamble:	Registration of the IUT (User Location SCF) and the tester (application) to the

Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpUserLocation interface through selecting that service and signing the required service agreement.

Test Sequence:

ļ

- 1. Method call **periodicLocationReportingStartReq(**) Parameters: appLocation, users with unknown or absent subscriber, request, reportingInterval Check: valid value of TpSessionID is returned
- 2. Triggered action: cause IUT to call **periodicLocationReportErr** () method on the tester's (Application) **IpAppUserLocation** interface.

Parameters: assignmentId, cause indicating P\_M\_UNKNOWN\_SUBSCRIBER or P\_ABSENT\_SUBSCRIBER, diagnostic

erLocation	Logation Paparting Start P	adopple action unknow	or chaont unora reque	nt roporting interval	IpUserL
1. pendarc	LocationReportingStatik	eq(appLocation, unknowr	Tor absent users, reques	st, reportingintervar	/
					-
		assignmentID			
					L
	Depart Fre/appianmented				
2. periodicLocation	ReportErr(assignmentId,	P_M_UNKNOWN_SUBS	SCRIBER or P_ABSENT	_SUBSCRIBER, di	agnostic)
2. periodicLocation	ReportErr(assignmentId,	P_M_UNKNOWN_SUBS	SCRIBER or P_ABSENT	_SUBSCRIBER, di	agnostic)
2. periodicLocation	ReportErr(assignmentId,	P_M_UNKNOWN_SUBS	SCRIBER or P_ABSENT	_SUBSCRIBER, di	agnostic)
2. periodicLocation	ReportErr(assignmentId,	P_M_UNKNOWN_SUBS	SCRIBER or P_ABSENT	_SUBSCRIBER, di	agnostic)

 $Summary: periodic Location Reporting Start Req, P\_REQUESTED\_ACCURACY\_CANNOT\_BE\_DELIVERED$ 

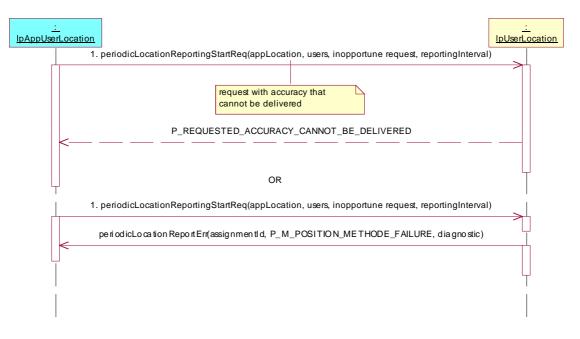
Reference: ES 201 915-6 [1], clause 8.1

Selection: periodocLocationReportingStartReq method supported - PICS item: [2] UL3

Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpUserLocation interface through selecting that service and signing the required service agreement.

Test Sequence:

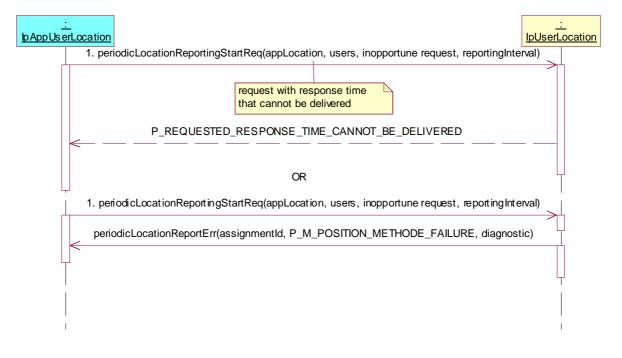
1. Method call **periodicLocationReportingStartReq** () Parameters: appLocation, users, request with accuracy that cannot be delivered, reportingInterval Check: P\_REQUESTED\_ACCURACY\_CANNOT\_BE\_DELIVERED is returned, or periodicLocationReportErr() with P\_M\_POSITION\_METHOD\_FAILURE value of TpMobilityError.



Summary:	periodicLocationReportingStartReq, P_REQUESTED_RESPONSE_TIME_CANNOT_BE_DELIVERED
Reference:	ES 201 915-6 [1], clause 8.1
Selection:	periodocLocationReportingStartReq method supported - PICS item: [2] UL3
Preamble:	Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpUserLocation interface through selecting that service and signing the required service agreement.

#### Test Sequence:

#### 1. Method call **periodicLocationReportingStartReq** () Parameters: AppLocation, users, request with response time that cannot be delivered, reportingInterval Check: P\_REQUESTED\_RESPONSE\_TIME\_CANNOT\_BE\_DELIVERED is returned, or periodicLocationReportErr() with P\_M\_POSITION\_METHOD\_FAILURE value of TpMobilityError.



Summary:	$periodicLocationReportingStartReq, P\_INVALID\_REPORTING\_INTERVAL$
Reference:	ES 201 915-6 [1], clause 8.1

Selection: periodocLocationReportingStartReq method supported - PICS item: [2] UL3

Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpUserLocation interface through selecting that service and signing the required service agreement.

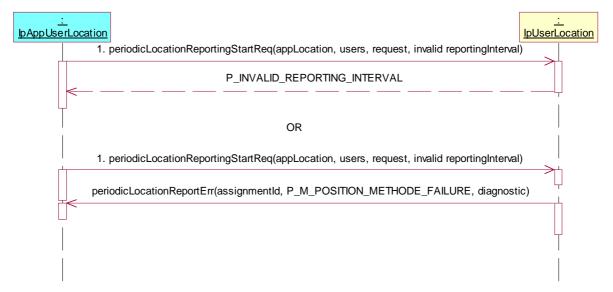
Test Sequence:

 1. Method call periodicLocationReportingStartReq ()

 Parameters:
 appLocation, users, request, invalid reportingInterval

 Check:
 P\_INVALID\_REPORTING\_INTERVAL is returned, or periodicLocationReportErr() with

 P\_M\_POSITION\_METHOD\_FAILURE value of TpMobilityError.



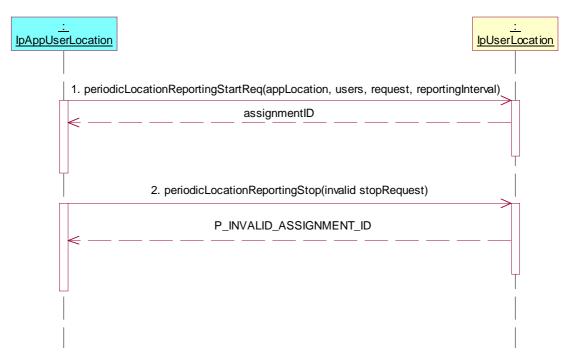
Summary:	periodicLocationReportingStop, P_INVALID_ASSIGNMENT_ID	
----------	--	--

Reference: ES 201 915-6 [1], clause 8.1

Selection: periodocLocationReportingStartReq method supported - PICS item: [2] UL3

Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpUserLocation interface through selecting that service and signing the required service agreement.

- 1. Method call **periodicLocationReportingStartReq**() Parameters: appLocation, users, request, reportingInterval Check: valid value of TpSessionID is returned
- 2. Method call **periodicLocationReportingStop**() Parameters: invalid stopRequest Check: P\_INVALID\_ASSIGNMENT\_ID, is returned.



#### 5.2.1.2 IpTrigerredUserLocation interface

#### Test M\_TUL\_01

Summary: all methods, successful

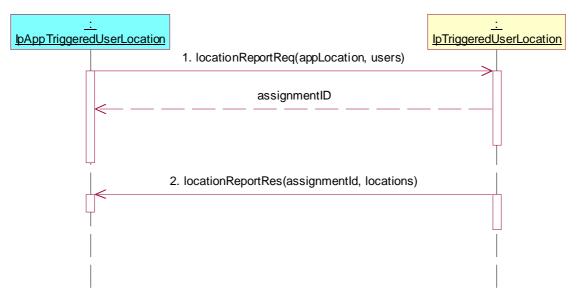
Reference: ES 201 915-6 [1], clause 8.1

Selection: locationReportReq method supported - PICS item: [2] TUL1

Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpTriggeredUserLocation interface through selecting that service and signing the required service agreement.

20

- 1. Method call **locationReportReq**() Parameters: appLocation, users Check: valid value of TpSessionID is returned
- Triggered action: cause IUT to call locationReportRes() method on the tester's (Application) IpAppTriggeredUserLocation interface. Parameters: assignmentId, locations



Summary:	locationReportReq, locationReportErr
Reference:	ES 201 915-6 [1], clause 8.1
Selection:	locationReportReq method supported - PICS item: [2] TUL1
Preamble:	Registration of the IUT (User Location SCF) and the tester (applic

Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpTriggeredUserLocation interface through selecting that service and signing the required service agreement.

Test Sequence:

- 1. Method call **locationReportReq(**) Parameters: appLocation, users with unknown or absent subscriber Check: valid value of TpSessionID is returned
- Triggered action: cause IUT to call locationReportErr() method on the tester's (Application) IpAppTriggeredUserLocation interface. Parameters: assignmentId, cause indicating P\_M\_UNKNOWN\_SUBSCRIBER or

P\_ABSENT\_SUBSCRIBER, diagnostic

<u>:</u> IpAppTriggeredUserL	ocation	<u>:</u> IpTriggeredUserLocation
	1. locationReportReq(appLocation, unknown or absent users)	
<	assignmentlD	
2. location Report	Err(assignmentld, P_M_UNKNOWN_SUBSCRIBER or P_ABSENT_SUE	BSCRIBER, diagnostic)

Summary:	all methods, successful
Reference:	ES 201 915-6 [1], clause 8.1
Selection:	extendedLocationReportReq method supported - PICS item: [2] TUL2
Preamble:	Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpTriggeredUserLocation interface through selecting that service and signing the required service agreement.

Test Sequence:

- 1. Method call **extendedLocationReportReq**() Parameters: appLocation, users, request Check: valid value of TpSessionID is returned
- Triggered action: cause IUT to call extendedLocationReportRes() method on the tester's (Application) IpAppTriggeredUserLocation interface. Parameters: assignmentId, locations

22

<u>.</u> IpAppTriggered	<u>.</u> dUserLocation			: IpTriggeredUserLocation
	1. exter	ndedLocationReportR	eq(appLocation, users, r	request)
	<	assig	nmentID	
_	2. ex	tendedLocationReport	tRes(assignmentId, loca	tions)
				Ļ

Summary:	extendedLocationReportReq, extendedLocationReportErr
Reference:	ES 201 915-6 [1], clause 8.1
Selection:	extendedLocationReportReq method supported - PICS item: [2] TUL2
Preamble:	Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpTriggeredUserLocation interface through selecting that service and signing the required service agreement.
Test Sequence	

23

- 1. Method call extendedLocationReportReq() Parameters: appLocation, users with unknown or absent subscriber, request Check: valid value of TpSessionID is returned
- Triggered action: cause IUT to call extendedLocationReportErr() method on the tester's (Application) IpAppTriggeredUserLocation interface.
   Parameters: assignmentId, cause indicating P\_M\_UNKNOWN\_SUBSCRIBER or

P\_ABSENT\_SUBSCRIBER, diagnostic

<u>:</u> IpAppTriggeredUserLocation	LipTriggeredUserLo	ocation
	1. extendedLocationReportReq(appLocation, unknown or absent users, request)	
	assignmentID	
	T	
2. extende	edLocationReportErr(assignmentId, P_M_UNKNOWN_SUBSCRIBER or P_ABSENT_SUBSCRIBER, diagnostic)	
Ţ		

Summary:extendedLocationReportReq, P\_REQUESTED\_ACCURACY\_CANNOT\_BE\_DELIVEREDReference:ES 201 915-6 [1], clause 8.1

24

Selection: extendedLocationReportReq method supported - PICS item: [2] TUL2

Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpTriggeredUserLocation interface through selecting that service and signing the required service agreement.

Test Sequence:

#### 1. Method call extendedLocationReportReq() Parameters: appLocation, users, request with accuracy that cannot be delivered Check: P\_REQUESTED\_ACCURACY\_CANNOT\_BE\_DELIVERED is returned, or extendedLocationReportErr() with P\_M\_POSITION\_METHOD\_FAILURE value of TpMobilityError.

	cannot be delivered
LipAppTriggeredUserLocation 1. extendedLocationF	ReportReq(appLocation, users, inopportune request)
< P_REQUESTE	ED_ACCURACY_CANNOT_BE_DELIVERED
	OR
	OR
1. extendedLocationF	ReportReq(appLocation, users, inopportune request)
extendedLocationReportErr(assi	ignmentId, P_M_POSITION_METHODE_FAILURE, diagnostic)
I	

 $Summary: extended Location Report Req, P\_REQUESTED\_RESPONSE\_TIME\_CANNOT\_BE\_DELIVERED$ 

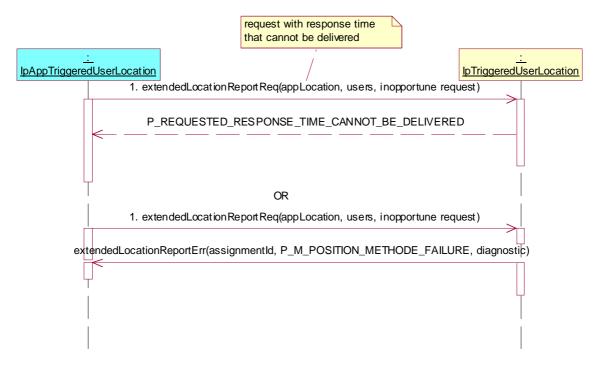
Reference: ES 201 915-6 [1], clause 8.1

Selection: extendedLocationReportReq method supported - PICS item: [2] TUL2

Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpTriggeredUserLocation interface through selecting that service and signing the required service agreement.

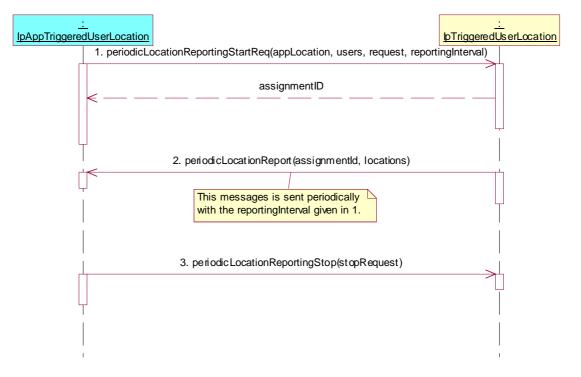
Test Sequence:

1. Method call extendedLocationReportReq() Parameters: appLocation, users, request with response time that cannot be delivered Check: P\_REQUESTED\_RESPONSE\_TIME\_CANNOT\_BE\_DELIVERED is returned, or extendedLocationReportErr() with P\_M\_POSITION\_METHOD\_FAILURE value of TpMobilityError.



Summary:	all methods, successful
Reference:	ES 201 915-6 [1], clause 8.1
Selection:	periodocLocationReportingStartReq method supported - PICS item: [2] TUL3
Preamble:	Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpTriggeredUserLocation interface through selecting that service and signing the required service agreement.

- 1. Method call **periodicLocationReportingStartReq**() Parameters: appLocation, users, request, reportingInterval Check: valid value of TpSessionID is returned
- Triggered action: periodically cause IUT to call periodicLocationReport () method on the tester's (Application) IpAppTriggeredUserLocation interface.
   Parameters: assignmentId, locations
   Check: These messages are sent with correct reporting Interval given in 1.
- 3. Method call **periodicLocationReportingStop**() Parameters: stopRequest Check: No further periodicLocationReport is sent.



Summary:	all methods,	unknown o	or absent	subscriber
----------	--------------	-----------	-----------	------------

Reference: ES 201 915-6 [1], clause 8.1

Selection: periodocLocationReportingStartReq method supported - PICS item: [2] TUL3

Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpTriggeredUserLocation interface through selecting that service and signing the required service agreement.

- 1. Method call periodicLocationReportingStartReq()

   Parameters:
   appLocation, users with unknown or absent subscriber, request, reportingInterval

   Check:
   valid value of TpSessionID is returned
- Triggered action: cause IUT to call periodicLocationReportErr () method on the tester's (Application) IpAppTriggeredUserLocation interface.
   Perameters: assignmentId cause indicating P M UNKNOWN SUBSCRIBER or

'arameters:	assignmentId, cause indicating P_M_UNKNOWN_SUBSCRIBER or
	P_ABSENT_SUBSCRIBER, diagnostic

: IpAppTriggered	UserLocation		<u>:</u> IpTriggeredUserLocation
	1. peri	odicLocationReportingStartReq(appLocation, unknown or absent users, request, reportingInter	v al)
	<	assignmentID	
Ĺ	2. periodicLoca	ationReportErr(assignmentId, P_M_UNKNOWN_SUBSCRIBER or P_ABSENT_SUBSCRIBER	, diagnostic)
			T

 $Summary: periodic Location Reporting Start Req, P\_REQUESTED\_ACCURACY\_CANNOT\_BE\_DELIVERED$ 

Reference: ES 201 915-6 [1], clause 8.1

Selection: periodocLocationReportingStartReq method supported - PICS item: [2] TUL3

Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpTriggeredUserLocation interface through selecting that service and signing the required service agreement.

Test Sequence:

1. Method call **periodicLocationReportingStartReq** () Parameters: appLocation, users, request with accuracy that cannot be delivered, reportingInterval Check: P\_REQUESTED\_ACCURACY\_CANNOT\_BE\_DELIVERED is returned, or periodicLocationReportErr() with P\_M\_POSITION\_METHOD\_FAILURE value of TpMobilityError.

<u>:</u> lpTriggeredUserLocation
gInterval)
gInterval)
nostic)

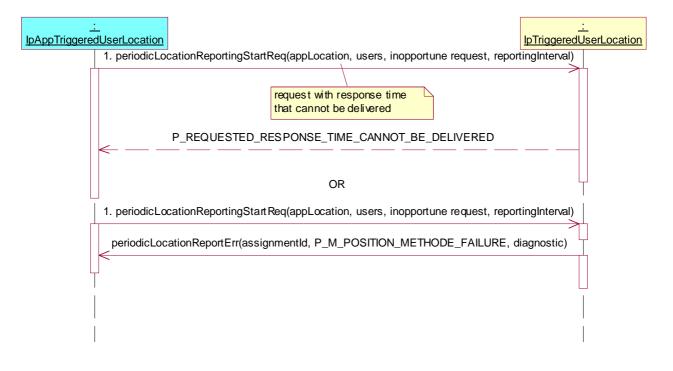
The tester

#### Test M\_TUL\_10

Summary:	periodicLocationReportingStartReq, P_REQUESTED_RESPONSE_TIME_CANNOT_BE_DELIVERED
Reference:	ES 201 915-6 [1], clause 8.1
Selection:	periodocLocationReportingStartReq method supported - PICS item: [2] TUL3
Preamble:	Registration of the IUT (User Location SCF) and the tester (application) to the framework. The term must have obtained a reference to an instance of the IpTriggeredUserLocation interface through selecting that service and signing the required service agreement.

#### Test Sequence:

#### 1. Method call **periodicLocationReportingStartReq** () Parameters: Check: P\_REQUESTED\_RESPONSE\_TIME\_CANNOT\_BE\_DELIVERED is returned, or periodicLocationReportErr() with P\_M\_POSITION\_METHOD\_FAILURE value of TpMobilityError.



Summary:	periodicLocationReportingStartReq, P_INVALID_REPORTING_INTERVAL
Reference:	ES 201 915-6 [1], clause 8.1
Selection:	periodocLocationReportingStartReq method supported - PICS item: [2] TUL3
<b>D</b> 11	

Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpTriggeredUserLocation interface through selecting that service and signing the required service agreement.

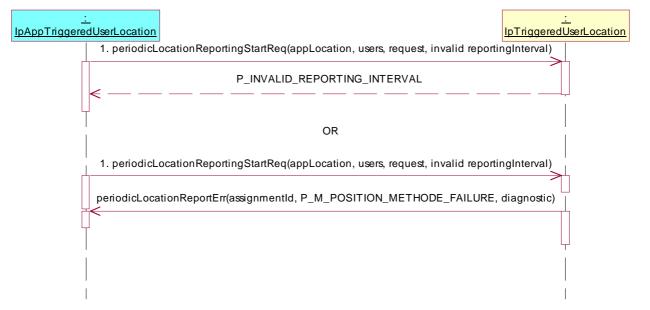
Test Sequence:

 1. Method call periodicLocationReportingStartReq ()

 Parameters:
 appLocation, users, request, invalid reportingInterval

 Check:
 P\_INVALID\_REPORTING\_INTERVAL is returned, or periodicLocationReportErr() with

 P\_M\_POSITION\_METHOD\_FAILURE value of TpMobilityError.



Summary: periodicLocationReportingStop, P\_INVALID\_ASSIGNMENT\_ID

Reference: ES 201 915-6 [1], clause 8.1

Selection: periodocLocationReportingStartReq method supported - PICS item: [2] TUL3

Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpTriggeredUserLocation interface through selecting that service and signing the required service agreement.

31

- 1. Method call periodicLocationReportingStartReq()

   Parameters:
   appLocation, users, request, reportingInterval

   Check:
   valid value of TpSessionID is returned
- 2. Method call **periodicLocationReportingStop**() Parameters: invalid stopRequest Check: P\_INVALID\_ASSIGNMENT\_ID, is returned.

<u>:</u> IpAppTriggeredUserLocation		<u>:</u> pTriggeredUserLocation
1. periodicLoc	ationReportingStartReq(appLocation, users, request, reporti	ngInterval)
<	assignmentID	
	2. periodicLocationReportingStop(invalid stopRequest)	
<	P_INVALID_ASSIGNMENT_ID	

IpTriggeredUserLocation

#### Test M\_TUL\_13

Summary:	all methods, successful
----------	-------------------------

Reference: ES 201 915-6 [1], clause 8.1

Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpTriggeredUserLocation interface through selecting that service and signing the required service agreement.

Test Sequence:

- 1. Method call **triggeredLocationReportingStartReq**() Parameters: appLocation, users, request, triggers Check: valid value of TpSessionID is returned
- Triggered action: cause IUT to call triggeredLocationReport () method on the tester's (Application) IpAppTriggeredUserLocation interface.
   Parameters: assignmentId, location, criterion
- 3. Method call triggeredLocationReportingStop() Parameters: stoprequest Check: no exception is returned

<u>IpAppTriggeredUserLocation</u>

1. triggeredLocationReportingStartReq(appLocation, users, request, triggers) assignmentID

2. triggeredLocationReport(assignmentId, location, criterion)

3. triggeredLocationReportingStop(stopRequest)

Summary: all methods, unknown or absent subscriber

Reference: ES 201 915-6 [1], clause 8.1

Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpTriggeredUserLocation interface through selecting that service and signing the required service agreement.

33

- 1. Method call **triggeredLocationReportingStartReq**() Parameters: appLocation, users with unknown or absent subscriber, request, triggers Check: valid value of TpSessionID is returned
- Triggered action: cause IUT to call triggeredLocationReportErr () method on the tester's (Application) IpAppTriggeredUserLocation interface.
   Parameters: assignmentId, cause indicating P\_M\_UNKNOWN\_SUBSCRIBER or
  - P\_ABSENT\_SUBSCRIBER, diagnostic

<u>:</u> IpAppTriggered	<u>.</u> dUsenLocation		<u>:</u> IpTriggeredU	_ IserLocation
   	1	I. triggeredLocationReportingStartReq(appLocation, unknown or absent users, request, triggers)assignmentID		
	2. triggeredLc	ocationReportErr(assignmentId, P_M_UNKNOWN_SUBSCRIBER or P_ABSENT_SUBSCRIBER,	diagnostic)	
				1

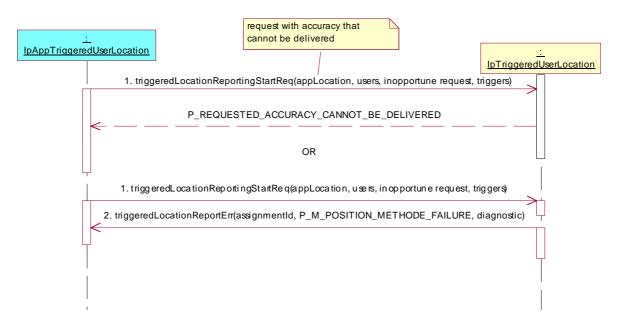
Summary: triggeredLocationReportingStartReq, P\_REQUESTED\_ACCURACY\_CANNOT\_BE\_DELIVERED

Reference: ES 201 915-6 [1], clause 8.1

Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpTriggeredUserLocation interface through selecting that service and signing the required service agreement.

Test Sequence:

1. Method call triggeredLocationReportingStartReq () Parameters: appLocation, users, request with accuracy that cannot be delivered, triggers Check: P\_REQUESTED\_ACCURACY\_CANNOT\_BE\_DELIVERED is returned, or triggeredLocationReportErr() with P\_M\_POSITION\_METHOD\_FAILURE value of TpMobilityError is returned.



Summary: triggeredLocationReportingStartReq, P\_REQUESTED\_RESPONSE\_TIME\_CANNOT\_BE\_DELIVERED

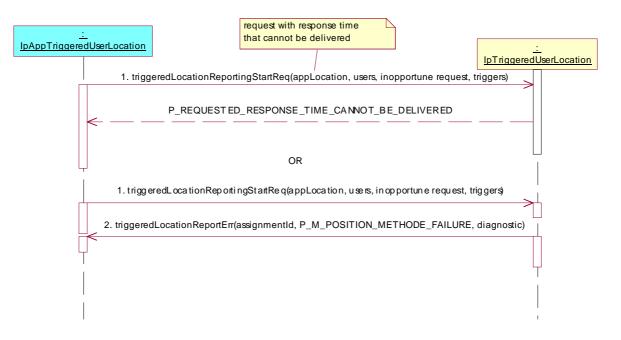
Reference: ES 201 915-6 [1], clause 8.1

Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpTriggeredUserLocation interface through selecting that service and signing the required service agreement.

35

Test Sequence:

1. Method call triggeredLocationReportingStartReq () Parameters: appLocation, users, request with response time that cannot be delivered, triggers Check: P\_REQUESTED\_RESPONSE\_TIME\_CANNOT\_BE\_DELIVERED is returned, or triggeredLocationReportErr() with P\_M\_POSITION\_METHOD\_FAILURE value of TpMobilityError is returned.



Summary:	triggeredLocationReportingStartReq, P_TRIGGER_CONDITIONS_NOT_SUBSCRIBED

Reference: ES 201 915-6 [1], clause 8.1

Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpTriggeredUserLocation interface through selecting that service and signing the required service agreement.

Test Sequence:

Method call triggeredLocationReportingStartReq ()
 Parameters: appLocation, users, request, triggers with conditions not subscribed
 Check: P\_TRIGGER\_CONDITIONS\_NOT\_SUBSCRIBED is returned, or
 triggeredLocationReportErr() with P\_M\_POSITION\_METHOD\_FAILURE value of
 TpMobilityError is returned.

<u>.</u>	triggers with conditions that are not subscribed	
IpAppTriggeredUserLocation		<u>:</u>
	′ \ <u> </u>	pTriggeredUserLocation
1. trigg	eredLocationReportingStartReq(appLocation, users, request, inopportune trig	igers)
<	P_TRIGGER_CONDITIONS_NOT_SUBSCRIBED	
	OR	
trigger	edLocationReportingStartReq(appLocation, users, request, inopportune trigge	are)
linggen		
triggeredLo	ocationReportErr(assignmentId, P_M_POSITION_METHODE_FAILURE, diag	nostic)

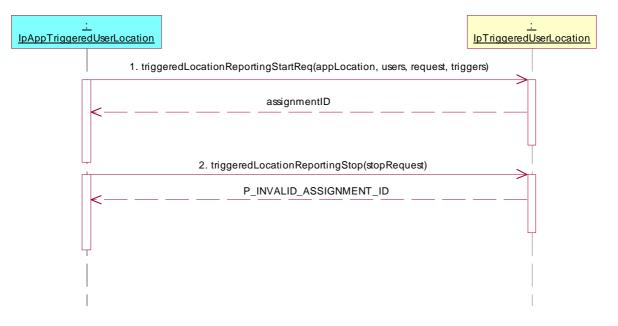
Summary: triggeredLocationReportingStop, P\_INVALID\_ASSIGNMENT\_ID

Reference: ES 201 915-6 [1], clause 8.1

Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpTriggeredUserLocation interface through selecting that service and signing the required service agreement.

37

- 1. Method call **triggeredLocationReportingStartReq**() Parameters: appLocation, users, request, triggers Check: valid value of TpSessionID is returned
- 2. Method call triggeredLocationReportingStop () Parameters: invalid stopRequest Check: P\_INVALID\_ASSIGNMENT\_ID, is returned.



## 5.2.2 User Location Camel

## Test M\_ULC\_01

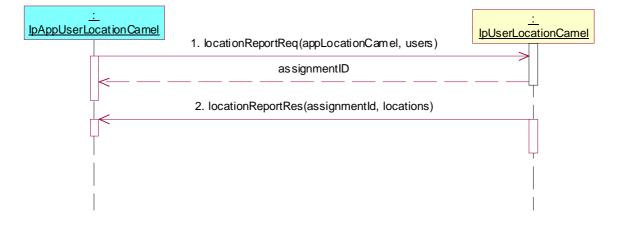
Summary: all methods, successful

Reference: ES 201 915-6 [1], clause 8.2

Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpUserLocationCamel interface through selecting that service and signing the required service agreement.

Selection Condition: locationReportReq() supported.

- 1. Method call **locationReportReq**() Parameters: appLocationCamel, users Check: valid value of TpSessionID is returned
- Triggered action: cause IUT to call locationReportRes() method on the tester's (Application) IpAppUserLocationCamel interface. Parameters: assignmentId, locations



Summary: all methods, unknown or absent subscriber

Reference: ES 201 915-6 [1], clause 8.2

Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpUserLocationCamel interface through selecting that service and signing the required service agreement.

Selection Condition: locationReportReq() supported.

- 1. Method call **locationReportReq()** Parameters: appLocationCamel, users with unknown or absent subscriber Check: valid value of TpSessionID is returned
- Triggered action: cause IUT to call locationReportErr() method on the tester's (Application) IpAppUserLocationCamel interface.
   Parameters: assignmentId, cause indicating P\_M\_UNKNOWN\_SUBSCRIBER or P\_ABSENT\_SUBSCRIBER, diagnostic

IpAppUserLo	cationCamel	IpUserLocationCamel
Γ	1. locationReportReq(appLocationCamel, unknown or absent users) assignmentID	
_	2. locationReportErr(assignmentId, P_UNKNOWN_SUBSCRIBER OR P_ABSENT_SUBSCRIBER,	diagnostic)
L		

40

## Test M\_ULC\_03

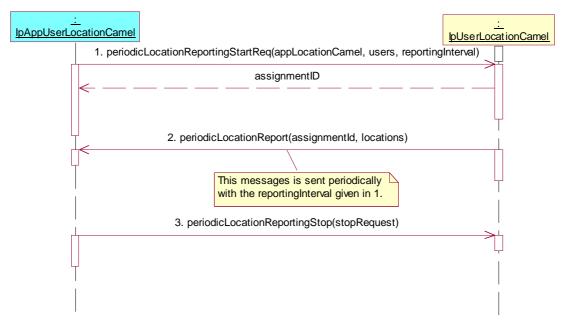
Summary: all methods, successful

Reference: ES 201 915-6 [1], clause 8.2

Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpUserLocationCamel interface through selecting that service and signing the required service agreement.

Selection Condition: periodicLocationReportingStartReq() supported.

- 1. Method call **periodicLocationReportingStartReq**() Parameters: appLocationCamel, users, reportingInterval Check: valid value of TpSessionID is returned
- 2. Triggered action: cause IUT to call periodicLocationReport () method on the tester's (Application)
   IpAppUserLocationCamel interface.
   Parameters: assignmentId, locations
   Check: no exception is returned. This message is sending with correct reporting Interval given in 1.
- 3. Method call **periodicLocationReportingStop**() Parameters: stopRequest Check: no exception is returned. No periodicLocationReport is still triggered.



Summary: all methods, unknown or absent subscriber

Reference: ES 201 915-6 [1], clause 8.2

Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpUserLocationCamel interface through selecting that service and signing the required service agreement.

41

Selection Condition: periodicLocationReportingStartReq() supported.

Test Sequence:

- 1. Method call **periodicLocationReportingStartReq()** Parameters: appLocationCamel, users with unknown or absent subscriber, reportingInterval Check: valid value of TpSessionID is returned
- 2. Triggered action: cause IUT to call **periodicLocationReportErr** () method on the tester's (Application) **IpAppUserLocationCamel** interface.

Parameters: assignmentId, cause indicating P\_M\_UNKNOWN\_SUBSCRIBER or P\_ABSENT\_SUBSCRIBER, diagnostic

<u>ipAppUserLocationCamel</u>	periodicLocationReportingStartReq(appLocationCamel, unknown or absent users, reportingInterval)	<u>:</u> erLocationCamel
	assignmentID	
☐ 2. periodicL	_ocationReportErr(assignmentId, P_UNKNOWN_SUBSCRIBER OR P_ABSENT_SUBSCRIBER, diagno	ostic)

Summary: periodicLocationReportingStartReq, P\_INVALID\_REPORTING\_INTERVAL

Reference: ES 201 915-6 [1], clause 8.2

Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpUserLocationCamel interface through selecting that service and signing the required service agreement.

42

Selection Condition: periodicLocationReportingStartReq() supported.

Test Sequence:

 1. Method call periodicLocationReportingStartReq ()

 Parameters:
 appLocationCamel, users, invalid reportingInterval

 Check:
 P\_INVALID\_REPORTING\_INTERVAL is returned, or periodicLocationReportErr() with

 P\_M\_POSITION\_METHOD\_FAILURE value of TpMobilityError.

<u>IpAppUserLocationCam</u>	<u>1</u>	<u>:</u> lpUserLocationCamel
	1. periodicLocationReportingStartReq(appLocationCamel, users, invalid reportingInterval)	
<	P_INVALID_REPORTING_INTERVAL	
	OR	
1	1. periodicLocationReportingStartReq(appLocationCamel, users, invalid reportingInterval)	
	periodicLocationReportErr(assignmentId, P_M_POSITION_METHOD_FAILURE, diagnostic)	

Summary: periodicLocationReportingStop, P\_INVALID\_ASSIGNMENT\_ID

Reference: ES 201 915-6 [1], clause 8.2

Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpUserLocationCamel interface through selecting that service and signing the required service agreement.

43

Selection Condition: periodicLocationReportingStartReq() supported.

- 1. Method call **periodicLocationReportingStartReq**() Parameters: appLocationCamel, users, reportingInterval Check: valid value of TpSessionID is returned
- 2. Method call **periodicLocationReportingStop**() Parameters: invalid stopRequest Check: P\_INVALID\_ASSIGNMENT\_ID, is returned.

<u>:</u> IpAppUserLocationCamel		<u>:</u> <u>lpUserLocationCamel</u>
1. periodicLo	ocationReportingStartReq(appLocationCamel, users, reportin	ngInterval)
	assignmentID	
		Ļ
Ļ	2. periodicLocationReportingStop(stopRequest)	
<	P_INVALID_ASSIGNMENT_ID	
		L L

Summary: all methods, successful

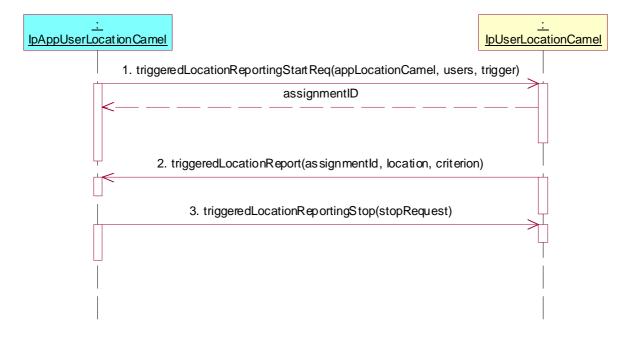
Reference: ES 201 915-6 [1], clause 8.2

Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpUserLocationCamel interface through selecting that service and signing the required service agreement.

44

 $Selection\ Condition:\ triggeredLocationReportingStartReq()\ supported.$ 

- 1. Method call **triggeredLocationReportingStartReq()** Parameters: appLocationCamel, users, trigger Check: valid value of TpSessionID is returned
- Triggered action: cause IUT to call triggeredLocationReport () method on the tester's (Application) IpAppUserLocationCamel interface. Parameters: assignmentId, location, criterion
- 3. Method call triggeredLocationReportingStop() Parameters: stoprequest Check: no exception is returned



Summary: all methods, unknown or absent subscriber

Reference: ES 201 915-6 [1], clause 8.2

Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpUserLocationCamel interface through selecting that service and signing the required service agreement.

Selection Condition: triggeredLocationReportingStartReq() supported.

Test Sequence:

- 1. Method call **triggeredLocationReportingStartReq**() Parameters: appLocationCamel, users with unknown or absent subscriber, trigger Check: valid value of TpSessionID is returned
- Triggered action: cause IUT to call triggeredLocationReportErr () method on the tester's (Application) IpAppUserLocationCamel interface.
   Parameters: assignmentId, cause indicating P\_M\_UNKNOWN\_SUBSCRIBER or

rameters: assignmentid, cause indicating P\_M\_UNKNOWN\_SUBSCRIBER of P\_ABSENT\_SUBSCRIBER, diagnostic

<u>:</u> DAppUserLo	<u>.</u> 		<u>:</u> <u>IpUserLocationCam</u>	<u>el</u>
		1. triggeredLocationReportingStartReq(appLocationCamel, unknown or absent users, trigger)		
	<	assignmentID		
L	2. triggeree	dLocationReportErr(assignmentId, P_UNKNOWN_SUBSCRIBER OR P_ABSENT_SUBSCRIBER, dia	agnostic)	
Ĺ				

Summary: triggeredLocationReportingStop, P\_INVALID\_ASSIGNMENT\_ID

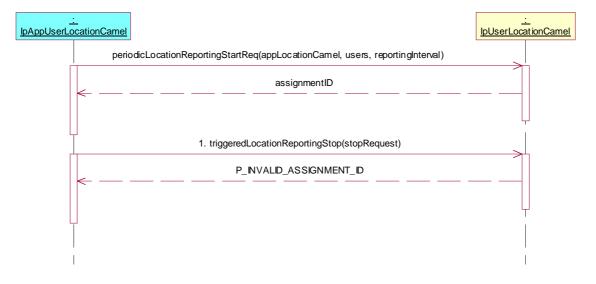
Reference: ES 201 915-6 [1], clause 8.2

Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpUserLocationCamel interface through selecting that service and signing the required service agreement.

46

Selection Condition: triggeredLocationReportingStartReq() supported.

- 1. Method call **triggeredLocationReportingStartReq()** Parameters: appLocationCamel, users, trigger Check: valid value of TpSessionID is returned
- 2. Method call triggeredLocationReportingStop () Parameters: invalid stopRequest Check: P\_INVALID\_ASSIGNMENT\_ID, is returned.



# 5.2.3 User Location Emergency

## Test M\_ULE\_01

Summary: all methods, successful

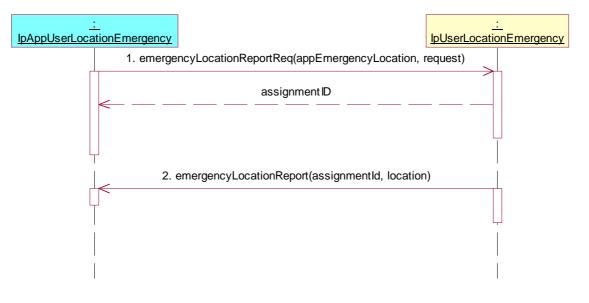
Reference: ES 201 915-6 [1], clause 8.3

Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpUserLocationEmergency interface through selecting that service and signing the required service agreement.

47

Selection Condition: emergencyLocationReportReq() supported.

- 1. Method call emergencyLocationReportReq() Parameters: appEmergencyLocation, request Check: valid value of TpSessionID is returned
- Triggered action: cause IUT to call emergencyLocationReport() method on the tester's (Application) IpAppUserLocationEmergency interface. Parameters: assignmentId, location



Summary: all methods, unknown or absent subscribe

Reference: ES 201 915-6 [1], clause 8.3

Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpUserLocationEmergency interface through selecting that service and signing the required service agreement.

Selection Condition: emergencyLocationReportReq() supported.

Test Sequence:

- 1. Method call **emergencyLocationReportReq**() Parameters: appEmergencyLocation, request with unknown or absent subscriber Check: valid value of TpSessionID is returned
- Triggered action: cause IUT to call emergencyLocationReportErr() method on the tester's (Application) IpAppUserLocationEmergency interface.
   Parameters: assignmentId, cause indicating P\_M\_UNKNOWN\_SUBSCRIBER or

P\_ABSENT\_SUBSCRIBER, diagnostic

IpAppUserLoca	tionEmergency		<u>:</u> <u>lpUserLocationEmergency</u>
г	1	. emergencyLocationReportReq(appEmergencyLocation, request unknown or absent subscriber)	
	<	assignmentID	
Г	2. emergency	LocationReportErr(assignmentId, P_UNKNOW N_SUBSCRIBER OR P_ABSENT_SUBSCRIBER,	diagnostic)
L			

Summary: all methods, successful

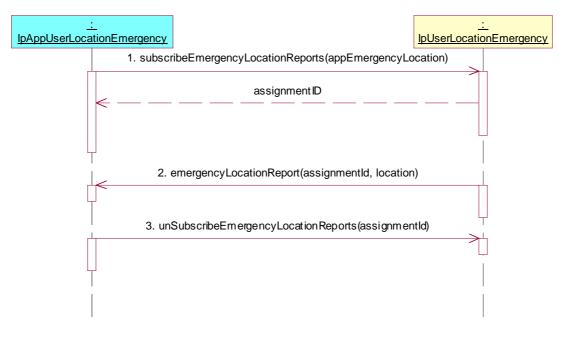
Reference: ES 201 915-6 [1], clause 8.3

Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpUserLocationEmergency interface through selecting that service and signing the required service agreement.

49

Selection Condition: subscribeEmergencyLocationReports() supported.

- 1. Method call **subscribeEmergencyLocationReports**() Parameters: appEmergencyLocation Check: valid value of TpSessionID is returned
- Triggered action: cause IUT to call emergencyLocationReport () method on the tester's (Application) IpAppUserLocationEmergency interface. Parameters: assignmentId, locations
- Method call unSubsribeEmergencyLocationReports()
   Parameters: assignmentId
   Check: no exception is returned. No emergencyLocationReport is still triggered.



Summary: unSubsribeEmergencyLocationReports, P\_INVALID\_ASSIGNMENT\_ID

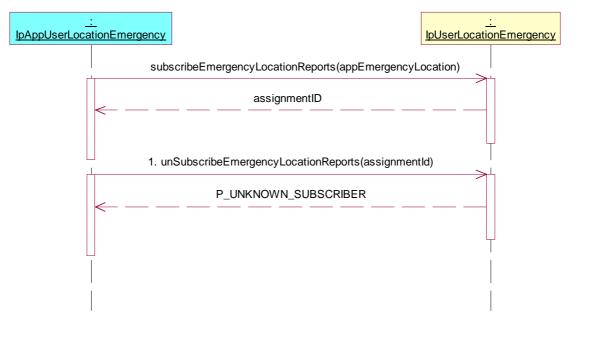
Reference: ES 201 915-6 [1], clause 8.3

Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpUserLocationEmergency interface through selecting that service and signing the required service agreement.

50

Selection Condition: subscribeEmergencyLocationReports() supported.

- 1. Method call **subscribeEmergencyLocationReports**() Parameters: appEmergencyLocation Check: valid value of TpSessionID is returned
- 2. Method call unSubsribeEmergencyLocationReports() Parameters: invalid assignmentId Check: P\_INVALID\_ASSIGNMENT\_ID, is returned.



## 5.2.4 User status

## Test M\_US\_01

Summary: all methods, successful

Reference: ES 201 915-6 [1], clause 8.4

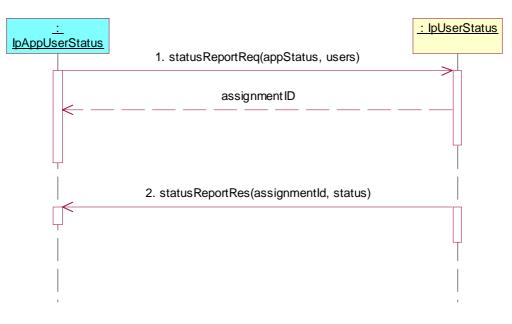
Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpUserStatus interface through selecting that service and signing the required service agreement.

Selection Condition: statusReportReq() supported.

Test Sequence:

- 1. Method call **statusReportReq**() Parameters: appStatus, users Check: valid value of TpSessionID is returned
- 2. Triggered action: cause IUT to call **statusReportRes**() method on the tester's (Application) **IpAppUserStatus** interface.

Parameters: assignmentId, status



## Test M\_US\_02

Summary: all methods, unknown or absent subscriber

Reference: ES 201 915-6 [1], clause 8.4

Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpUserStatus interface through selecting that service and signing the required service agreement.

Selection Condition: statusReportReq() supported.

Test Sequence:

- 1. Method call **statusReportReq**() Parameters: appStatus, users with unknown or absent subscriber Check: valid value of TpSessionID is returned
- 2. Triggered action: cause IUT to call **statusReportErr**() method on the tester's (Application) **IpAppUserStatus** interface.

Parameters: assignmentId, cause indicating P\_M\_UNKNOWN\_SUBSCRIBER or P\_ABSENT\_SUBSCRIBER, diagnostic

<u>:</u> pAppUserStatus		: lpUserStatus
	<ol> <li>statusReportReq(appStatus, unknown or absent users)</li> </ol>	
<	assignmentID	
2. statu	sReportErr(assignmentId, P_UNKNOWN_SUBSCRIBER OR P_ABSENT_SUBSCRIBER, diagr	nostic)
L.		
		Ť

## Test M\_US\_03

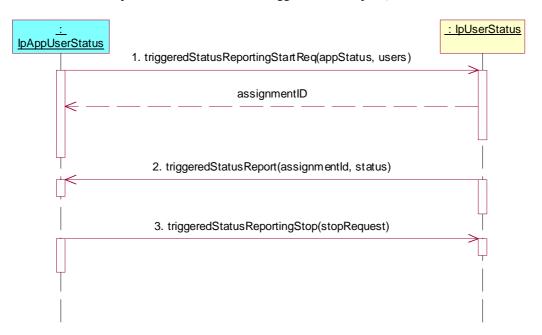
Summary: all methods, successful

Reference: ES 201 915-6 [1], clause 8.4

Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpUserStatus interface through selecting that service and signing the required service agreement.

 $Selection\ Condition:\ triggered Status Reporting Start Req()\ supported.$ 

- 1. Method call triggeredStatusReportingStartReq() Parameters: appStatus, users Check: valid value of TpSessionID is returned
- Triggered action: cause IUT to call triggeredStatusReport () method on the tester's (Application) IpAppUserStatus interface.
   Parameters: assignmentId, Status
- Method call triggeredStatusReportingStop() Parameters: stoprequest Check: no exception is returned. No more triggeredStatusReport() is returned



Summary: all methods, unknown or absent subscriber

Reference: ES 201 915-6 [1], clause 8.4

Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpUserStatus interface through selecting that service and signing the required service agreement.

54

Selection Condition: triggeredStatusReportingStartReq() supported.

Test Sequence:

- 1. Method call **triggeredStatusReportingStartReq()** Parameters: appStatus, users with unknown or absent subscriber Check: valid value of TpSessionID is returned
- Triggered action: cause IUT to call triggeredStatusReportErr() method on the tester's (Application) IpAppUserStatus interface.
   Parameters: assignmentId, cause indicating P\_M\_UNKNOWN\_SUBSCRIBER or

P\_ABSENT\_SUBSCRIBER, diagnostic

	1. triggered Status	ReportingStartReq(app	Status, unknown or ab	sent users)	>
		assignmen	tlD		
<					
triggeredStatus	Pen of Fri(assignment	HA P LINKNOWN SUE			=R diagnostic
2. triggeredStatus <	sReportErr(assignmen	tld, P_UNKNOWN_SUE	SCRIBER OR P_ABS	SENT_SUBSCRIB	ER, diagnostic
2. triggeredStatus <	sReportErr(assignmen	tid, P_UNKNOWN_SUE	SCRIBER OR P_ABS	SENT_SUBSCRIB	ER, diagnostic

## Test M\_US\_05

Summary: triggeredStatusReportingStop, P\_INVALID\_ASSIGNMENT\_ID

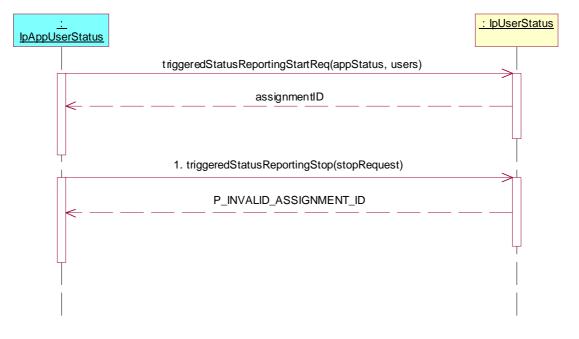
Reference: ES 201 915-6 [1], clause 8.4

Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpUserStatus interface through selecting that service and signing the required service agreement.

55

Selection Condition: triggeredStatusReportingStartReq() supported.

- 1. Method call **triggeredStatusReportingStartReq**() Parameters: appStatus, users Check: valid value of TpSessionID is returned
- 2. Method call triggeredStatusReportingStop () Parameters: invalid assignmentId Check: P\_INVALID\_ASSIGNMENT\_ID exception is returned.



# 5.3 TPs for the application using the Mobility SCF

All PICS items referred to in this clause are as specified in ES 202 170 [2] unless indicated otherwise by another numbered reference.

All parameters specified in method calls are valid unless specified.

The procedures to trigger the application to call methods in the application are dependant on the underlying network architecture and are out of the scope of this test specification. Those method calls are preceded by the words "Triggered action".

## 5.3.1 User Location

## 5.3.1.1 IpAppUserLocation interface

Precondition: IpAppUserLocation interface implemented

## Test M\_APP\_UL\_01

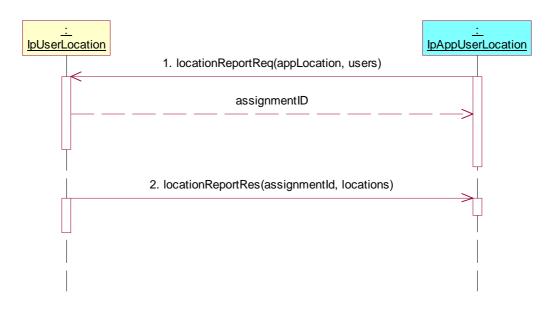
Summary: request of report of location, successful

Reference: ES 201 915-6 [1], clause 8.1

Precondition: IUT capable of invoking locationReportReq()

Preamble: Registration of the IUT (application) and the tester (User Location SCF) to the framework. The tester must have obtained a reference to an instance of the IpUserLocation interface through selecting that service and signing the required service agreement.

- Triggered action: cause IUT to call locationReportReq() method on the tester's (SCF) IpUserLocation interface.
   Parameters: appLocation, users
- 2. Method call **locationReportRes()** Parameters: assignmentId, locations Check: no exception is returned



## Test M\_APP\_UL\_02

Summary: request of report of location, unsuccessful

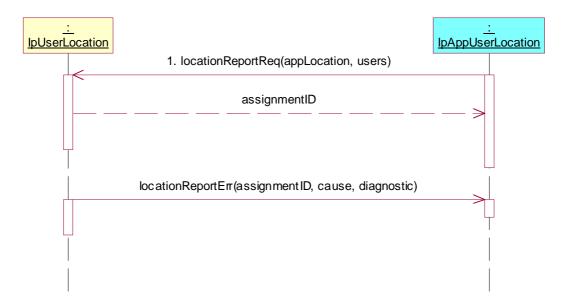
Reference: ES 201 915-6 [1], clause 8.1

Precondition: IUT capable of invoking locationReportReq()

Selection: locationReportReq method supported - PICS item: [2] UL1

Preamble: Registration of the IUT (application) and the tester (User Location SCF) to the framework. The tester must have obtained a reference to an instance of the IpUserLocation interface through selecting that service and signing the required service agreement.

- Triggered action: cause IUT to call locationReportReq() method on the tester's (SCF) IpUserLocation interface.
   Parameters: appLocation, users
- 2. Method call **locationReportErr(**) Parameters: assignmentId, cause, diagnostic Check: no exception is returned



Summary: advanced request of report of location, successful

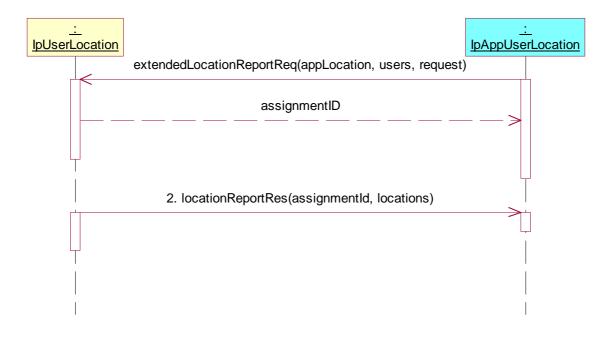
Reference: ES 201 915-6 [1], clause 8.1

Precondition: IUT capable of invoking **extendedLocationReportReq(**)

Preamble: Registration of the IUT (application) and the tester (User Location SCF) to the framework. The tester must have obtained a reference to an instance of the IpUserLocation interface through selecting that service and signing the required service agreement.

58

- Triggered action: cause IUT to call extendedLocationReportReq() method on the tester's (SCF) IpUserLocation interface.
   Parameters: appLocation, users, request
- 2. Method call **extendedLocationReportRes**() Parameters: assignmentId, locations Check: no exception is returned



59

## Test M\_APP\_UL\_04

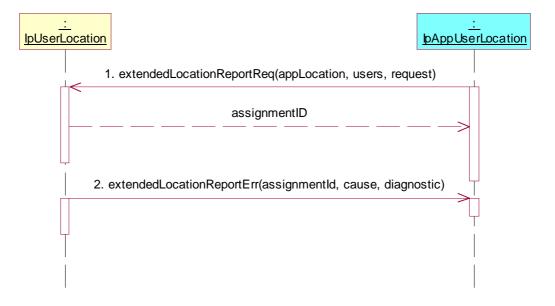
Summary: advanced request of report of location, unsuccessful

Reference: ES 201 915-6 [1], clause 8.1

Precondition: IUT capable of invoking **extendedLocationReportReq(**)

Preamble: Registration of the IUT (application) and the tester (User Location SCF) to the framework. The tester must have obtained a reference to an instance of the IpUserLocation interface through selecting that service and signing the required service agreement.

- Triggered action: cause IUT to call extendedLocationReportReq() method on the tester's (SCF) IpUserLocation interface.
   Parameters: appLocation, users, request
- 2. Method call **extendedLocationReportErr**() Parameters: assignmentId, cause, diagnostic Check: no exception is returned



## Test M\_APP\_UL\_05

Summary: request for periodic reports of location, successful

Reference: ES 201 915-6 [1], clause 8.1

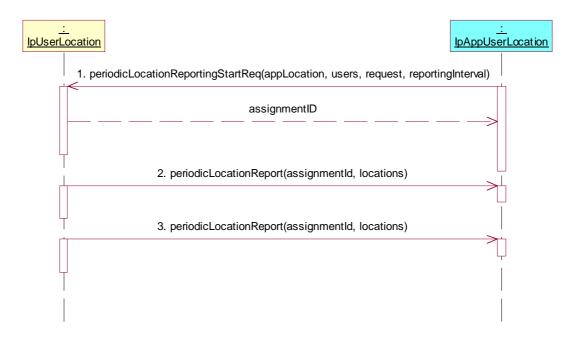
Precondition: IUT capable of invoking periodicLocationReportingStartReq()

Preamble: Registration of the IUT (application) and the tester (User Location SCF) to the framework. The tester must have obtained a reference to an instance of the IpUserLocation interface through selecting that service and signing the required service agreement.

Test Sequence:

- Triggered action: cause IUT to call periodicLocationReportingStartReq() method on the tester's (SCF) IpUserLocation interface.
   Parameters: appLocation, users, request, reportingInterval
- 2. Method call **periodicLocationReport**() Parameters: assignmentId, locations Check: no exception is returned

NOTE: Repeat method call periodicLocationReport() with delays as specified in received reportingInterval.



# Summary:start and stop of a request for periodic reports of location, successfulReference:ES 201 915-6 [1], clause 8.1Precondition:IUT capable of invoking periodicLocationReportingStartReq() and periodicLocationReportingStop<br/>()Preamble:Registration of the IUT (application) and the tester (User Location SCF) to the framework. The tester<br/>must have obtained a reference to an instance of the IpUserLocation interface through selecting that<br/>service and signing the required service agreement.

61

Test Sequence:

- Triggered action: cause IUT to call periodicLocationReportingStartReq() method on the tester's (SCF) IpUserLocation interface.
   Parameters: appLocation, users, request, reportingInterval
- 2. Method call **periodicLocationReport**() Parameters: assignmentId, locations Check: no exception is returned

NOTE: Repeat method call **periodicLocationReport()** with delays as specified in received reportingInterval.

Triggered action: cause IUT to call periodicLocationReportingStop() method on the tester's (SCF)
 IpUserLocation interface.
 Parameters: stopRequest

IpUserLocation	<u>:</u> lpAppUserLocation
1. periodicLocationReportingStartReq(appLocation, users, request, reporti	
assignmentID	>
2. periodicLocationReport(assignmentId, locations)	
3. periodicLocationReport(assignmentId, locations)	
4. periodicLocationReportingStop(stopRequest)	Ч 

## Test M\_APP\_UL\_06

## Test M\_APP\_UL\_07

Summary: request for periodic reports of location, unsuccessful

Reference: ES 201 915-6 [1], clause 8.1

Precondition: IUT capable of invoking periodicLocationReportingStartReq()

Preamble: Registration of the IUT (application) and the tester (User Location SCF) to the framework. The tester must have obtained a reference to an instance of the IpUserLocation interface through selecting that service and signing the required service agreement.

62

Test Sequence:

 Triggered action: cause IUT to call periodicLocationReportingStartReq() method on the tester's (SCF) IpUserLocation interface.
 Parameters: appLocation, users, request, reportingInterval

2. Method call **periodicLocationReportErr**() Parameters: assignmentId, cause, diagnostic Check: no exception is returned

<u>pUserLocation</u>	<u>:</u> IpAppUserLocation
│ 1. periodicLocationReportingStartReq(appLocation, users, request, reportin	igInterval)
assignmentID	>
2. periodic Location Report Err(assignmentId, cause, diagnostic)	

## 5.3.1.2 IpAppTrigerredUserLocation interface

Precondition: IpAppTrigerredUserLocation interface implemented

## Test M\_APP\_TUL\_01

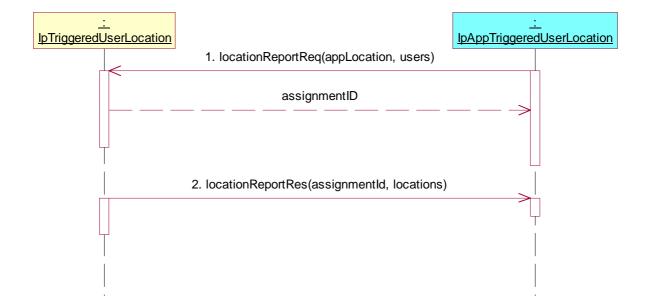
Summary: request of report of location, successful

Reference: ES 201 915-6 [1], clause 8.1

Precondition: IUT capable of invoking locationReportReq()

Preamble: Registration of the IUT (application) and the tester (User Location SCF) to the framework. The tester must have obtained a reference to an instance of the IpTriggeredUserLocation interface through selecting that service and signing the required service agreement.

- Triggered action: cause IUT to call locationReportReq() method on the tester's (SCF) IpTriggeredUserLocation interface. Parameters: appLocation, users
- 2. Method call **locationReportRes()** Parameters: assignmentId, locations Check: no exception is returned



## Test M\_APP\_TUL\_02

Summary: request of report of location, unsuccessful

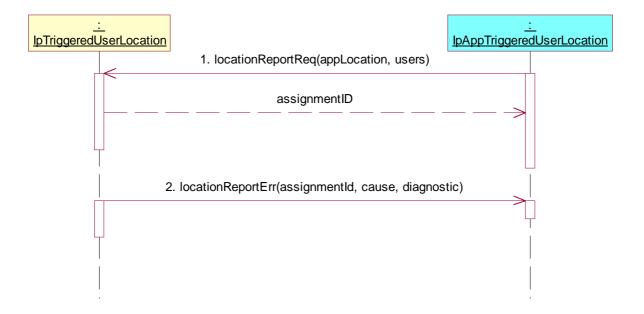
Reference: ES 201 915-6 [1], clause 8.1

Precondition: IUT capable of invoking **locationReportReq(**)

Selection: locationReportReq method supported - PICS item: [2] UL1

Preamble: Registration of the IUT (application) and the tester (User Location SCF) to the framework. The tester must have obtained a reference to an instance of the IpTriggeredUserLocation interface through selecting that service and signing the required service agreement.

- Triggered action: cause IUT to call locationReportReq() method on the tester's (SCF) IpTriggeredUserLocation interface. Parameters: appLocation, users
- 2. Method call **locationReportErr(**) Parameters: assignmentId, cause, diagnostic Check: no exception is returned



## Test M\_APP\_TUL\_03

Summary: advanced request of report of location, successful

Reference: ES 201 915-6 [1], clause 8.1

Precondition: IUT capable of invoking **extendedLocationReportReq(**)

Preamble: Registration of the IUT (application) and the tester (User Location SCF) to the framework. The tester must have obtained a reference to an instance of the IpTriggeredUserLocation interface through selecting that service and signing the required service agreement.

- Triggered action: cause IUT to call extendedLocationReportReq() method on the tester's (SCF) IpTriggeredUserLocation interface. Parameters: appLocation, users, request
- 2. Method call **extendedLocationReportRes(**) Parameters: assignmentId, locations Check: no exception is returned

<u>:</u> IpTriggeredUserLocation		<u>.</u> IpAppTriggered	_ dUserLocation
1. e	xtendedLocationReportReq(appLocation, users, re	quest)	7
	assignmentID	>	
2.	extendedLocationReportRes(assignmentId, locati	ons)	]

Summary: advanced request of report of location, unsuccessful

Reference: ES 201 915-6 [1], clause 8.1

Precondition: IUT capable of invoking **extendedLocationReportReq(**)

Preamble: Registration of the IUT (application) and the tester (User Location SCF) to the framework. The tester must have obtained a reference to an instance of the IpTriggeredUserLocation interface through selecting that service and signing the required service agreement.

66

- Triggered action: cause IUT to call extendedLocationReportReq() method on the tester's (SCF) IpTriggeredUserLocation interface. Parameters: appLocation, users, request
- 2. Method call **extendedLocationReportErr(**) Parameters: assignmentId, cause, diagnostic Check: no exception is returned

<u>:</u> IpTriggeredUserLocation		<u>IpAppTriggere</u>	_ dUserLocation
1. e	xtendedLocationReportReq(appLocation, users, re	quest)	
	assignmentID	>	
2. ex	tendedLocationReportErr(assignmentId, cause,dia	gnostic)	]

## Test M\_APP\_TUL\_05

Summary: request for periodic reports of location, successful

Reference: ES 201 915-6 [1], clause 8.1

Precondition: IUT capable of invoking periodicLocationReportingStartReq()

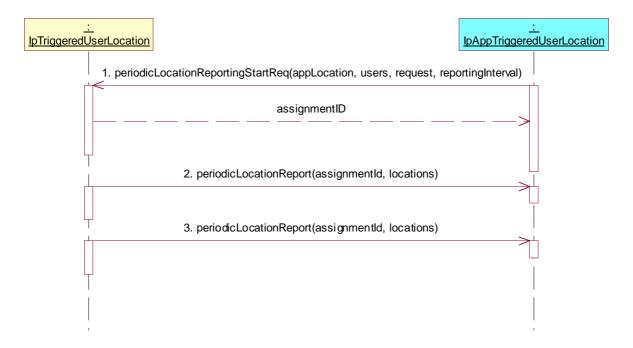
Preamble: Registration of the IUT (application) and the tester (User Location SCF) to the framework. The tester must have obtained a reference to an instance of the IpTriggeredUserLocation interface through selecting that service and signing the required service agreement.

67

Test Sequence:

- Triggered action: cause IUT to call periodicLocationReportingStartReq() method on the tester's (SCF) IpTriggeredUserLocation interface.
   Parameters: appLocation, users, request, reportingInterval
- 2. Method call **periodicLocationReport**() Parameters: assignmentId, locations Check: no exception is returned

NOTE: Repeat method call periodicLocationReport() with delays as specified in received reportingInterval.



## Test M\_APP\_TUL\_06

Summary: start and stop of a request for periodic reports of location, successful

Reference: ES 201 915-6 [1], clause 8.1

- Precondition: IUT capable of invoking **periodicLocationReportingStartReq()** and **periodiclocationReportingStopReq()**
- Preamble: Registration of the IUT (application) and the tester (User Location SCF) to the framework. The tester must have obtained a reference to an instance of the IpTriggeredUserLocation interface through selecting that service and signing the required service agreement.

Test Sequence:

- Triggered action: cause IUT to call periodicLocationReportingStartReq() method on the tester's (SCF) IpTriggeredUserLocation interface.
   Parameters: appLocation, users, request, reportingInterval
- 2. Method call **periodicLocationReport**() Parameters: assignmentId, locations Check: no exception is returned

NOTE: Repeat method call **periodicLocationReport()** with delays as specified in received reportingInterval.

 Triggered action: cause IUT to call periodicLocationReportingStop() method on the tester's (SCF) IpTriggeredUserLocation interface. Parameters: stopRequest

<u>:</u> lpTriggeredUserLocation		<u>:</u> IpAppTriggeredUserLocation
1. periodic	LocationReportingStartReq(appLocation, users, request, repo	ortingInterval)
	assignmentID	>
	2. periodicLocationReport(assignmentId, locations)	
	3. periodicLocationReport(assignmentId, locations)	
T	4. periodicLocationReportingStop(stopRequest)	

Summary: request for periodic reports of location, unsuccessful

Reference: ES 201 915-6 [1], clause 8.1

Precondition: IUT capable of invoking periodicLocationReportingStartReq()

Preamble: Registration of the IUT (application) and the tester (User Location SCF) to the framework. The tester must have obtained a reference to an instance of the IpTriggeredUserLocation interface through selecting that service and signing the required service agreement.

69

- Triggered action: cause IUT to call periodicLocationReportingStartReq() method on the tester's (SCF) IpTriggeredUserLocation interface.
   Parameters: appLocation, users, request, reportingInterval
- 2. Method call **periodicLocationReportErr(**) Parameters: assignmentId, cause, diagnostics Check: no exception is returned

<u>:</u> IpTriggeredUserLocation		: IpAppTriggeredUserLocation
1. periodic	LocationReportingStartReq(appLocation, users, request, repo	prtingInterval)
	assignmentID	
	2. periodicLocationReportErr(assignmentId, cause, diagnostic	c)
Ι		l.

Summary: request for triggered reports of location, successful

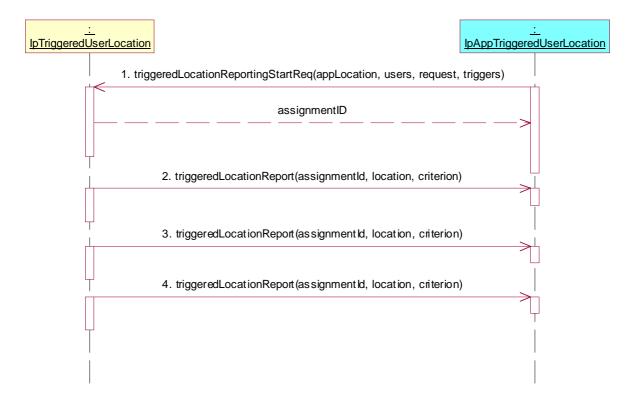
Reference: ES 201 915-6 [1], clause 8.1

Precondition: IUT capable of invoking triggeredLocationReportingStartReq()

Preamble: Registration of the IUT (application) and the tester (User Location SCF) to the framework. The tester must have obtained a reference to an instance of the IpTriggeredUserLocation interface through selecting that service and signing the required service agreement.

- Triggered action: cause IUT to call triggeredLocationReportingStartReq() method on the tester's (SCF) IpTriggeredUserLocation interface.
   Parameters: appLocation, users, request, triggers
- 2. Method call **triggeredLocationReport**() Parameters: assignmentId, location, criterion Check: no exception is returned





## Test M\_APP\_TUL\_09

Summary: start and stop of a request for triggered reports of location, successful

Reference: ES 201 915-6 [1], clause 8.1

- Precondition: IUT capable of invoking triggeredLocationReportingStartReq() and triggeredlocationReportingStopReq()
- Preamble: Registration of the IUT (application) and the tester (User Location SCF) to the framework. The tester must have obtained a reference to an instance of the IpTriggeredUserLocation interface through selecting that service and signing the required service agreement.

71

- Triggered action: cause IUT to call triggeredLocationReportingStartReq() method on the tester's (SCF) IpTriggeredUserLocation interface.
   Parameters: appLocation, users, request, triggers
- 2. Method call **triggeredLocationReport**() Parameters: assignmentId, location, criterion Check: no exception is returned
- NOTE: Repeat method call triggeredLocationReport() several times.
- 4. Triggered action: cause IUT to call triggeredlocationReportingStopReq() method on the tester's (SCF) IpTriggeredUserLocation interface. Parameters: stopRequest

<u>:</u> IpTriggeredUs	erLocation		<u>IpAppTriggere</u>	<u>dUserLocation</u>
	1. trigger	redLocationReportingStartReq(appLocation, users, reques	st, triggers)	L.
	<b>_</b>	assignmentID	>	
	2	2. triggeredLocationReport(assignmentId, location, criterio	in)	
	;	3. triggeredLocationReport(assignmentId, location, criteric	on)	
		4. triggeredLocationReportingStop(stopRequest)	L	

## Test M\_APP\_TUL\_10

Summary: request for triggered reports of location, unsuccessful

Reference: ES 201 915-6 [1], clause 8.1

Precondition: IUT capable of invoking triggeredLocationReportingStartReq()

Preamble: Registration of the IUT (application) and the tester (User Location SCF) to the framework. The tester must have obtained a reference to an instance of the IpTriggeredUserLocation interface through selecting that service and signing the required service agreement.

72

- Triggered action: cause IUT to call triggeredLocationReportingStartReq() method on the tester's (SCF) IpTriggeredUserLocation interface.
   Parameters: appLocation, users, request, triggers
- 2. Method call **triggeredLocationReportErr**() Parameters: assignmentId, cause, diagnostic Check: no exception is returned

<u>IpTriggered</u>	<u>:</u> JserLocation		<u>:</u> IpAppTriggeredUserLocation
-	1. trigger	edLocationReportingStartReq(appLocation, users, reques	st, triggers)
		assignmentID	
_	2.	triggeredLocationReportErr(assignmentId, cause, diagno	stic)
	Ļ		

# 5.3.2 User Location Camel

Precondition: IpAppUserLocationCamel interface implemented

# Test M\_APP\_ULC\_01

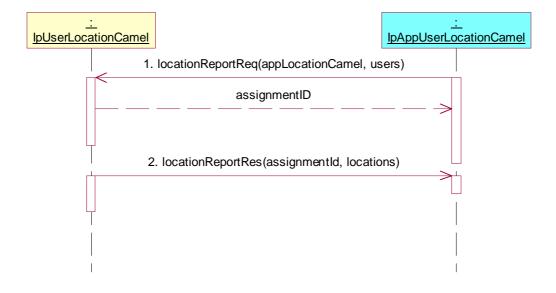
Summary: request of report of location, successful

Reference: ES 201 915-6 [1], clause 8.1

Precondition: IUT capable of invoking locationReportReq()

Preamble: Registration of the IUT (application) and the tester (User Location SCF) to the framework. The tester must have obtained a reference to an instance of the IpUserLocationCamel interface through selecting that service and signing the required service agreement.

- Triggered action: cause IUT to call locationReportReq() method on the tester's (SCF) IpUserLocationCamel interface.
   Parameters: appLocationCamel, users
- 2. Method call **locationReportRes**() Parameters: assignmentId, locations Check: no exception is returned



Summary: request of report of location, unsuccessful

Reference: ES 201 915-6 [1], clause 8.1

Precondition: IUT capable of invoking **locationReportReq(**)

Selection: locationReportReq method supported - PICS item: [2] UL1

Preamble: Registration of the IUT (application) and the tester (User Location SCF) to the framework. The tester must have obtained a reference to an instance of the IpUserLocationCamel interface through selecting that service and signing the required service agreement.

74

Test Sequence:

 Triggered action: cause IUT to call locationReportReq() method on the tester's (SCF) IpUserLocationCamel interface.
 Permeters: appL contionCamel users

Parameters: appLocationCamel, users

2. Method call **locationReportErr(**) Parameters: assignmentId, cause, diagnostic Check: no exception is returned

lpUserLoca	ationCamel				<u>IpAppUserLo</u>	<u>cationCamel</u>
_	1.	locationReportF	Req(appLocatio	nCamel, us	sers)	_
		a	assignmentID			
					>	
L	2. loc	ationReportErr(	ássignmentld, (	cause, diaç	gnostic)	
L						

Summary: request for periodic reports of location, successful

Reference: ES 201 915-6 [1], clause 8.1

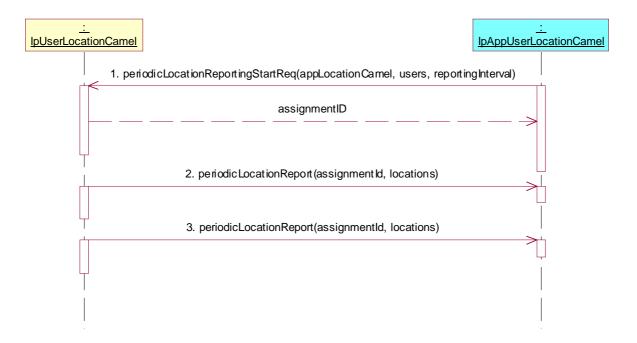
Precondition: IUT capable of invoking periodicLocationReportingStartReq()

Preamble: Registration of the IUT (application) and the tester (User Location SCF) to the framework. The tester must have obtained a reference to an instance of the IpUserLocationCamel interface through selecting that service and signing the required service agreement.

Test Sequence:

- Triggered action: cause IUT to call periodicLocationReportingStartReq() method on the tester's (SCF) IpUserLocationCamel interface.
   Parameters: appLocationCamel, users, reportingInterval
- 2. Method call **periodicLocationReport**() Parameters: assignmentId, locations Check: no exception is returned

NOTE: Repeat method call periodicLocationReport() with delays as specified in received reportingInterval.



Summary: start and stop of a request for periodic reports of location, successful

Reference: ES 201 915-6 [1], clause 8.1

- Precondition: IUT capable of invoking **periodicLocationReportingStartReq()** and **periodiclocationReportingStopReq()**
- Preamble: Registration of the IUT (application) and the tester (User Location SCF) to the framework. The tester must have obtained a reference to an instance of the IpUserLocationCamel interface through selecting that service and signing the required service agreement.

Test Sequence:

- Triggered action: cause IUT to call periodicLocationReportingStartReq() method on the tester's (SCF) IpUserLocationCamel interface.
   Parameters: appLocationCamel, users, reportingInterval
- 2. Method call **periodicLocationReport**() Parameters: assignmentId, locations Check: no exception is returned

NOTE: Repeat method call **periodicLocationReport()** with delays as specified in received reportingInterval.

 Triggered action: cause IUT to call periodicLocationReportingStop() method on the tester's (SCF) IpUserLocationCamel interface. Parameters: stopRequest

<u>:</u> lpUserLocationCamel		<u>:</u> IpAppUserLocationCamel
│ 1. peri	odicLocationReportingStartReq(appLocationCamel, users, reporting	gInterval)
	assignmentID	
	2. periodicLocationReport (assignment ld, locations)	
	3. periodicLocationReport(assignmentId, locations)	
	4. periodicLocationReportingStop(stopRequest)	

Summary: request for periodic reports of location, unsuccessful

Reference: ES 201 915-6 [1], clause 8.1

Precondition: IUT capable of invoking periodicLocationReportingStartReq()

Preamble: Registration of the IUT (application) and the tester (User Location SCF) to the framework. The tester must have obtained a reference to an instance of the IpUserLocationCamel interface through selecting that service and signing the required service agreement.

77

- Triggered action: cause IUT to call periodicLocationReportingStartReq() method on the tester's (SCF) IpUserLocationCamel interface.
   Parameters: appLocationCamel, users, reportingInterval
- 2. Method call **periodicLocationReportErr**() Parameters: assignmentId, cause, diagnostic Check: no exception is returned

<u>:</u> IpUserLocationCamel		<u>:</u> IpAppUserLocationCamel
1. peri	odicLocationReportingStartReq(appLocationCamel, users, reporting	gInterval)
	assignmentID	>
 	2. periodicLocationReportErr(assignmentId, cause, diagnostic)	
·		

Summary: request for triggered reports of location, successful

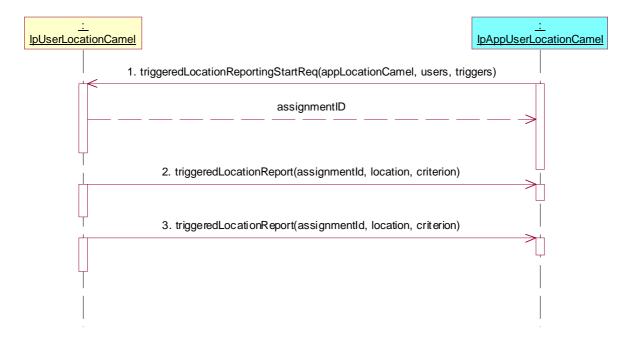
Reference: ES 201 915-6 [1], clause 8.1

Precondition: IUT capable of invoking triggeredLocationReportingStartReq()

Preamble: Registration of the IUT (application) and the tester (User Location SCF) to the framework. The tester must have obtained a reference to an instance of the IpUserLocationCamel interface through selecting that service and signing the required service agreement.

- Triggered action: cause IUT to call triggeredLocationReportingStartReq() method on the tester's (SCF) IpUserLocationCamel interface.
   Parameters: appLocationCamel, users, triggers
- 2. Method call **triggeredLocationReport**() Parameters: assignmentId, location, criterion Check: no exception is returned





Summary: start and stop of a request for triggered reports of location, successful

Reference: ES 201 915-6 [1], clause 8.1

- Precondition: IUT capable of invoking triggeredLocationReportingStartReq() and triggeredlocationReportingStopReq()
- Preamble: Registration of the IUT (application) and the tester (User Location SCF) to the framework. The tester must have obtained a reference to an instance of the IpUserLocationCamel interface through selecting that service and signing the required service agreement.

- Triggered action: cause IUT to call triggeredLocationReportingStartReq() method on the tester's (SCF) IpUserLocationCamel interface.
   Parameters: appLocationCamel, users, triggers
- 2. Method call **triggeredLocationReport**() Parameters: assignmentId, location, criterion Check: no exception is returned
- NOTE: Repeat method call triggeredLocationReport() several times.
- Triggered action: cause IUT to call triggeredLocationReportingStop() method on the tester's (SCF) IpUserLocationCamel interface. Parameters: stopRequest

<u>:</u> lpUserLocationCamel		<u>:</u> IpAppUserLocationCamel
1.1	riggeredLocationReportingStartReq(appLocationCamel, users, trig	gers)
	assignmentID	>
	2. triggeredLocationReport(assignmentId, location, criterion)	
	3. triggeredLocationReport(assignmentId, location, criterion)	
	4. triggeredLocationReportingStop(stopRequest)	

Summary: request for triggered reports of location, unsuccessful

Reference: ES 201 915-6 [1], clause 8.1

Precondition: IUT capable of invoking triggeredLocationReportingStartReq()

Preamble: Registration of the IUT (application) and the tester (User Location SCF) to the framework. The tester must have obtained a reference to an instance of the IpUserLocationCamel interface through selecting that service and signing the required service agreement.

- Triggered action: cause IUT to call triggeredLocationReportingStartReq() method on the tester's (SCF) IpUserLocationCamel interface.
   Parameters: appLocationCamel, users, triggers
- 2. Method call **triggeredLocationReportErr**() Parameters: assignmentId, cause, diagnostic Check: no exception is returned

<u>:</u> IpUserLocationCa	amel		<u>:</u> IpAppUserLo	_ cationCamel
	1. 1	riggeredLocationReportingStartReq(appLocationCamel, users, trig	gers)	1
		assignmentID	>	
		2. triggeredLocationReportErr(assignmentId, cause, diagnostic)	_ اح	]
I			I	

# 5.3.3 User Location Emergency

Precondition: IpAppUserLocationEmergency interface implemented

# Test M\_APP\_ULE\_01

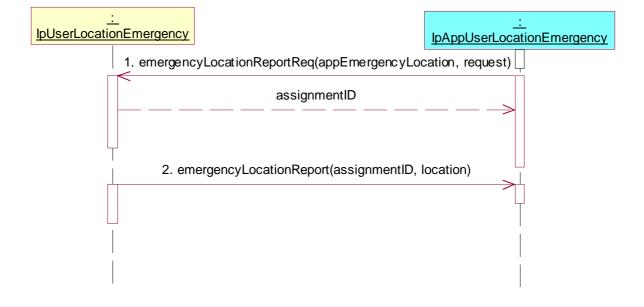
Summary: request of report of location, successful

Reference: ES 201 915-6 [1], clause 8.1

Precondition: IUT capable of invoking emergencyLocationReportReq()

Preamble: Registration of the IUT (application) and the tester (User Location SCF) to the framework. The tester must have obtained a reference to an instance of the IpUserLocationEmergency interface through selecting that service and signing the required service agreement.

- Triggered action: cause IUT to call emergencyLocationReportReq() method on the tester's (SCF) IpUserLocationEmergency interface.
   Parameters: appEmergencyLocation, request
- 2. Method call emergencyLocationReport() Parameters: assignmentId, location Check: no exception is returned



Summary: request of report of location, unsuccessful

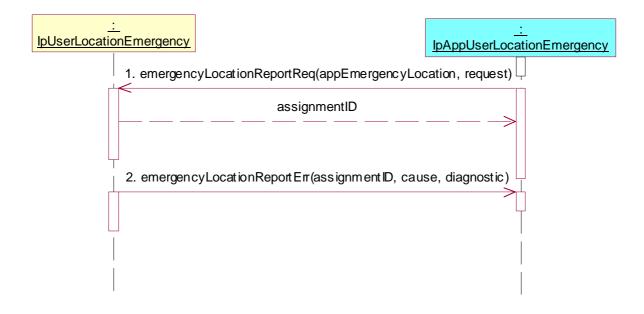
Reference: ES 201 915-6 [1], clause 8.1

Precondition: IUT capable of invoking emergencyLocationReportReq()

Selection: locationReportReq method supported - PICS item: [2] UL1

Preamble: Registration of the IUT (application) and the tester (User Location SCF) to the framework. The tester must have obtained a reference to an instance of the IpUserLocationEmergency interface through selecting that service and signing the required service agreement.

- Triggered action: cause IUT to call emergencyLocationReportReq() method on the tester's (SCF) IpUserLocationEmergency interface. Parameters: appLocationCamel, users
- 2. Method call **emergencyLocationReportErr**() Parameters: assignmentId, cause, diagnostic Check: no exception is returned



Summary: subscribe to network initiated reports of location

Reference: ES 201 915-6 [1], clause 8.1

Precondition: IUT capable of invoking subscribeEmergencyLocationReports()

Preamble: Registration of the IUT (application) and the tester (User Location SCF) to the framework. The tester must have obtained a reference to an instance of the IpUserLocationEmergency interface through selecting that service and signing the required service agreement.

83

Test Sequence:

 Triggered action: cause IUT to call subscribeEmergencyLocationReports() method on the tester's (SCF) IpUserLocationEmergency interface.
 Parameters: appEmergencyLocation

<u>:</u> IpUserLocationEmerger	<mark>су</mark>	 IpAppUserLocationEmergency
1. :	subscribeEmergencyLocationReports(appEmergenc	yLocation)
	assignmentID	- — — >
		T 
		1

Summary: subscribe and unsubscribe to network initiated reports of location

Reference: ES 201 915-6 [1], clause 8.1

Precondition: IUT capable of invoking **subscribeEmergencyLocationReports**() and **unsubscribeEmergencyLocationReports**()

Preamble: Registration of the IUT (application) and the tester (User Location SCF) to the framework. The tester must have obtained a reference to an instance of the IpUserLocationEmergency interface through selecting that service and signing the required service agreement.

84

- Triggered action: cause IUT to call subscribeEmergencyLocationReports() method on the tester's (SCF) IpUserLocationEmergency interface. Parameters: appEmergencyLocation
- Triggered action: cause IUT to call unsubscribeEmergencyLocationReports() method on the tester's (SCF) IpUserLocationEmergency interface. Parameters: assignmentId

<u>:</u> <u>IpUserLocationEmergency</u>		<u>:</u> lpAppUserLocationEmergency
1. sub	scribeEmergencyLocationReports(appEmergency	/Location)
	assignmentID	
	InSubscribeEmergencyLocationReports(assignm	ant(D)
2.0		
I		

# 5.3.4 User Status

Precondition: IpAppUserStatus interface implemented

# Test M\_APP\_US\_01

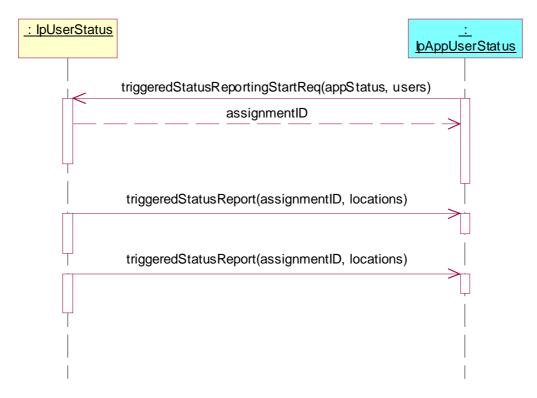
Summary: request of status report, successful

Reference: ES 201 915-6 [1], clause 8.1

Precondition: IUT capable of invoking statusReportReq()

Preamble: Registration of the IUT (application) and the tester (User Location SCF) to the framework. The tester must have obtained a reference to an instance of the IpUserStatus interface through selecting that service and signing the required service agreement.

- 1. Triggered action: cause IUT to call **statusReportReq**() method on the tester's (SCF) **IpUserStatus** interface. Parameters: appStatus, users
- 2. Method call **statusReportRes**() Parameters: assignmentId, status Check: no exception is returned



# Test M\_APP\_US\_02

Summary: request of status report, unsuccessful

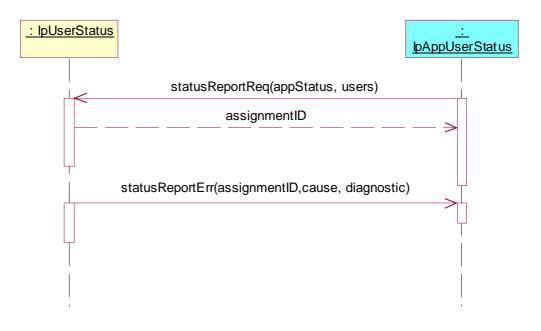
Reference: ES 201 915-6 [1], clause 8.1

Precondition: IUT capable of invoking statusReportReq()

Preamble: Registration of the IUT (application) and the tester (User Location SCF) to the framework. The tester must have obtained a reference to an instance of the IpUserStatus interface through selecting that service and signing the required service agreement.

86

- 1. Triggered action: cause IUT to call **statusReportReq()** method on the tester's (SCF) **IpUserStatus** interface. Parameters: appStatus, users
- 2. Method call **statusReportErr**() Parameters: assignmentId, cause, diagnostic Check: no exception is returned



Summary: request for triggered status reports, successful

Reference: ES 201 915-6 [1], clause 8.1

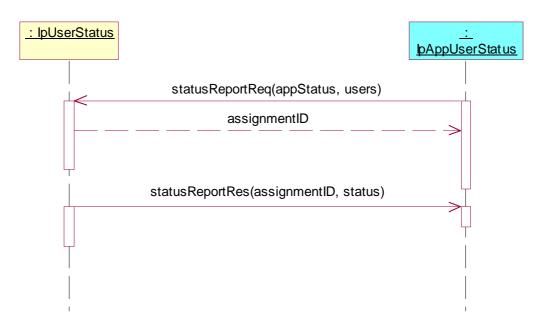
Precondition: IUT capable of invoking triggeredStatusReportingStartReq()

Preamble: Registration of the IUT (application) and the tester (User Location SCF) to the framework. The tester must have obtained a reference to an instance of the IpUserStatus interface through selecting that service and signing the required service agreement.

Test Sequence:

- Triggered action: cause IUT to call triggeredStatusReportingStartReq() method on the tester's (SCF) IpUserStatus interface.
   Parameters: appStatus, users
- 2. Method call **triggeredStatusReport**() Parameters: assignmentId, locations Check: no exception is returned

NOTE: Repeat method call triggeredStatusReport() with delays as specified in received reportingInterval.



#### Test M\_APP\_US\_04

Summary: start and stop triggered request of status report, successful

Reference: ES 201 915-6 [1], clause 8.1

Precondition: IUT capable of invoking triggeredStatusReportingStartReq() and triggeredStatusReportingStop()

88

Preamble: Registration of the IUT (application) and the tester (User Location SCF) to the framework. The tester must have obtained a reference to an instance of the IpUserStatus interface through selecting that service and signing the required service agreement.

Test Sequence:

- Triggered action: cause IUT to call triggeredStatusReportingStartReq() method on the tester's (SCF) IpUserStatus interface.
   Parameters: appStatus, users
- 2. Method call **triggeredStatusReport**() Parameters: assignmentId, locations Check: no exception is returned

NOTE: Repeat method call triggeredStatusReport() with delays as specified in received reportingInterval.

4. Triggered action: cause IUT to call triggeredStatusReportingStop() method on the tester's (SCF)
 IpUserStatus interface.
 Parameters: stopRequest



#### Test M\_APP\_US\_05

Summary: request for triggered status reports, unsuccessful

Reference: ES 201 915-6 [1], clause 8.1

Precondition: IUT capable of invoking triggeredStatusReportingStartReq() and triggeredStatusReportingStop()

Preamble: Registration of the IUT (application) and the tester (User Location SCF) to the framework. The tester must have obtained a reference to an instance of the IpUserStatus interface through selecting that service and signing the required service agreement.

- Triggered action: cause IUT to call triggeredStatusReportingStartReq() method on the tester's (SCF) IpUserStatus interface.
   Parameters: appStatus, users
- 2. Method call triggeredStatusReportErr() Parameters: assignmentId, cause, diagnostic Check: no exception is returned

: IpUse	erStatus		<u>l</u> pAppUs	<u>erStatus</u>
		triggeredStatusReportingStartReq(appStatus, us	ers)	٦
		assignmentID	$- \rightarrow$	
Γ	trigg	eredStatusReportErr(assignmentID, cause, diagn	ostic)	7
			L I	

# History

	Document history				
V1.1.1	August 2003	Publication			
V1.2.1	December 2004	Membership Approval Procedure MV 20050211: 2004-12-14 to 2005-02-11			

90