

# ETSI ES 202 196-6 V1.2.1 (2005-02)

---

*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

---

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

<http://portal.etsi.org/tb/status/status.asp>

If you find errors in the present document, please send your comment to one of the following services:

[http://portal.etsi.org/chaicor/ETSI\\_support.asp](http://portal.etsi.org/chaicor/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 2005.  
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

Intellectual Property Rights .....	4
Foreword.....	4
1 Scope .....	5
2 References .....	5
3 Definitions and abbreviations.....	5
3.1 Definitions .....	5
3.2 Abbreviations .....	6
4 Test Suite Structure (TSS).....	6
5 Test Purposes (TP) .....	7
5.1 Introduction .....	7
5.1.1 TP naming convention .....	7
5.1.2 Source of TP definition.....	7
5.1.3 Test strategy.....	7
5.2 TPs for the Mobility SCF .....	7
5.2.1 User Location.....	8
5.2.1.1 IpUserLocation interface.....	8
5.2.1.2 IpTrigerredUserLocation interface.....	20
5.2.2 User Location Camel .....	38
5.2.3 User Location Emergency .....	47
5.2.4 User status.....	51
5.3 TPs for the application using the Mobility SCF .....	56
5.3.1 User Location.....	56
5.3.1.1 IpAppUserLocation interface .....	56
5.3.1.2 IpAppTrigerredUserLocation interface.....	63
5.3.2 User Location Camel .....	73
5.3.3 User Location Emergency .....	81
5.3.4 User Status .....	85
History .....	90

---

## 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>).

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).

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 interactions 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 <http://docbox.etsi.org/Reference>.

- [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
M	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);
  - IpTriggerredUserLocation interface (TUL) (02).
- User Location Camel (ULC) (03).
- User Location Emergency (ULE) (04).
- User Status (US) (05).

## 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).

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

Identifier:	<suite_id>_<group>_<nnn>
<suite_id>	= SCG name: "M" for <b>M</b> obility 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 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".

## 5.2.1 User Location

### 5.2.1.1 IpUserLocation interface

#### Test M\_UL\_01

Summary: all methods, successful

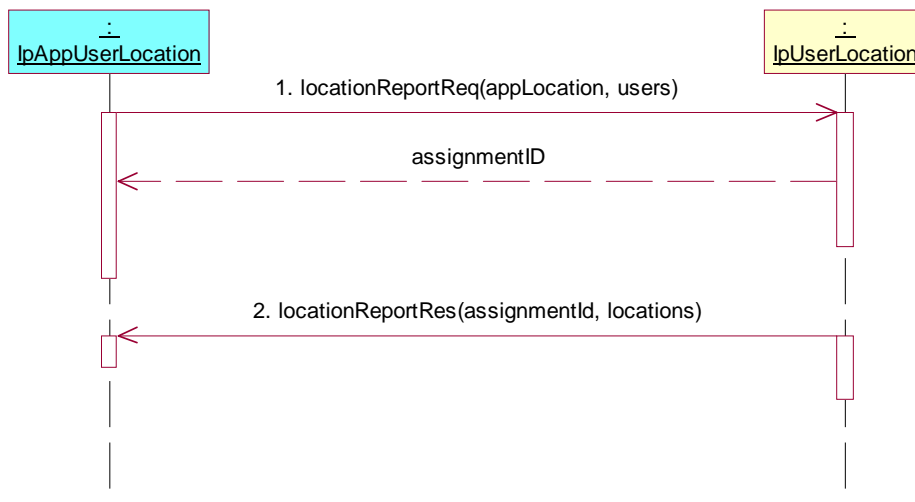
Reference: ES 201 915-6 [1], clause 8.1

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.

Test Sequence:

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





**Test M\_UL\_02**

Summary: locationReportReq, locationReportErr

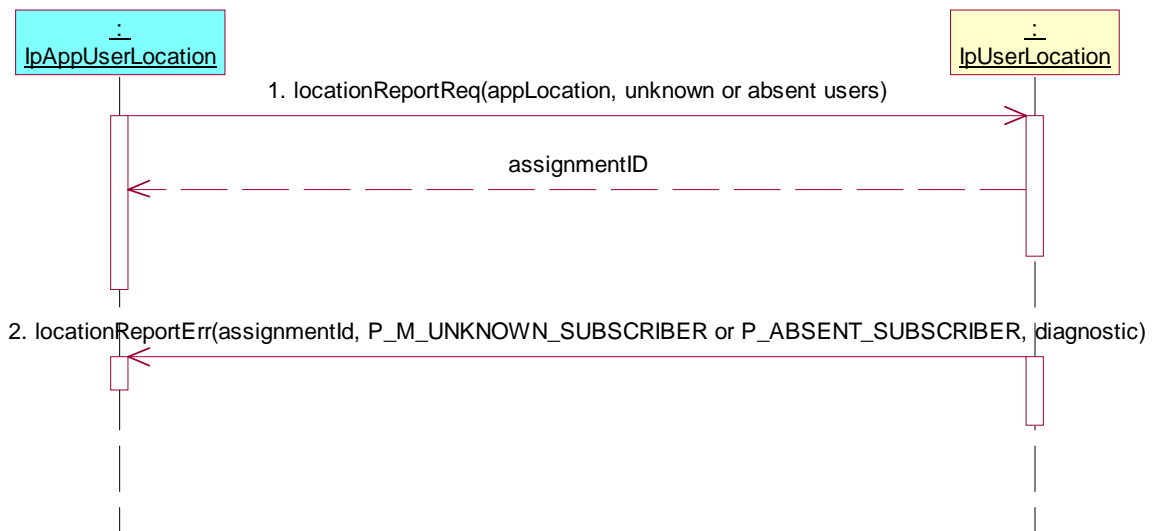
Reference: ES 201 915-6 [1], clause 8.1

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

Preamble: Registration of the IUT (User Location SCF) and the tester (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
2. Triggered action: cause IUT to call **locationReportErr()** method on the tester's (Application) **IpAppUserLocation** interface.  
 Parameters: assignmentId, cause indicating P\_M\_UNKNOWN\_SUBSCRIBER or P\_ABSENT\_SUBSCRIBER, diagnostic



**Test M\_UL\_03**

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.

Test Sequence:

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



**Test M\_UL\_04**

Summary: extendedLocationReportReq, extendedLocationReportErr

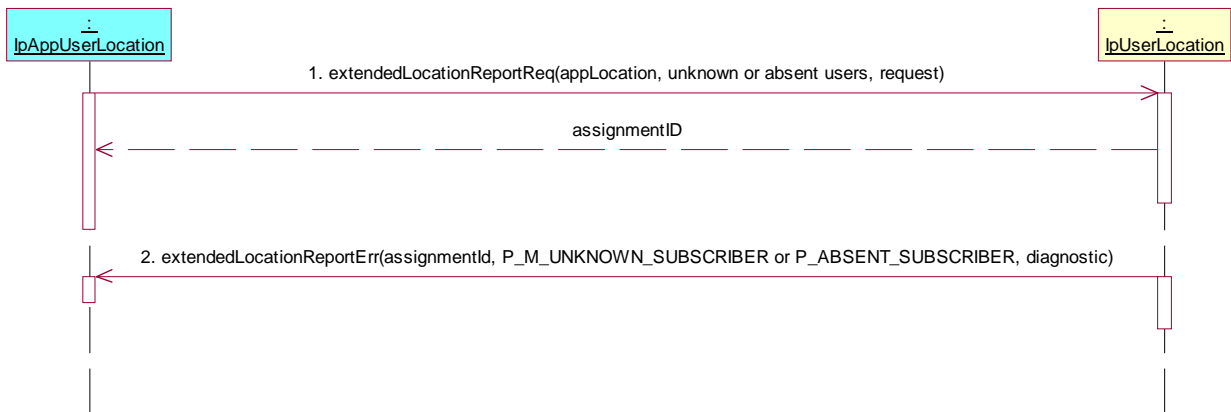
Reference: ES 201 915-6 [1], clause 8.1

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:

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



**Test M\_UL\_05**

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

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

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

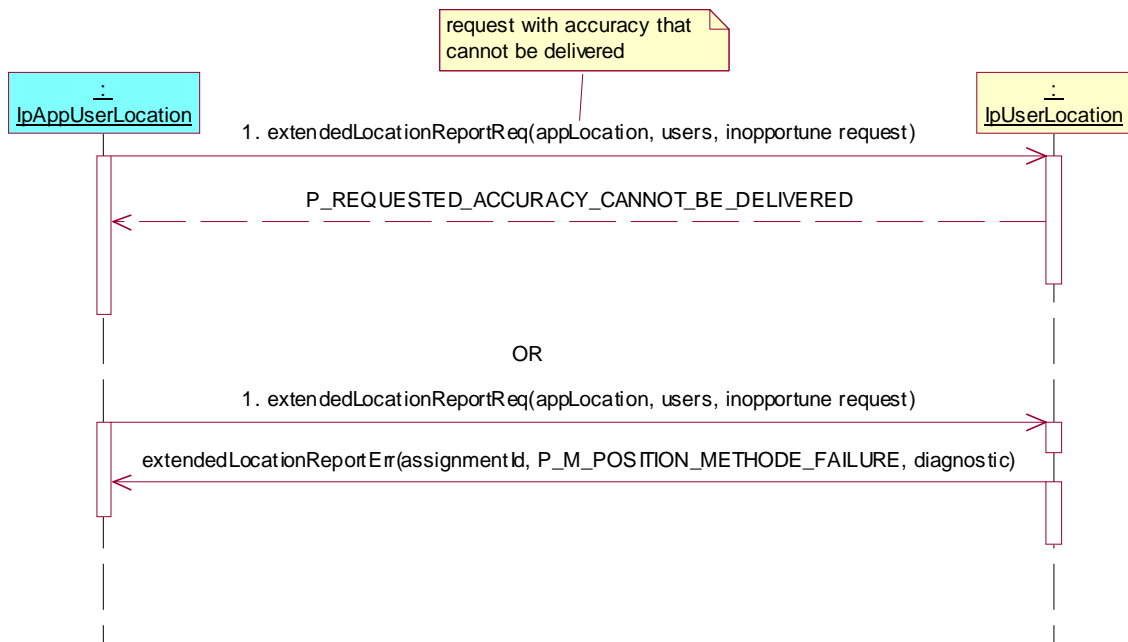
Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpUserLocation interface through selecting that service and signing the required service agreement.

Test Sequence:

1. Method call **extendedLocationReportReq()**

Parameters: appLocation, users, request with 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.



**Test M\_UL\_06**

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

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

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

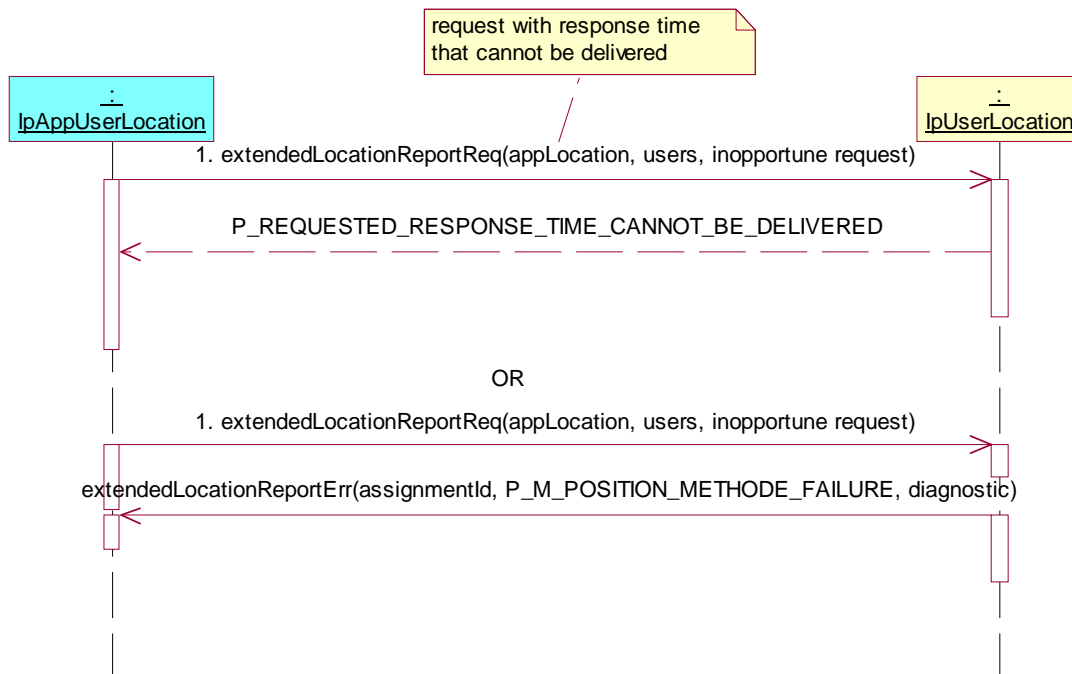
Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpUserLocation interface through selecting that service and signing the required service agreement.

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.



**Test M\_UL\_07**

Summary: all methods, successful

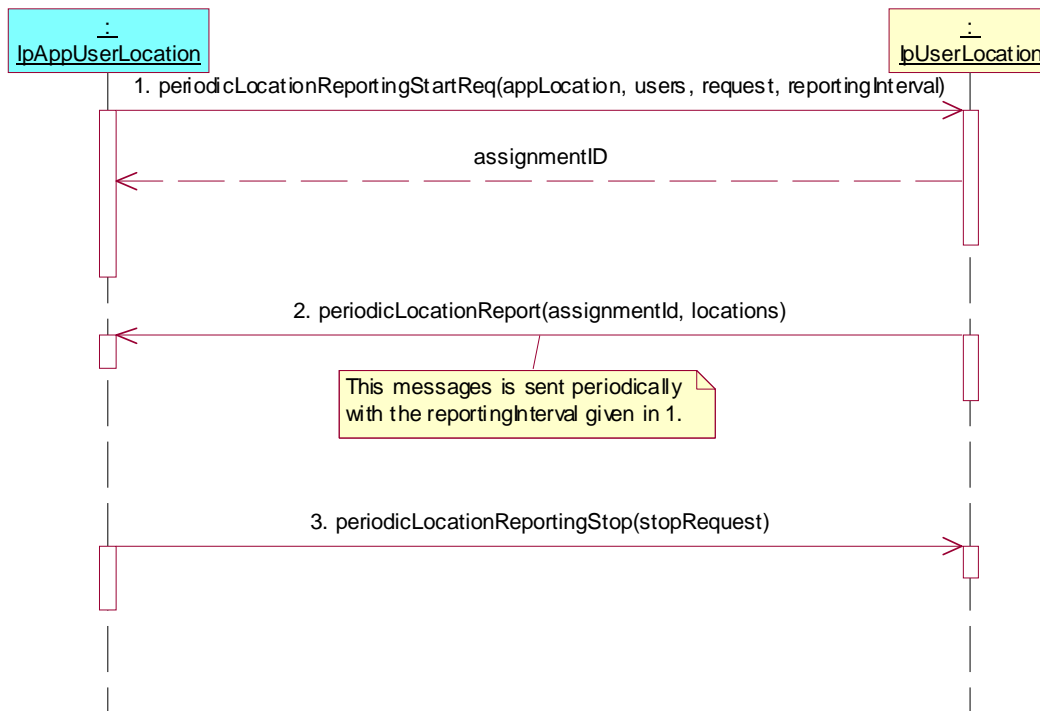
Reference: ES 201 915-6 [1], clause 8.1

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.

Test Sequence:

1. Method call **periodicLocationReportingStartReq()**  
 Parameters: appLocation, users, request, reportingInterval  
 Check: valid value of TpSessionID is returned
2. 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.



**Test M\_UL\_08**

Summary: all methods, unknown or absent subscriber

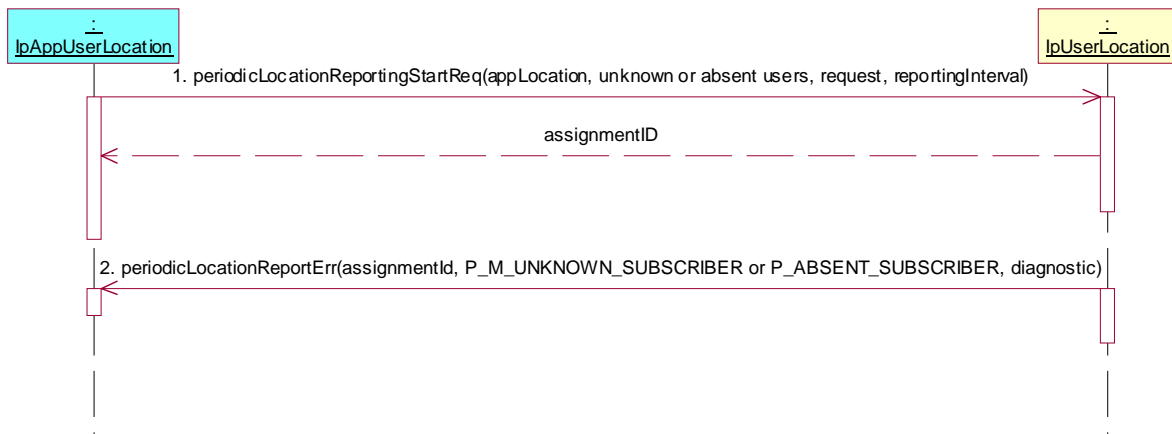
Reference: ES 201 915-6 [1], clause 8.1

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

Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the 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 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



**Test M\_UL\_09**

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

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

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

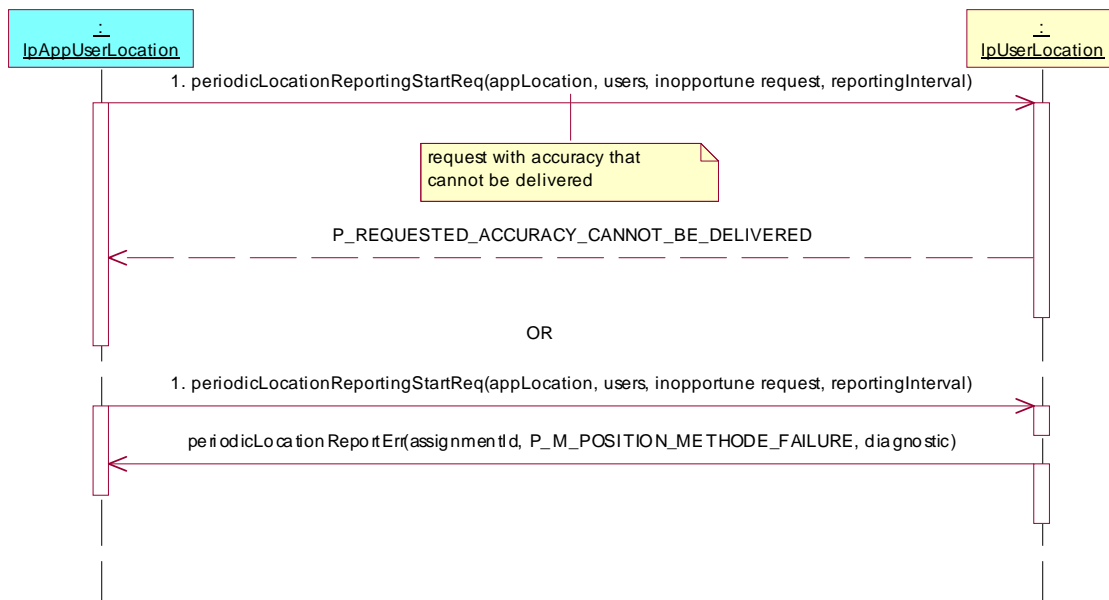
Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpUserLocation interface through selecting that service and signing the required service agreement.

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.





**Test M\_UL\_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] UL3

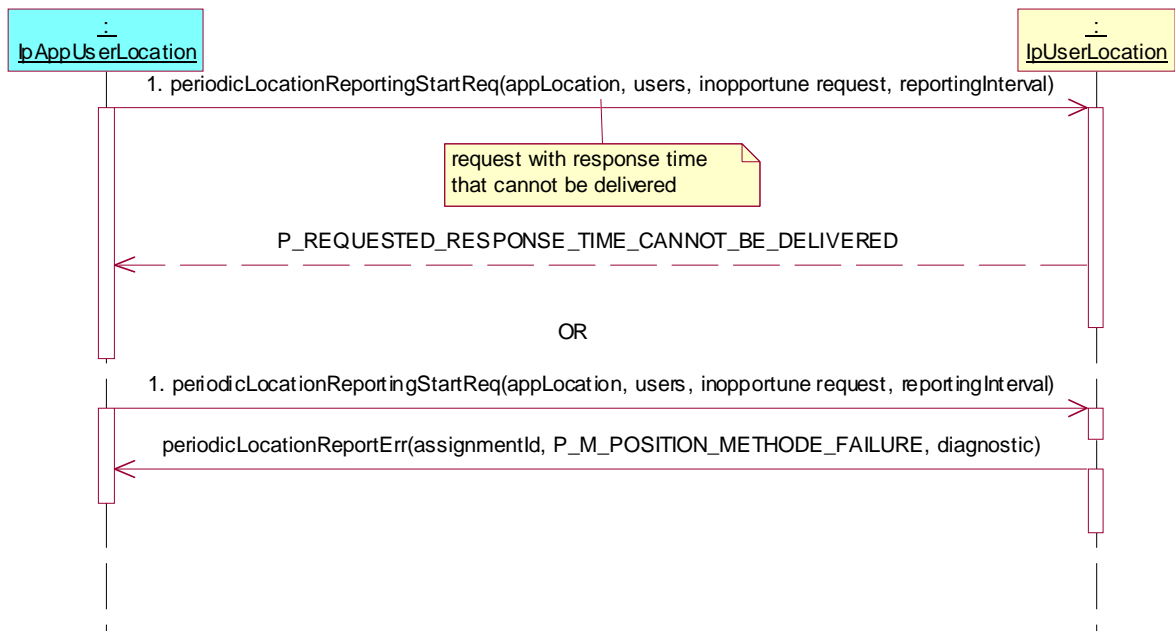
Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpUserLocation interface through selecting that service and signing the required service agreement.

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.



**Test M\_UL\_11**

Summary: periodicLocationReportingStartReq, P\_INVALID\_REPORTING\_INTERVAL

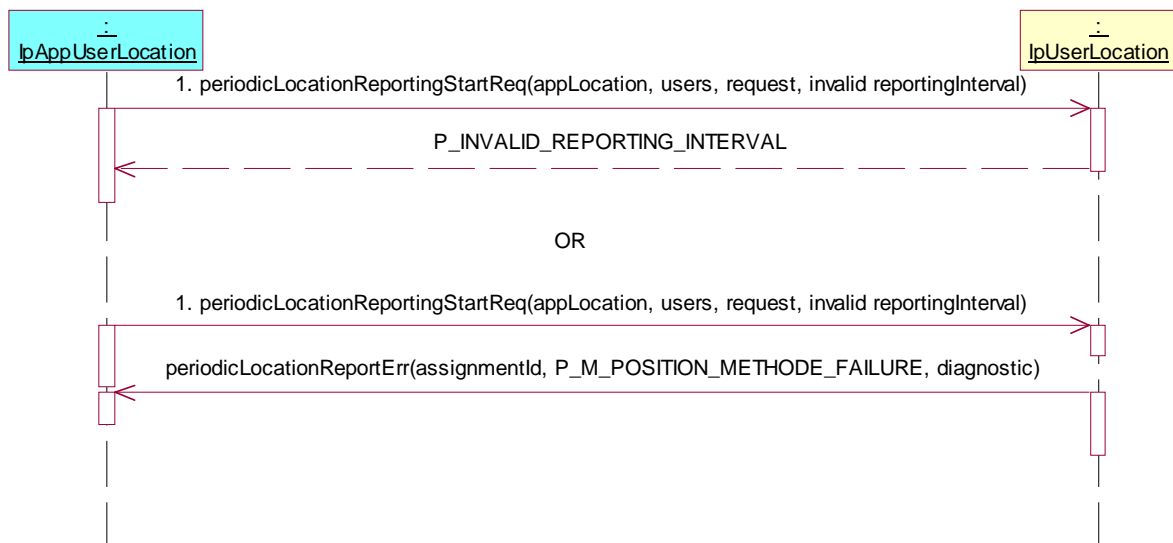
Reference: ES 201 915-6 [1], clause 8.1

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.

Test Sequence:

- 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.



**Test M\_UL\_12**

Summary: periodicLocationReportingStop, P\_INVALID\_ASSIGNMENT\_ID

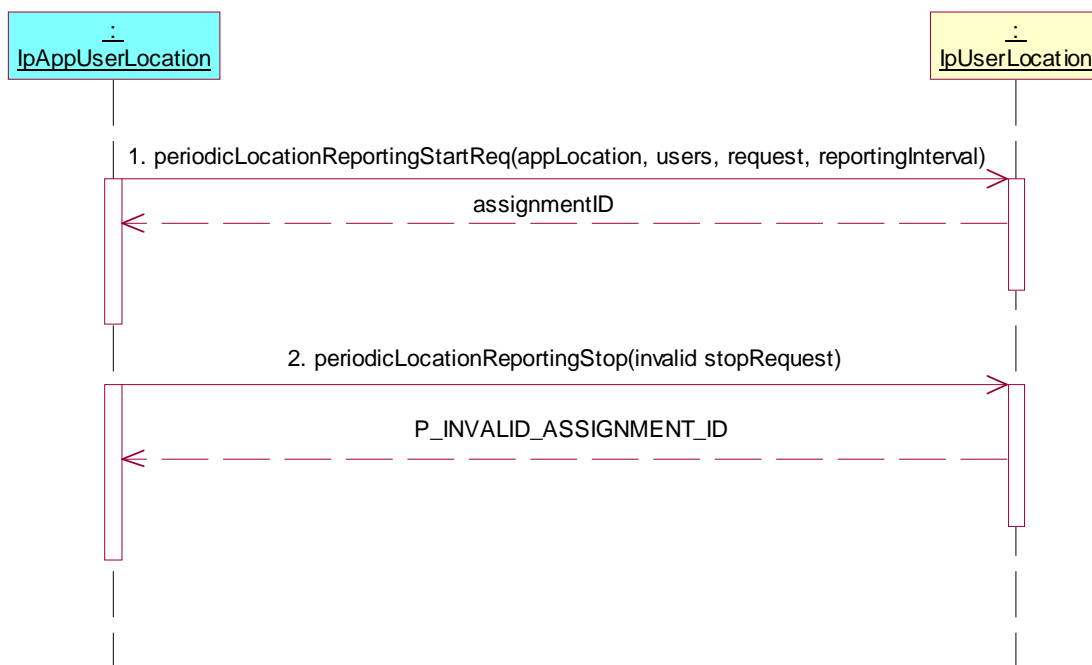
Reference: ES 201 915-6 [1], clause 8.1

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.

Test Sequence:

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 IpTriggerredUserLocation interface

#### Test M\_TUL\_01

Summary: all methods, successful

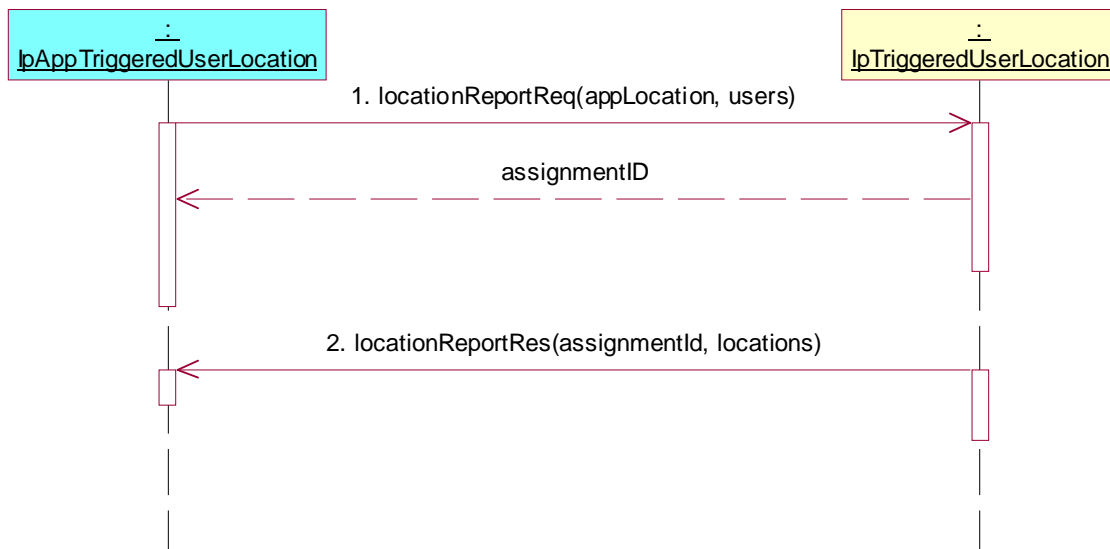
Reference: ES 201 915-6 [1], clause 8.1

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 IpTriggerredUserLocation interface through selecting that service and signing the required service agreement.

Test Sequence:

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



**Test M\_TUL\_02**

Summary: locationReportReq, locationReportErr

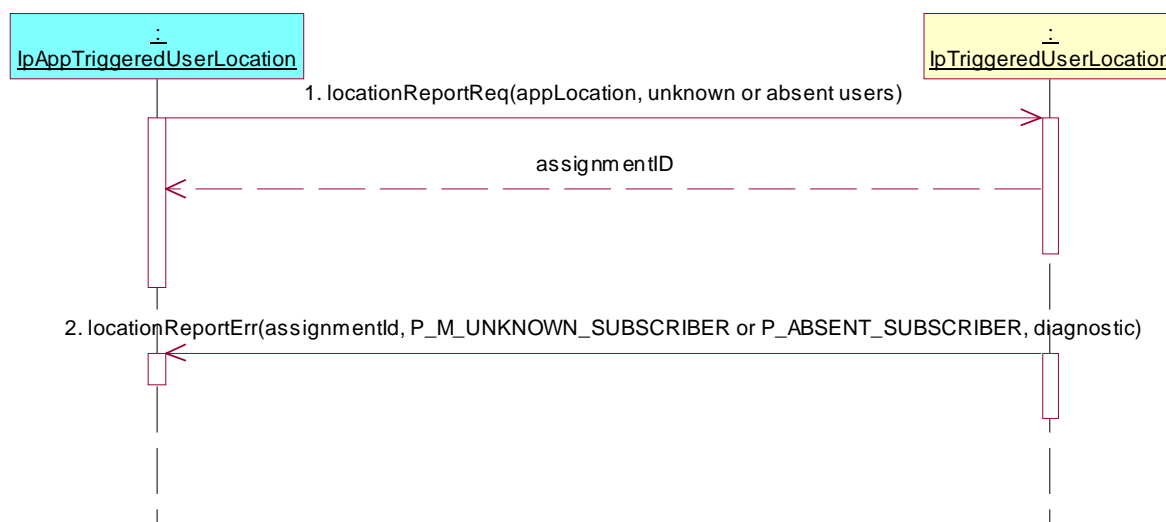
Reference: ES 201 915-6 [1], clause 8.1

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

Preamble: Registration of the IUT (User Location SCF) and the tester (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
2. 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



**Test M\_TUL\_03**

Summary: all methods, successful

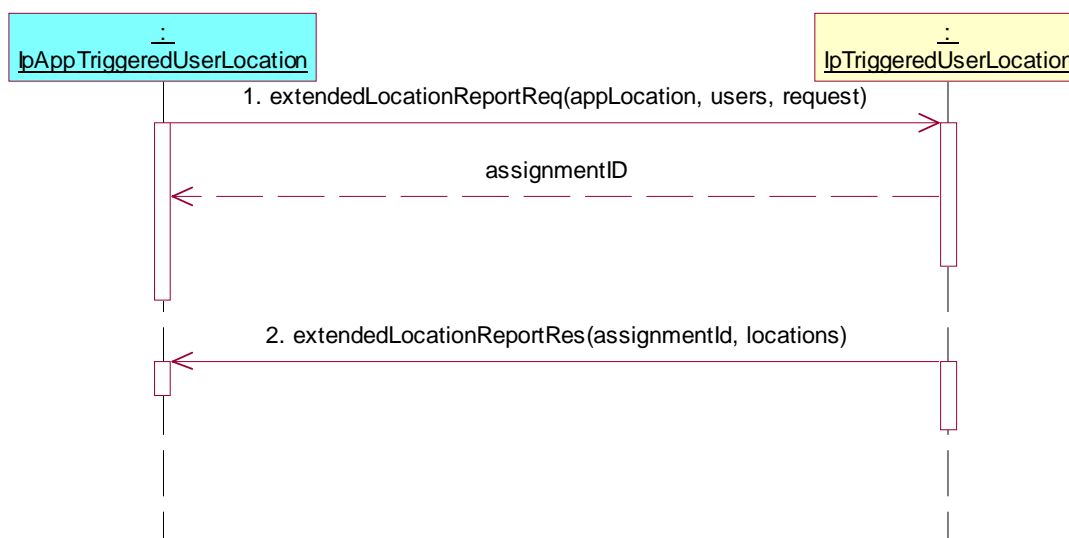
Reference: ES 201 915-6 [1], clause 8.1

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
2. Triggered action: cause IUT to call **extendedLocationReportRes()** method on the tester's (Application) **IpAppTriggeredUserLocation** interface.  
 Parameters: assignmentId, locations



**Test M\_TUL\_04**

Summary: extendedLocationReportReq, extendedLocationReportErr

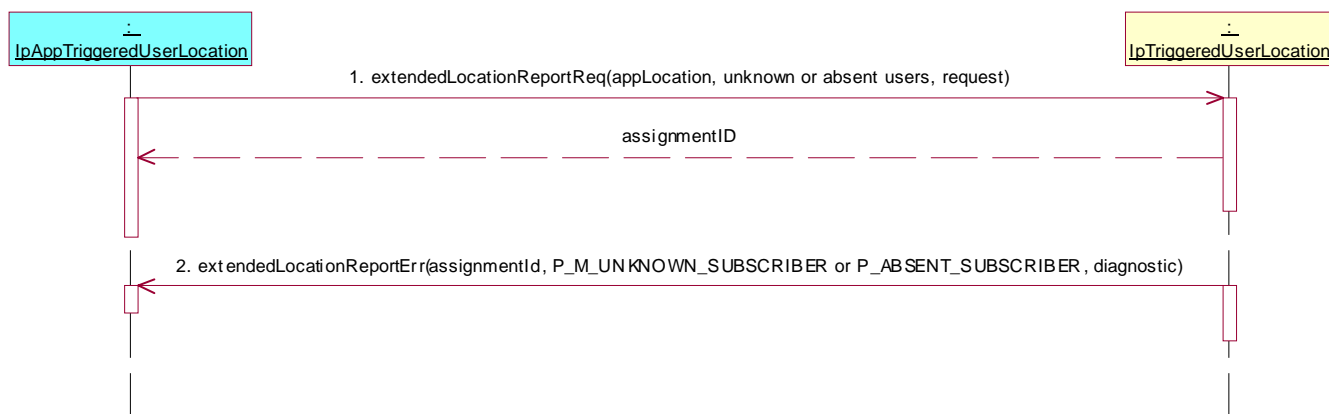
Reference: ES 201 915-6 [1], clause 8.1

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:

1. Method call **extendedLocationReportReq()**  
 Parameters: appLocation, users with unknown or absent subscriber, request  
 Check: valid value of TpSessionID is returned
2. 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



**Test M\_TUL\_05**

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

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

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

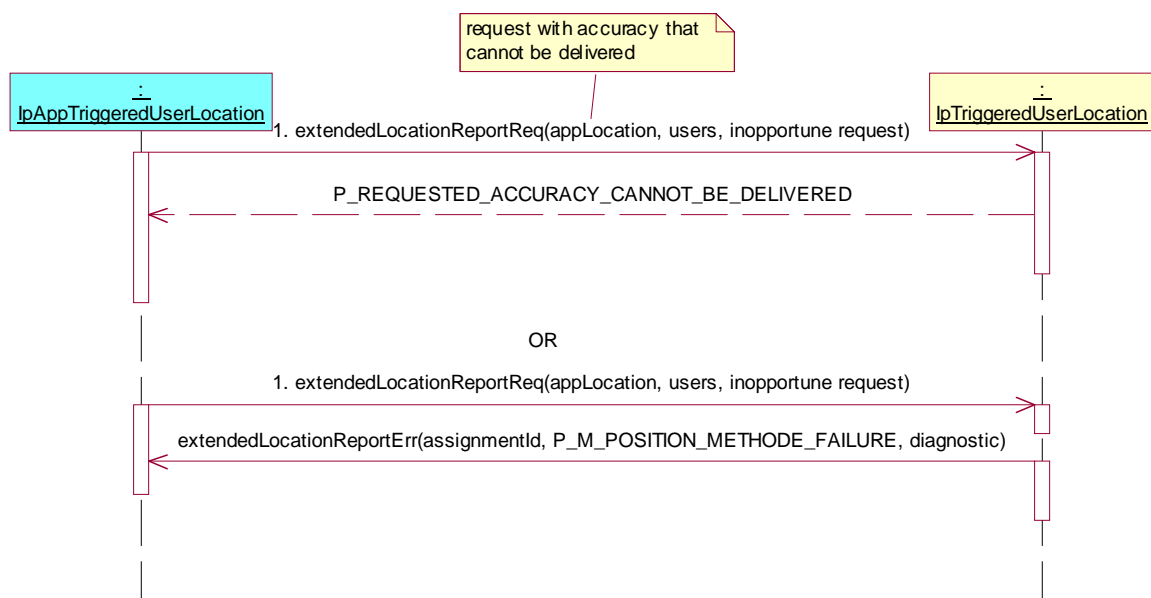
Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpTriggeredUserLocation interface through selecting that service and signing the required service agreement.

Test Sequence:

1. Method call **extendedLocationReportReq()**

Parameters: appLocation, users, request with 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.





**Test M\_TUL\_06**

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

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

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

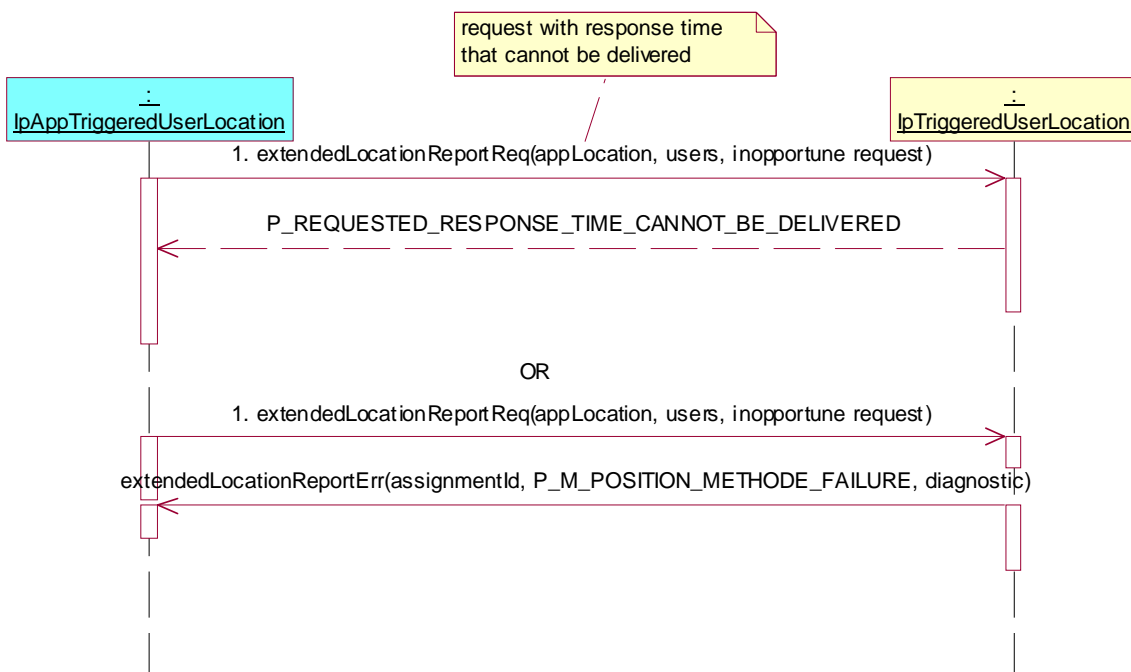
Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpTriggeredUserLocation interface through selecting that service and signing the required service agreement.

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.



**Test M\_TUL\_07**

Summary: all methods, successful

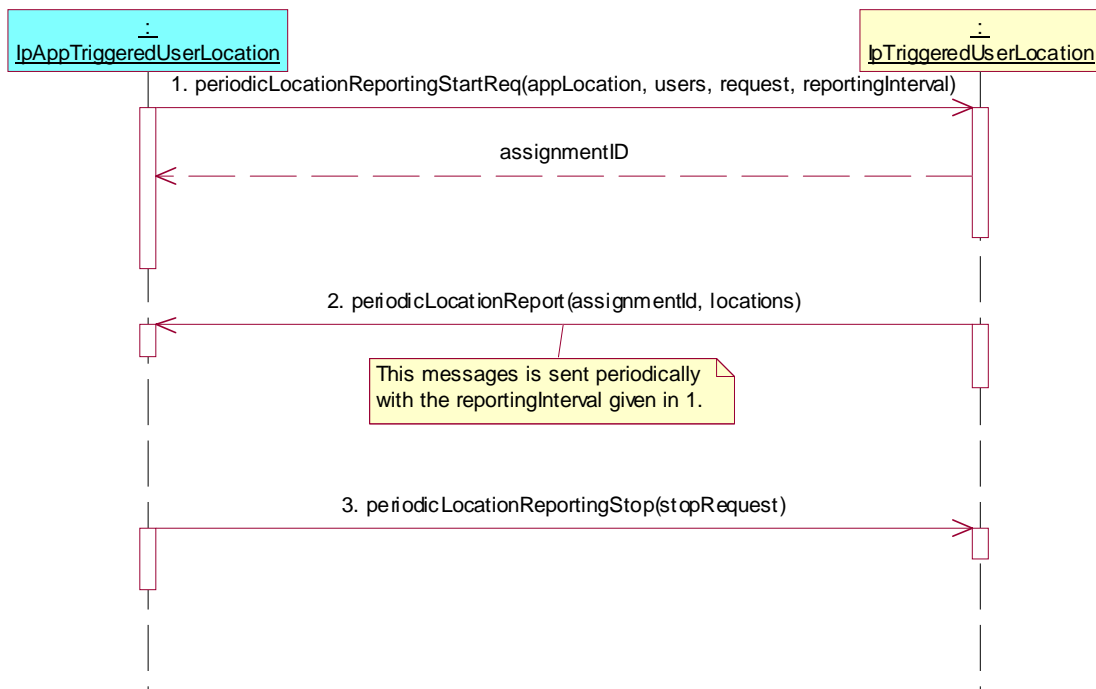
Reference: ES 201 915-6 [1], clause 8.1

Selection: periodicLocationReportingStartReq 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, reportingInterval  
Check: valid value of TpSessionID is returned
2. 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.



**Test M\_TUL\_08**

Summary: all methods, unknown or absent subscriber

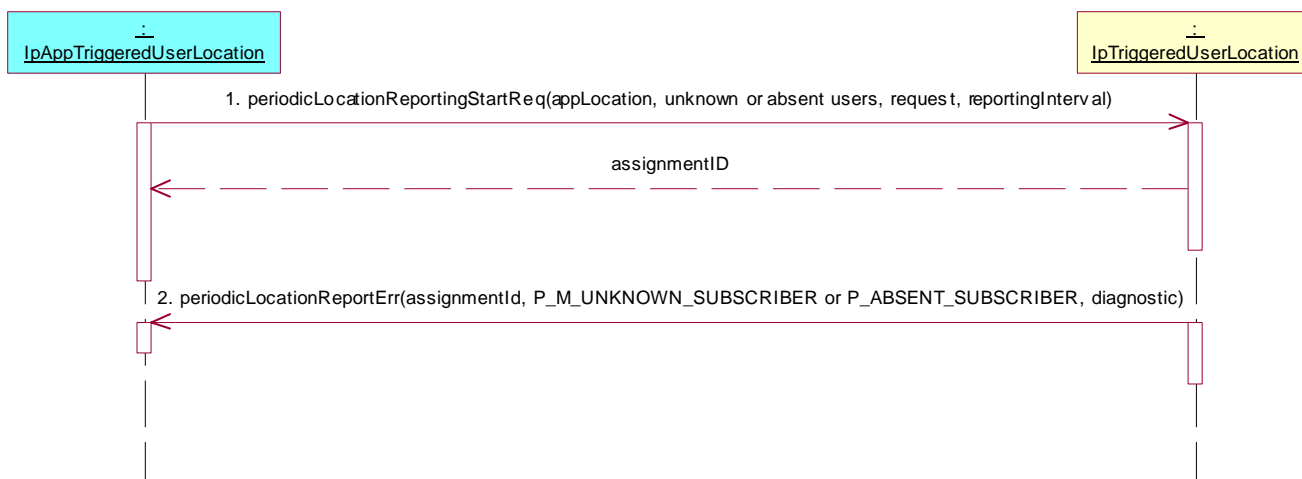
Reference: ES 201 915-6 [1], clause 8.1

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.

Test Sequence:

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) **IpAppTriggeredUserLocation** interface.  
 Parameters: assignmentId, cause indicating P\_M\_UNKNOWN\_SUBSCRIBER or P\_ABSENT\_SUBSCRIBER, diagnostic



**Test M\_TUL\_09**

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

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

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

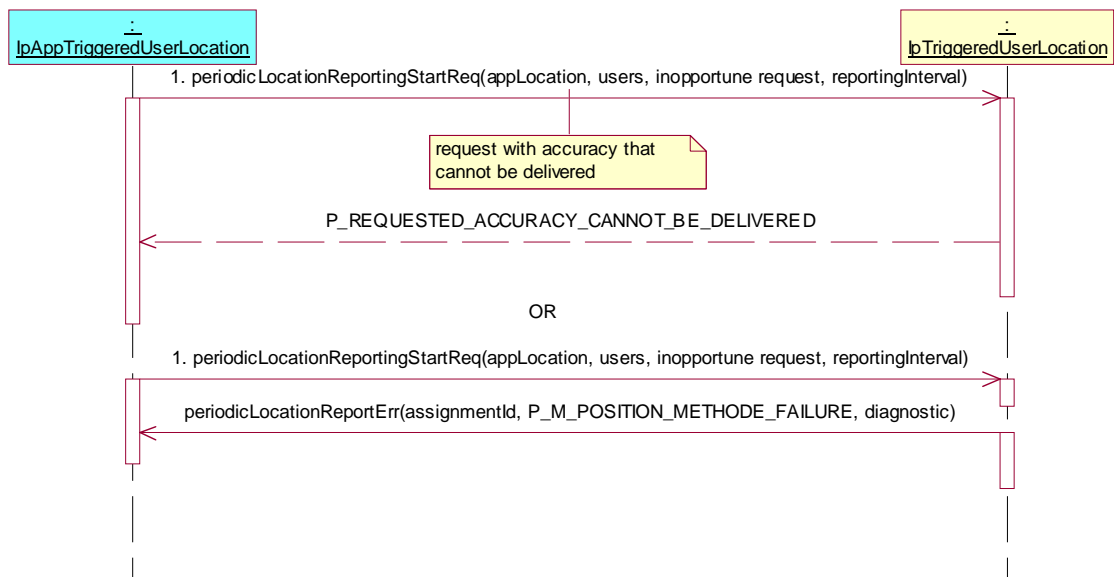
Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpTriggeredUserLocation interface through selecting that service and signing the required service agreement.

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.



**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

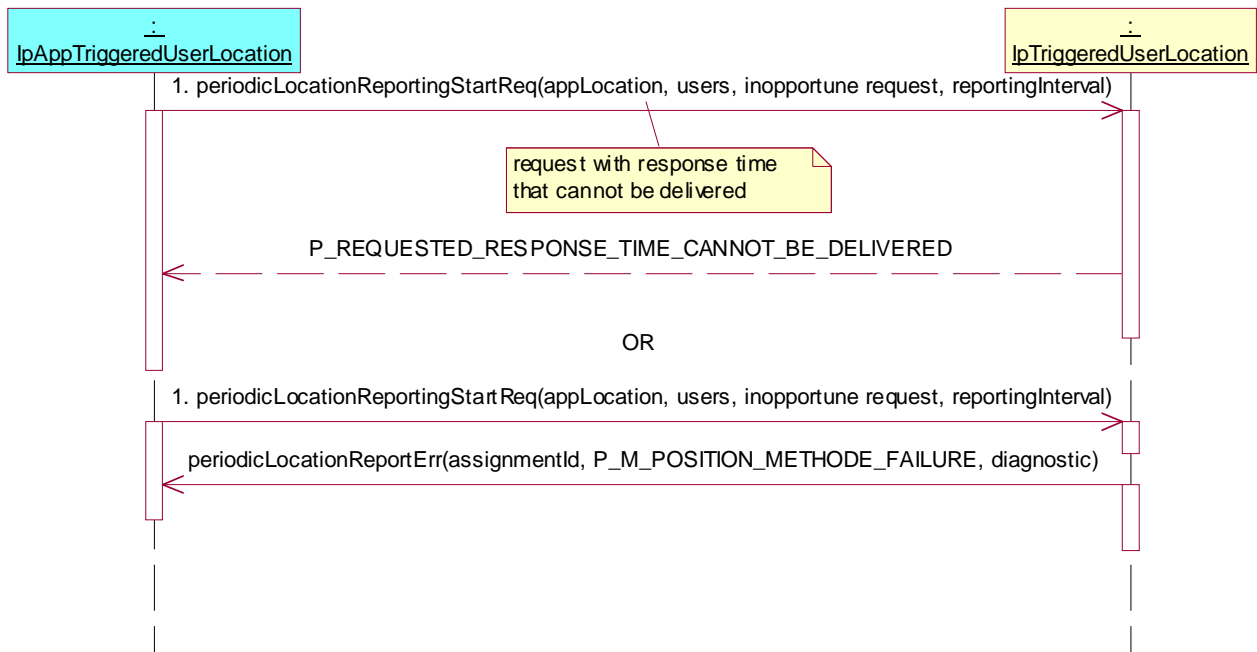
Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpTriggeredUserLocation interface through selecting that service and signing the required service agreement.

Test Sequence:

1. Method call **periodicLocationReportingStartReq ()**

Parameters: appLocation, users, request with 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.



**Test M\_TUL\_11**

Summary: periodicLocationReportingStartReq, P\_INVALID\_REPORTING\_INTERVAL

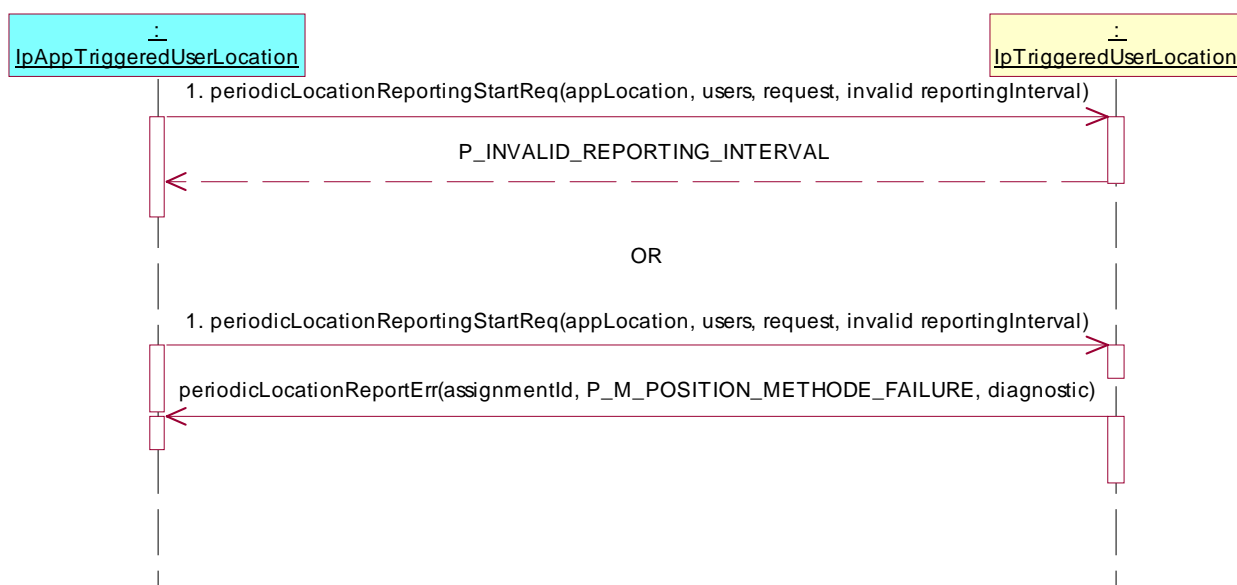
Reference: ES 201 915-6 [1], clause 8.1

Selection: periodicLocationReportingStartReq 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, invalid reportingInterval  
 Check: P\_INVALID\_REPORTING\_INTERVAL is returned, or periodicLocationReportErr() with P\_M\_POSITION\_METHOD\_FAILURE value of TpMobilityError.



**Test M\_TUL\_12**

Summary: periodicLocationReportingStop, P\_INVALID\_ASSIGNMENT\_ID

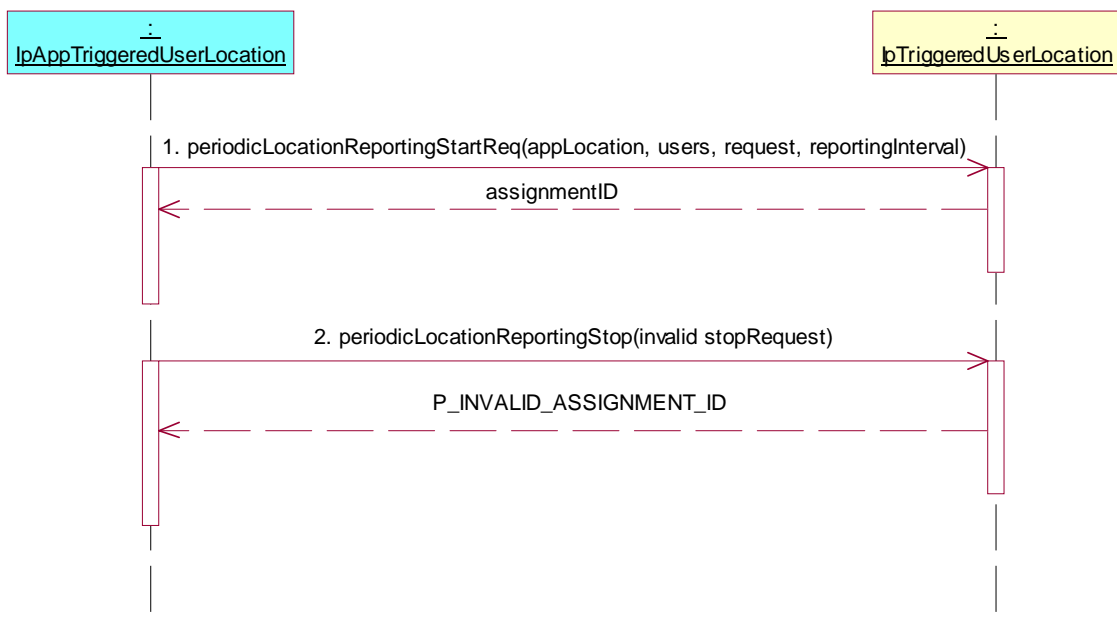
Reference: ES 201 915-6 [1], clause 8.1

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



**Test M\_TUL\_13**

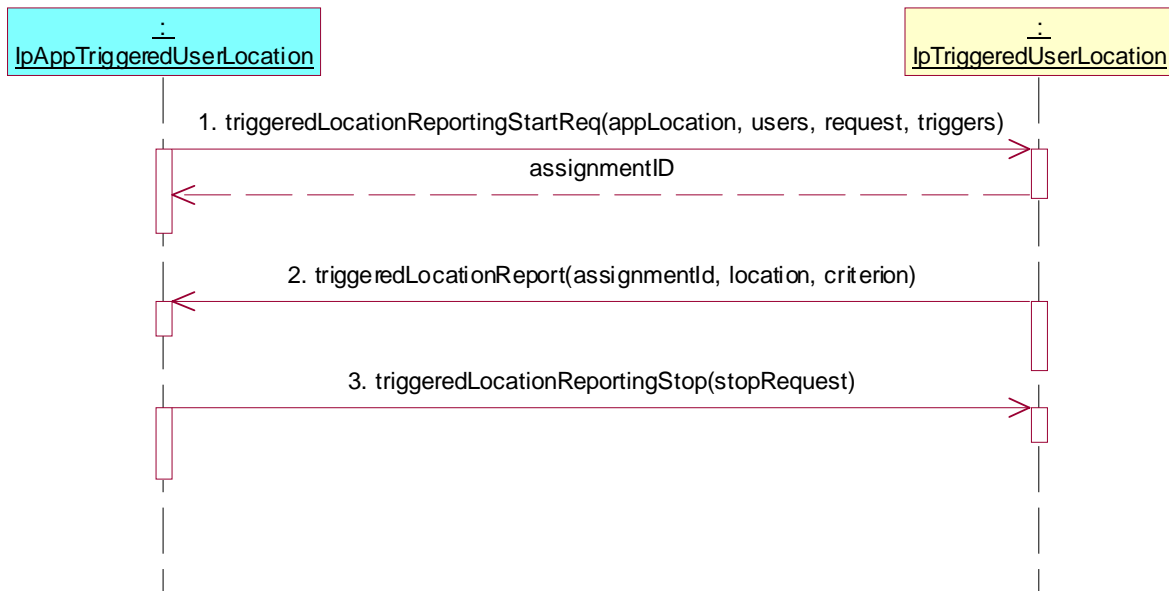
Summary: all methods, successful

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
2. 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





**Test M\_TUL\_14**

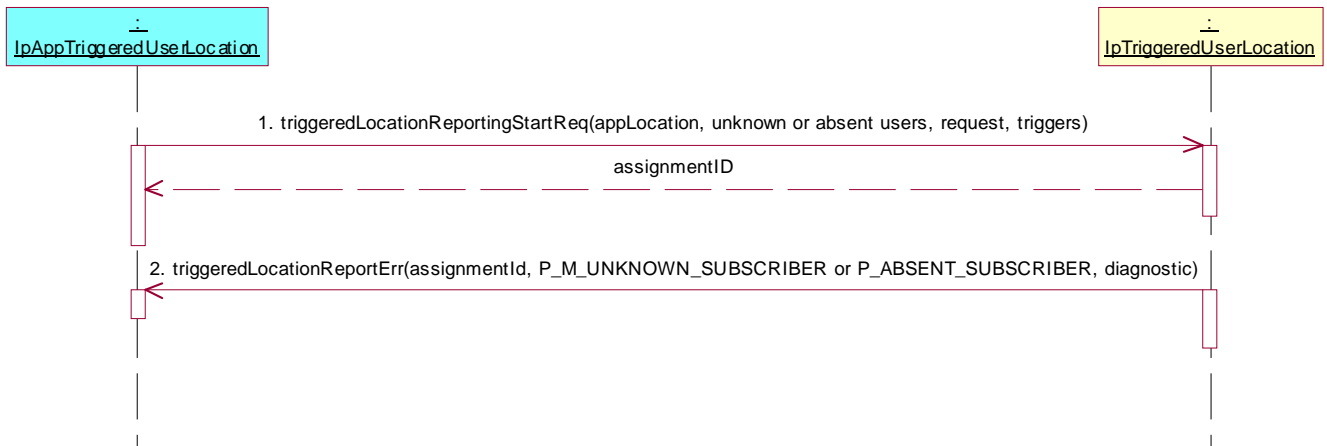
Summary: all methods, unknown or absent subscriber

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.

Test Sequence:

1. Method call **triggeredLocationReportingStartReq()**  
 Parameters: appLocation, users with unknown or absent subscriber, request, triggers  
 Check: valid value of TpSessionID is returned
2. 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



**Test M\_TUL\_15**

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

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

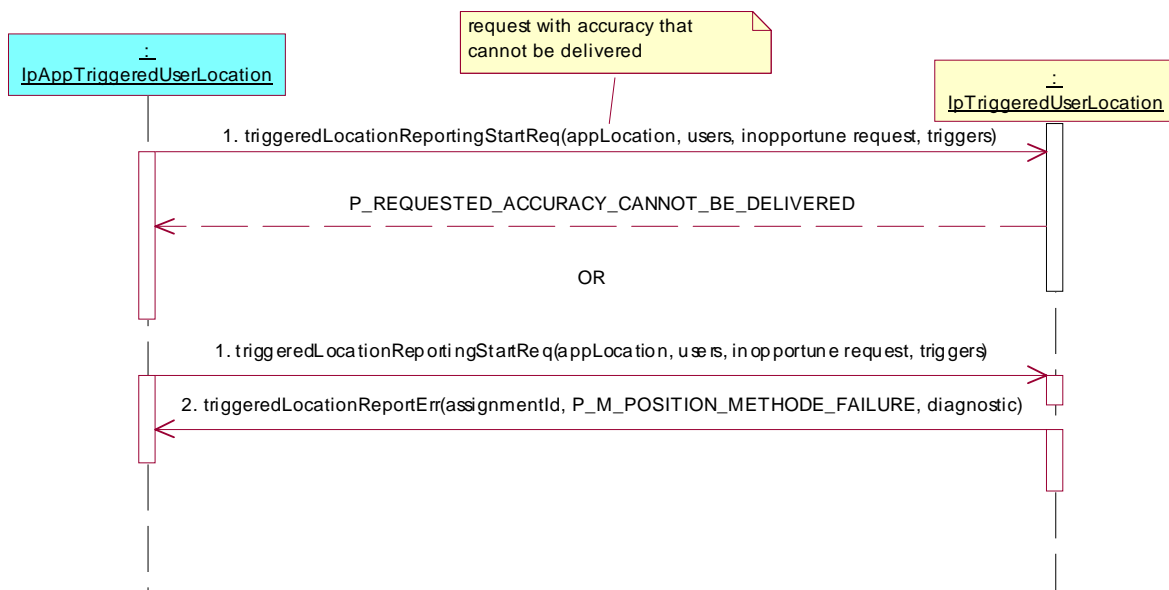
Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpTriggeredUserLocation interface through selecting that service and signing the required service agreement.

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.



**Test M\_TUL\_16**

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

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

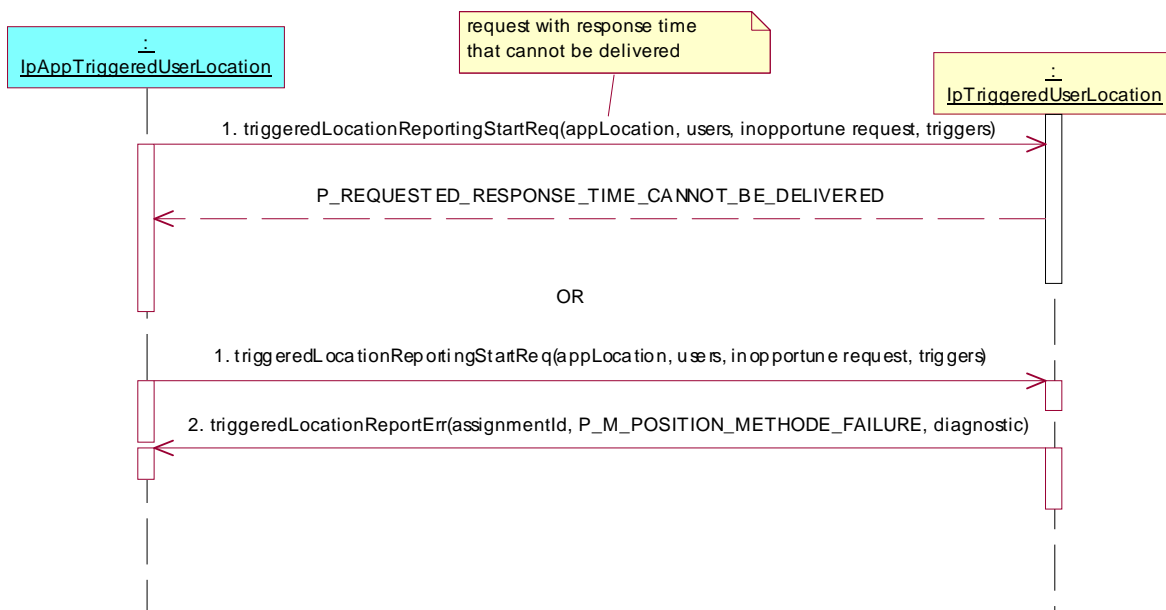
Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpTriggeredUserLocation interface through selecting that service and signing the required service agreement.

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.



**Test M\_TUL\_17**

Summary: triggeredLocationReportingStartReq, P\_TRIGGER\_CONDITIONS\_NOT\_SUBSCRIBED

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

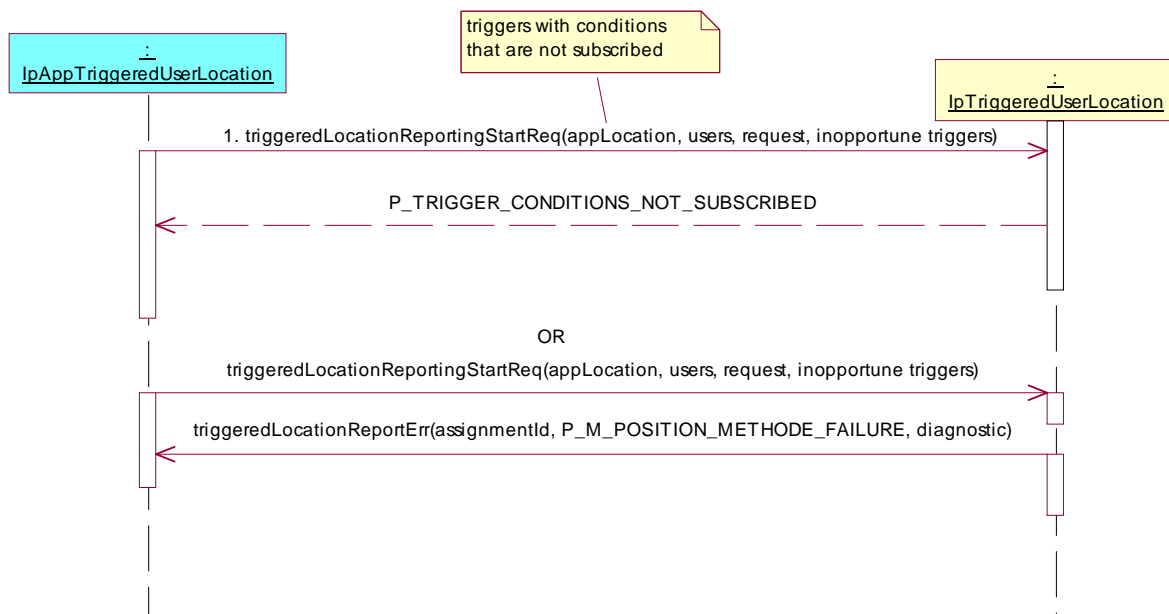
Preamble: Registration of the IUT (User Location SCF) and the tester (application) to the framework. The tester must have obtained a reference to an instance of the IpTriggeredUserLocation interface through selecting that service and signing the required service agreement.

Test Sequence:

1. 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.



**Test M\_TUL\_18**

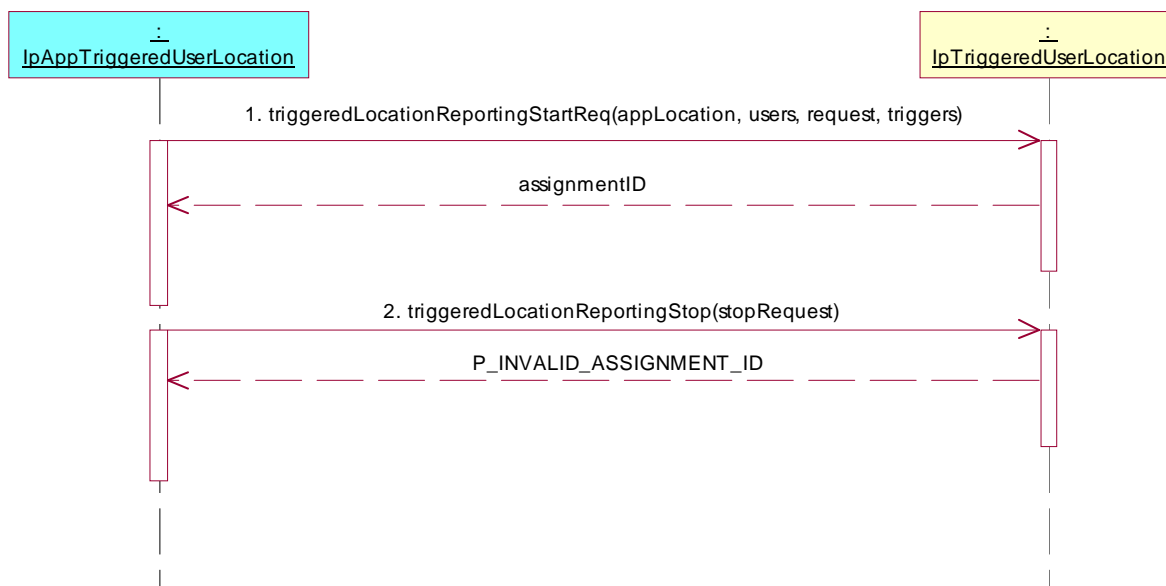
Summary: triggeredLocationReportingStop, P\_INVALID\_ASSIGNMENT\_ID

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.

Test Sequence:

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

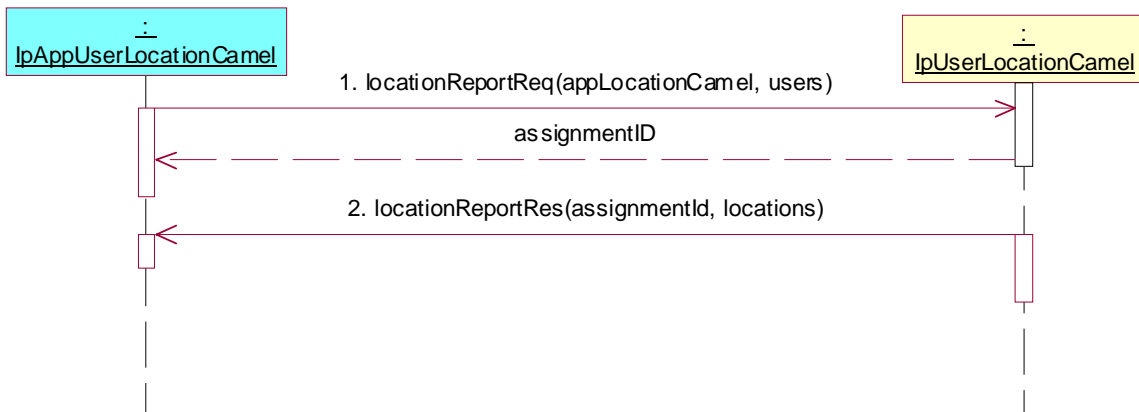
Reference: ES 201 915-6 [1], clause 8.2

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.

Test Sequence:

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



**Test M\_ULC\_02**

Summary: all methods, unknown or absent subscriber

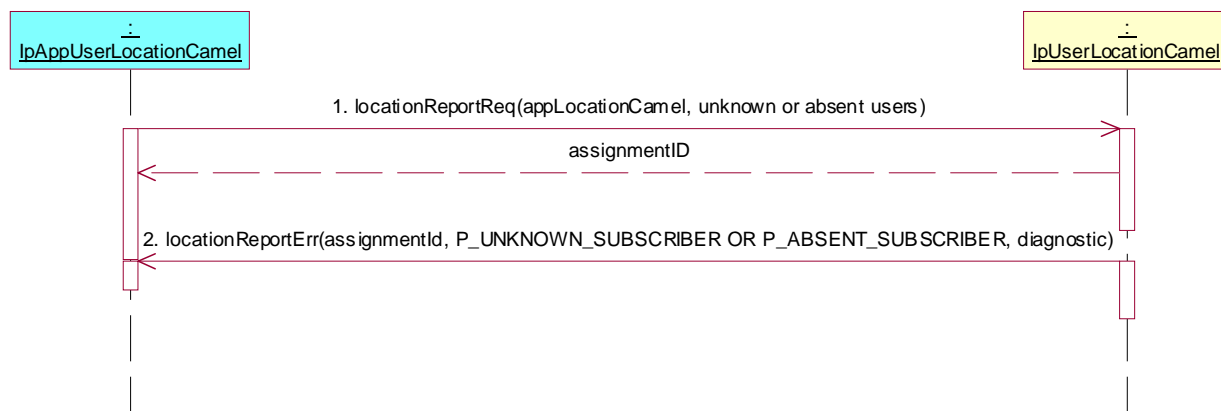
Reference: ES 201 915-6 [1], clause 8.2

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.

Test Sequence:

1. Method call **locationReportReq()**  
 Parameters: appLocationCamel, users with unknown or absent subscriber  
 Check: valid value of TpSessionID is returned
2. 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



**Test M\_ULC\_03**

Summary: all methods, successful

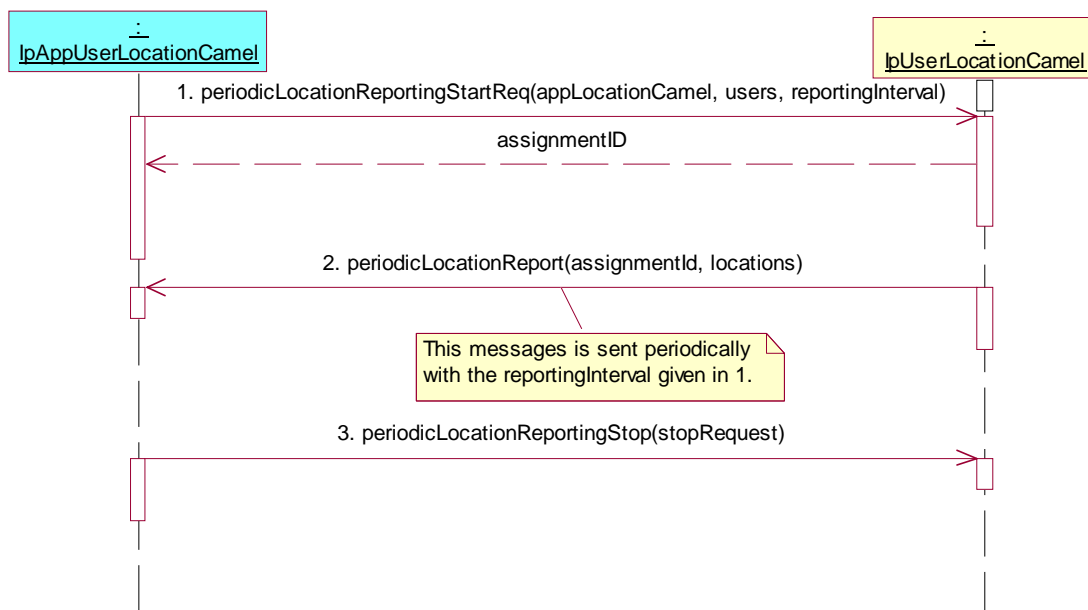
Reference: ES 201 915-6 [1], clause 8.2

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.

Test Sequence:

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.





**Test M\_ULC\_04**

Summary: all methods, unknown or absent subscriber

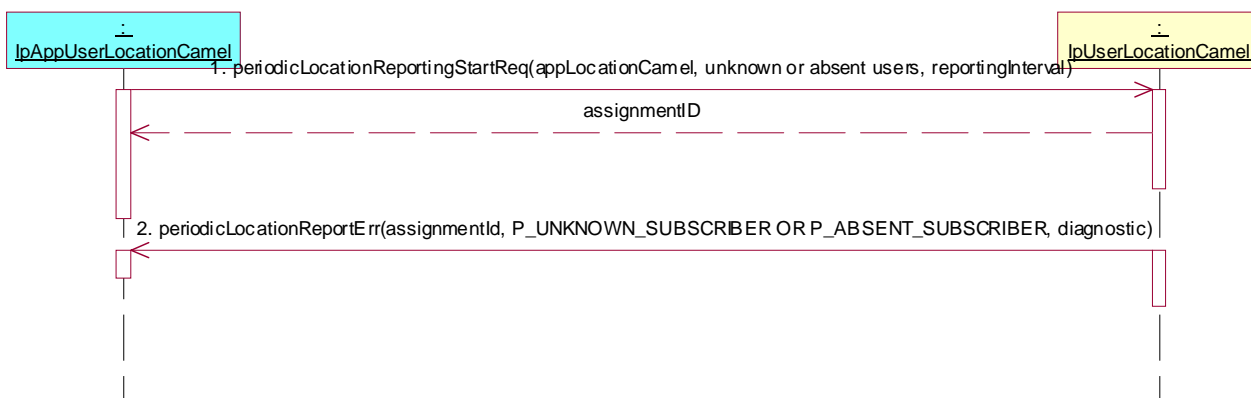
Reference: ES 201 915-6 [1], clause 8.2

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: 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



**Test M\_ULC\_05**

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.

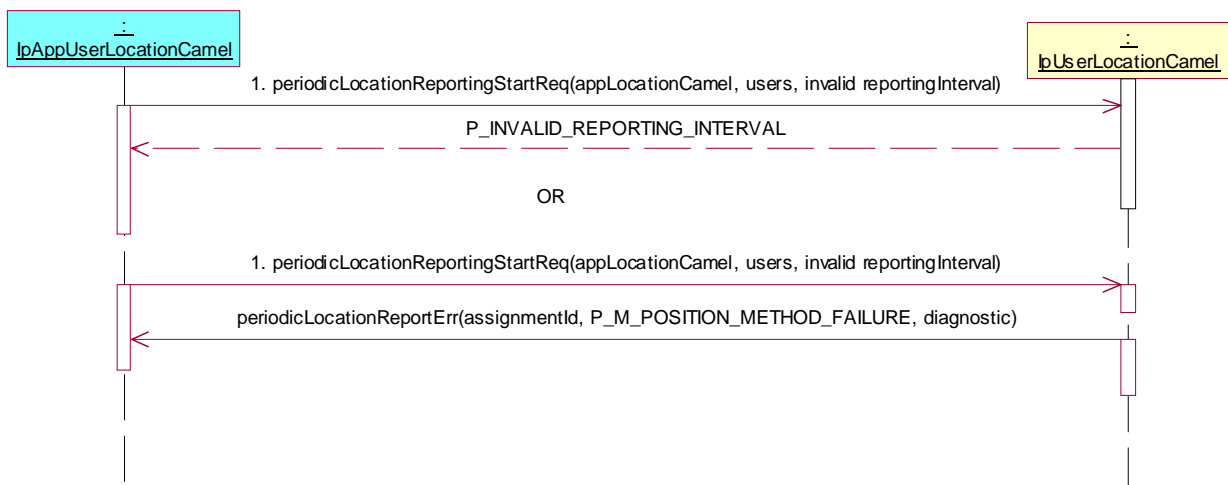
Selection Condition: periodicLocationReportingStartReq() supported.

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.



**Test M\_ULC\_06**

Summary: periodicLocationReportingStop, P\_INVALID\_ASSIGNMENT\_ID

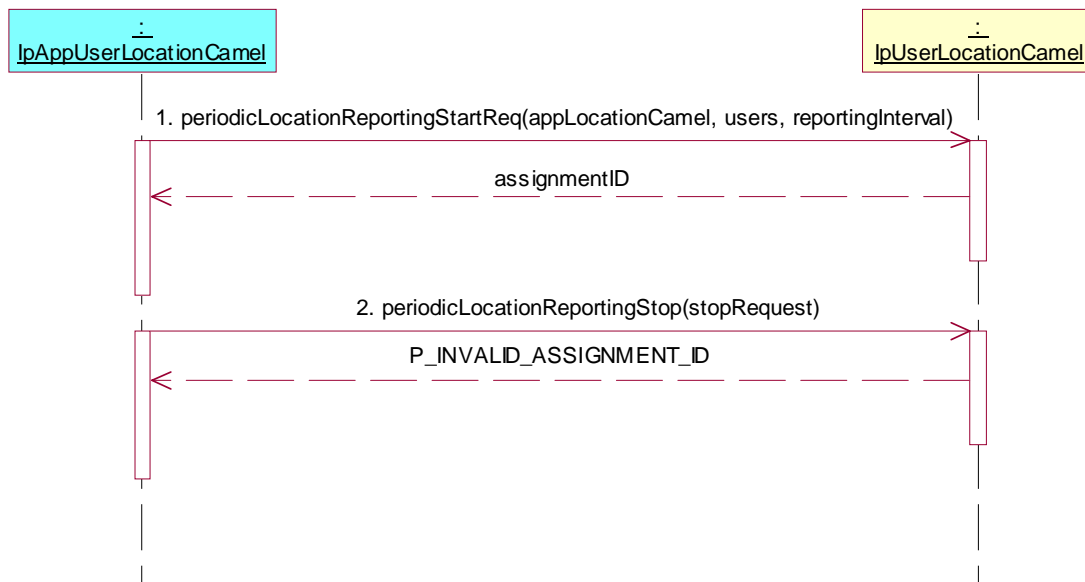
Reference: ES 201 915-6 [1], clause 8.2

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.

Selection Condition: periodicLocationReportingStartReq() supported.

Test Sequence:

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.



**Test M\_ULC\_07**

Summary: all methods, successful

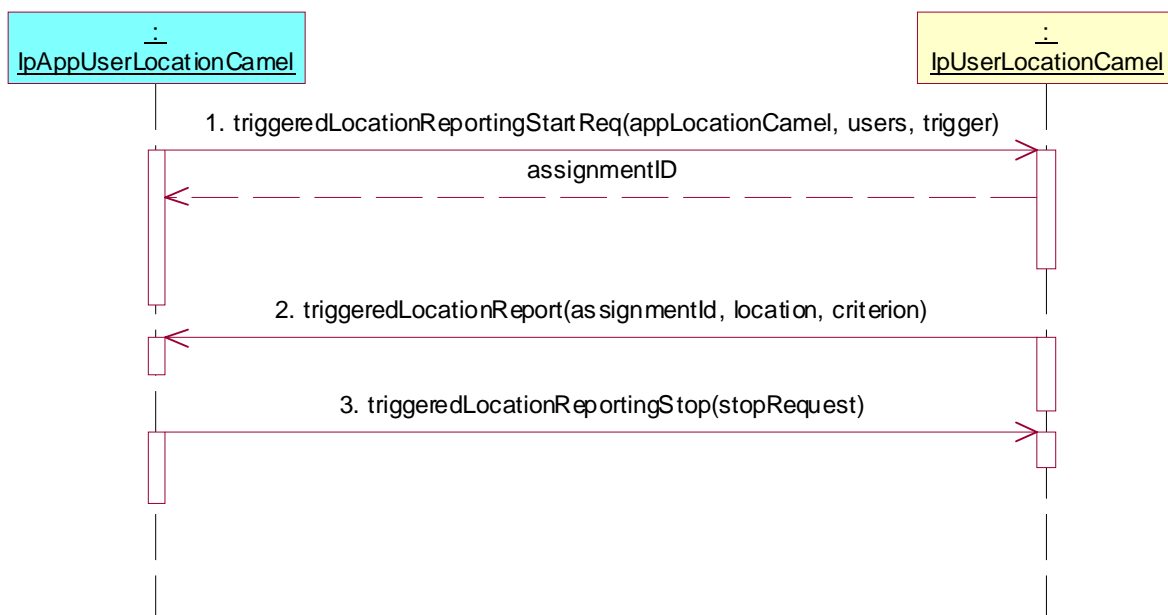
Reference: ES 201 915-6 [1], clause 8.2

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: triggeredLocationReportingStartReq() supported.

Test Sequence:

1. Method call **triggeredLocationReportingStartReq()**  
Parameters: appLocationCamel, users, trigger  
Check: valid value of TpSessionID is returned
2. 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



**Test M\_ULC\_08**

Summary: all methods, unknown or absent subscriber

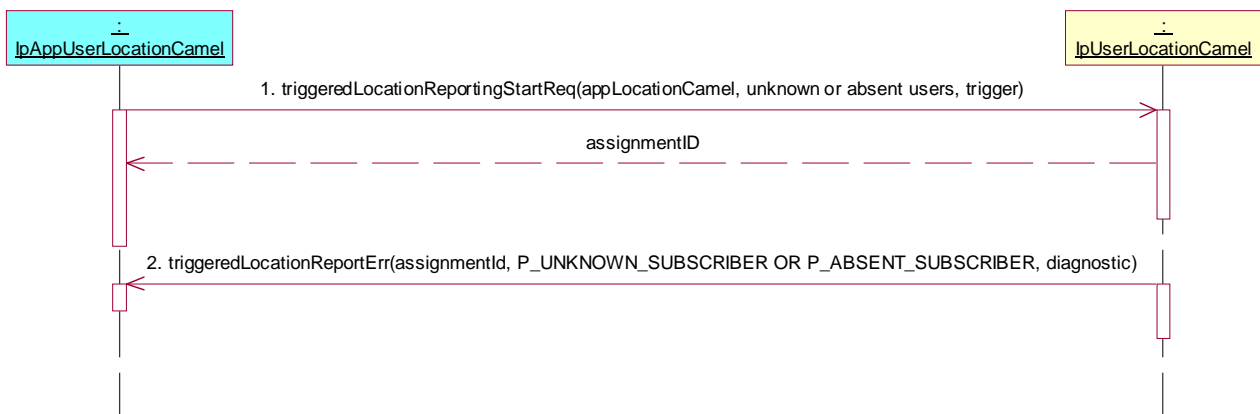
Reference: ES 201 915-6 [1], clause 8.2

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
2. Triggered action: cause IUT to call **triggeredLocationReportErr ()** method on the tester's (Application) **IpAppUserLocationCamel** interface.  
 Parameters: assignmentId, cause indicating P\_M\_UNKNOWN\_SUBSCRIBER or P\_ABSENT\_SUBSCRIBER, diagnostic



**Test M\_ULC\_10**

Summary: triggeredLocationReportingStop, P\_INVALID\_ASSIGNMENT\_ID

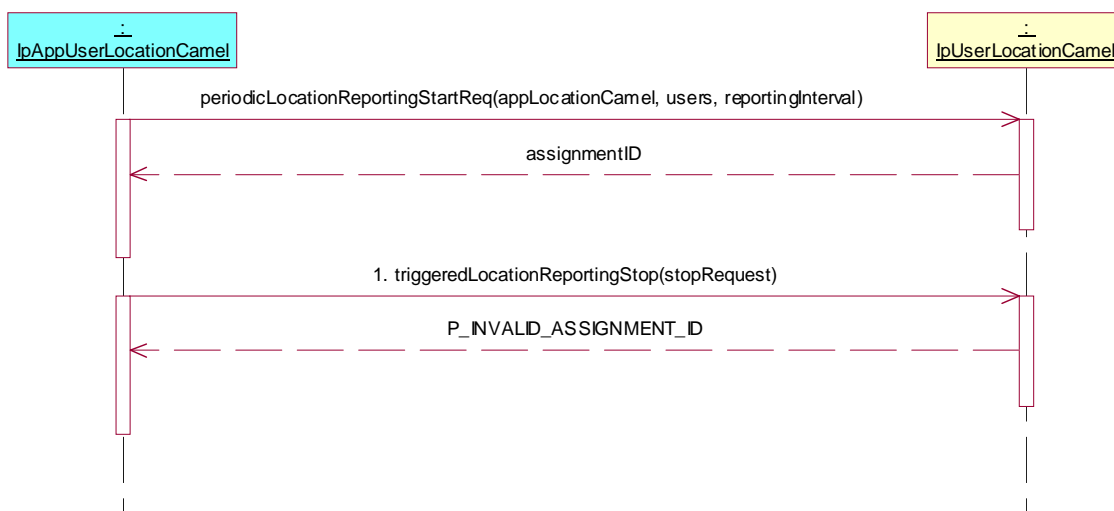
Reference: ES 201 915-6 [1], clause 8.2

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.

Selection Condition: triggeredLocationReportingStartReq() supported.

Test Sequence:

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

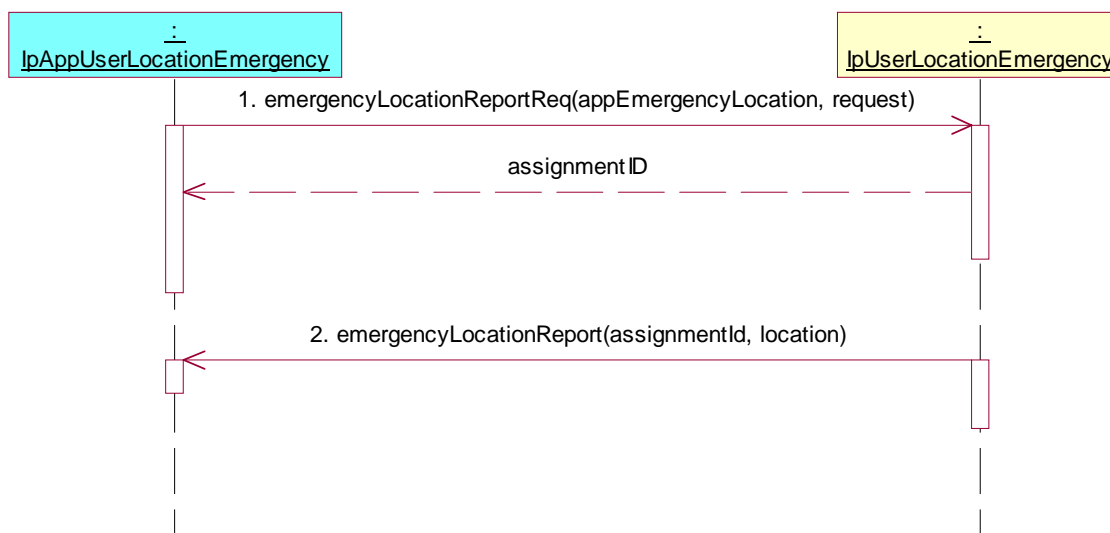
Reference: ES 201 915-6 [1], clause 8.3

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.

Selection Condition: emergencyLocationReportReq() supported.

Test Sequence:

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



**Test M\_ULE\_02**

Summary: all methods, unknown or absent subscriber

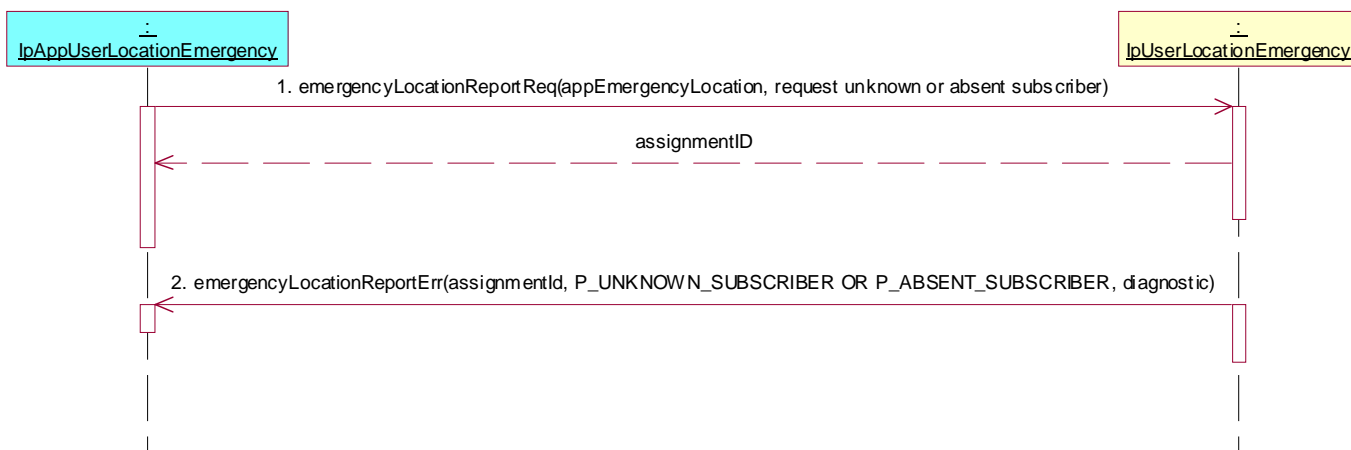
Reference: ES 201 915-6 [1], clause 8.3

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
2. 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





**Test M\_ULE\_03**

Summary: all methods, successful

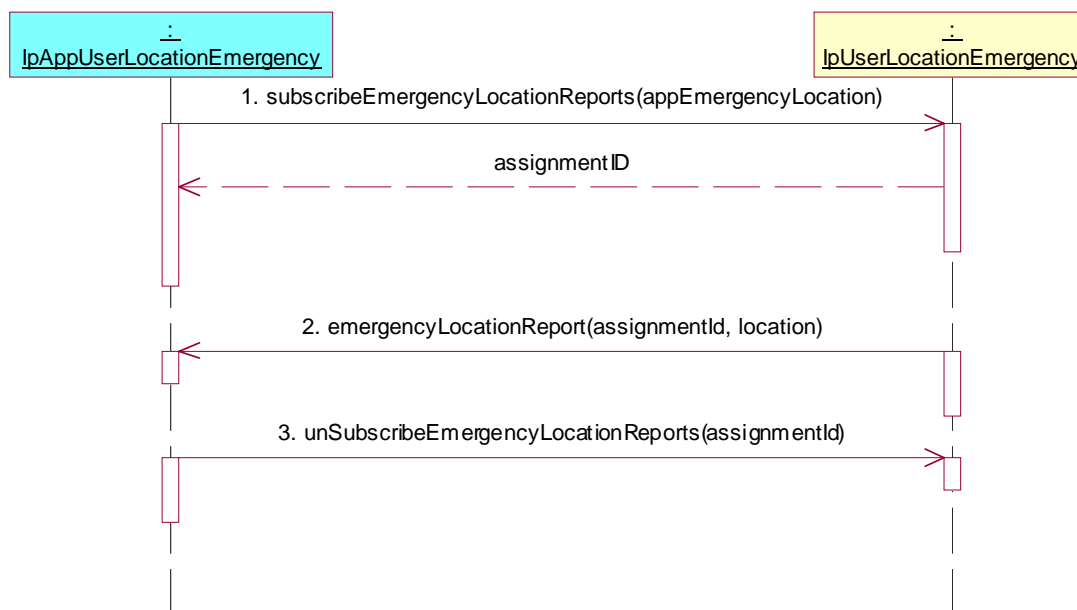
Reference: ES 201 915-6 [1], clause 8.3

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.

Selection Condition: subscribeEmergencyLocationReports() supported.

Test Sequence:

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



**Test M\_ULE\_04**

Summary: unSubscribeEmergencyLocationReports, P\_INVALID\_ASSIGNMENT\_ID

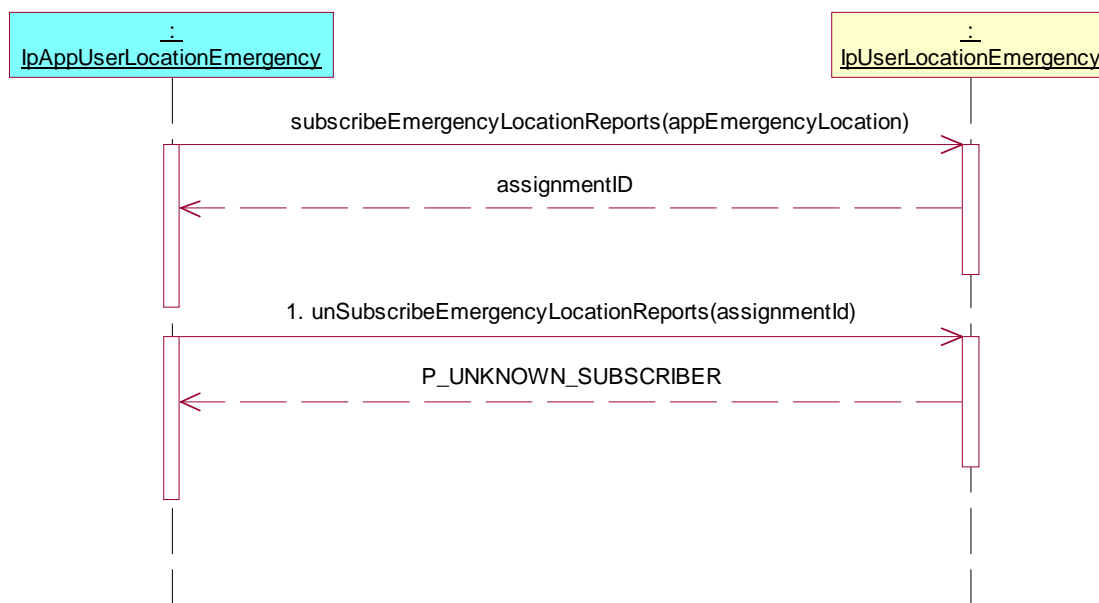
Reference: ES 201 915-6 [1], clause 8.3

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.

Selection Condition: subscribeEmergencyLocationReports() supported.

Test Sequence:

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



## 5.2.4 User status

### Test M\_US\_01

Summary: all methods, successful

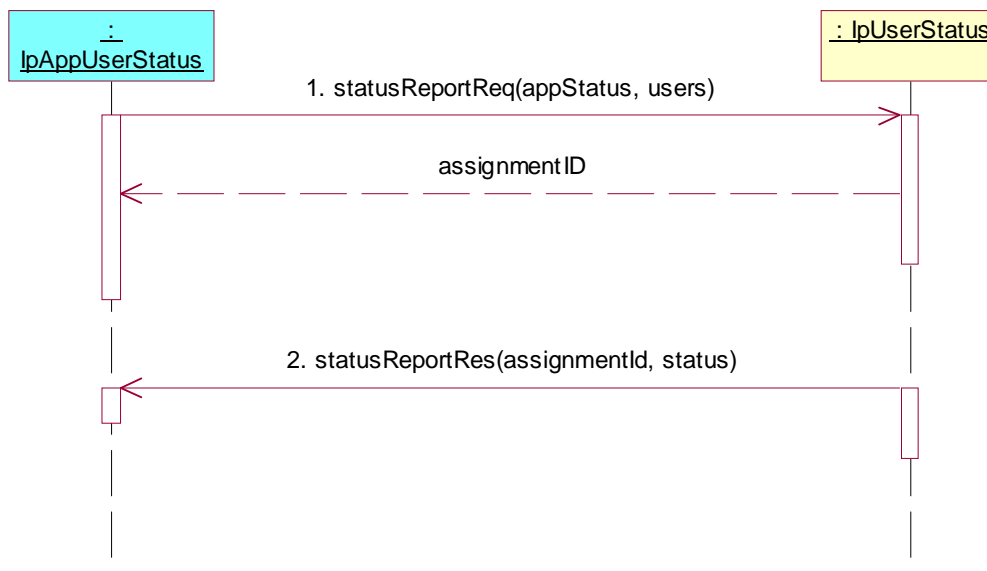
Reference: ES 201 915-6 [1], clause 8.4

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

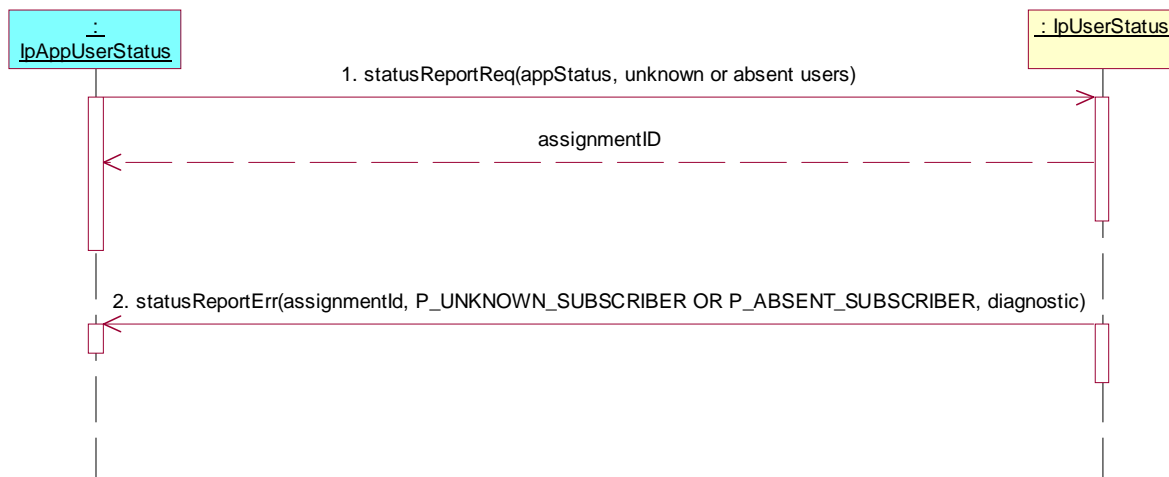
Reference: ES 201 915-6 [1], clause 8.4

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



**Test M\_US\_03**

Summary: all methods, successful

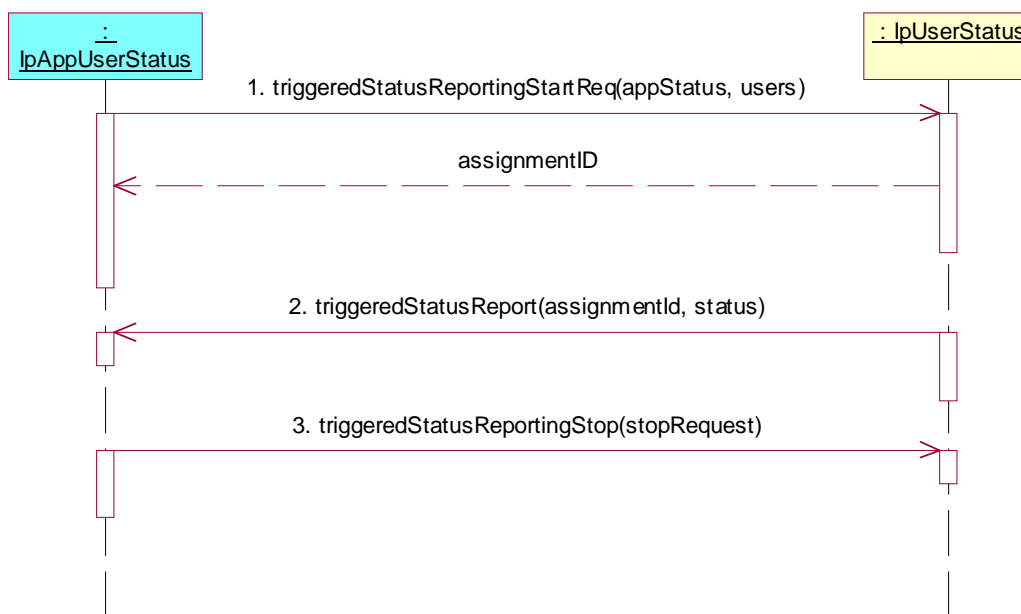
Reference: ES 201 915-6 [1], clause 8.4

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: triggeredStatusReportingStartReq() supported.

Test Sequence:

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



**Test M\_US\_04**

Summary: all methods, unknown or absent subscriber

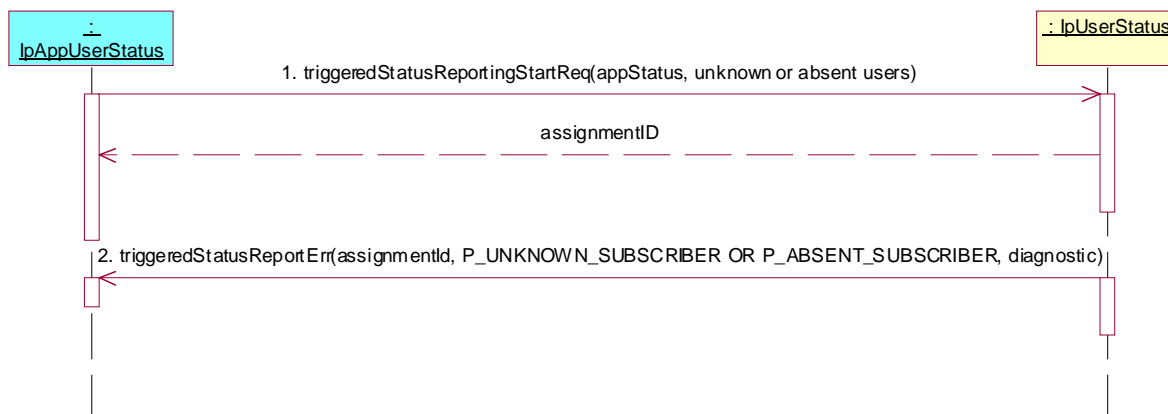
Reference: ES 201 915-6 [1], clause 8.4

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: triggeredStatusReportingStartReq() supported.

Test Sequence:

1. Method call **triggeredStatusReportingStartReq()**  
 Parameters: appStatus, users with unknown or absent subscriber  
 Check: valid value of TpSessionID is returned
2. 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



**Test M\_US\_05**

Summary: triggeredStatusReportingStop, P\_INVALID\_ASSIGNMENT\_ID

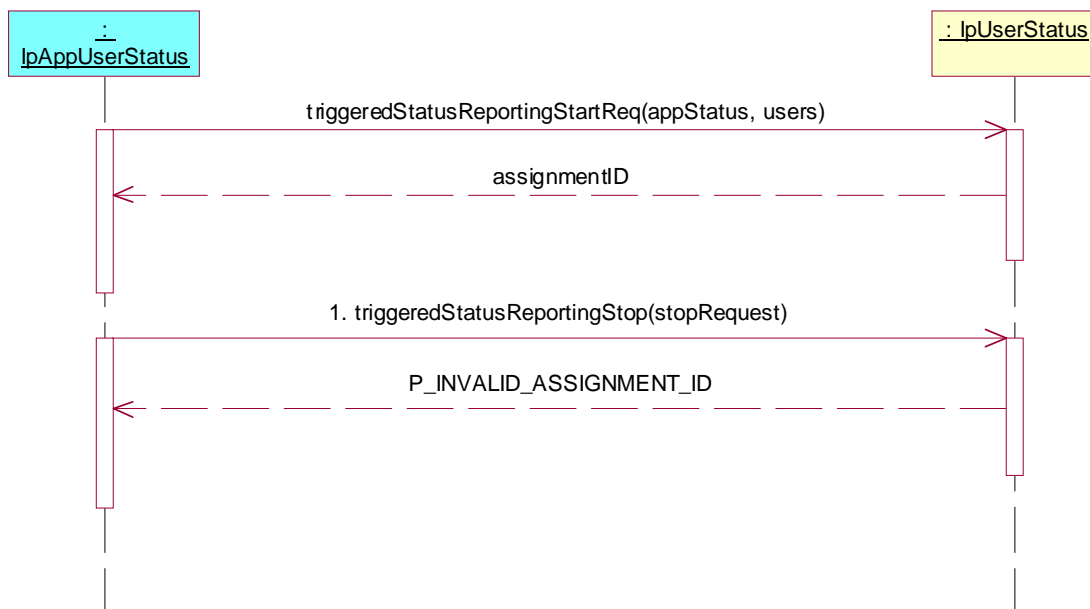
Reference: ES 201 915-6 [1], clause 8.4

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.

Selection Condition: triggeredStatusReportingStartReq() supported.

Test Sequence:

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

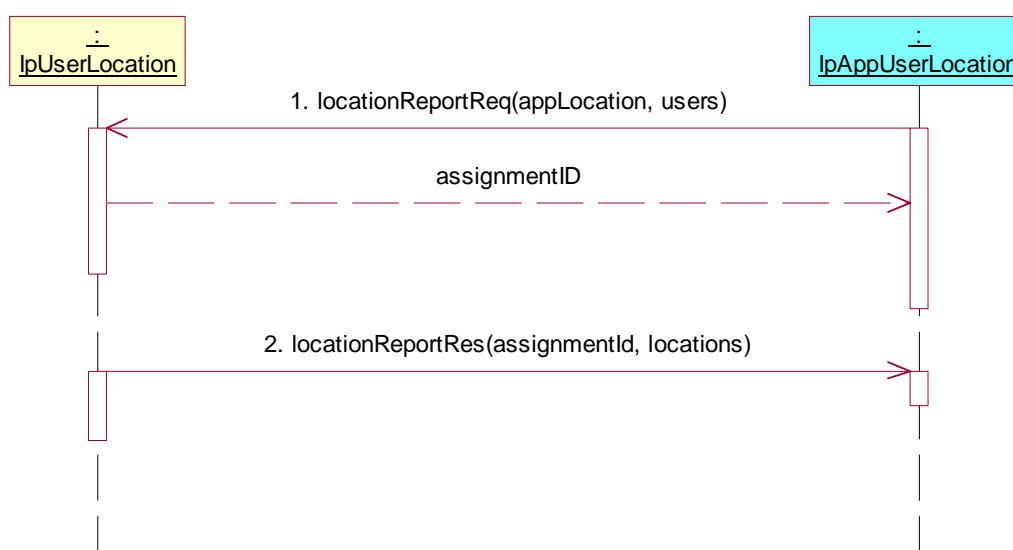
Reference: ES 201 915-6 [1], clause 8.1

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.

Test Sequence:

1. 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

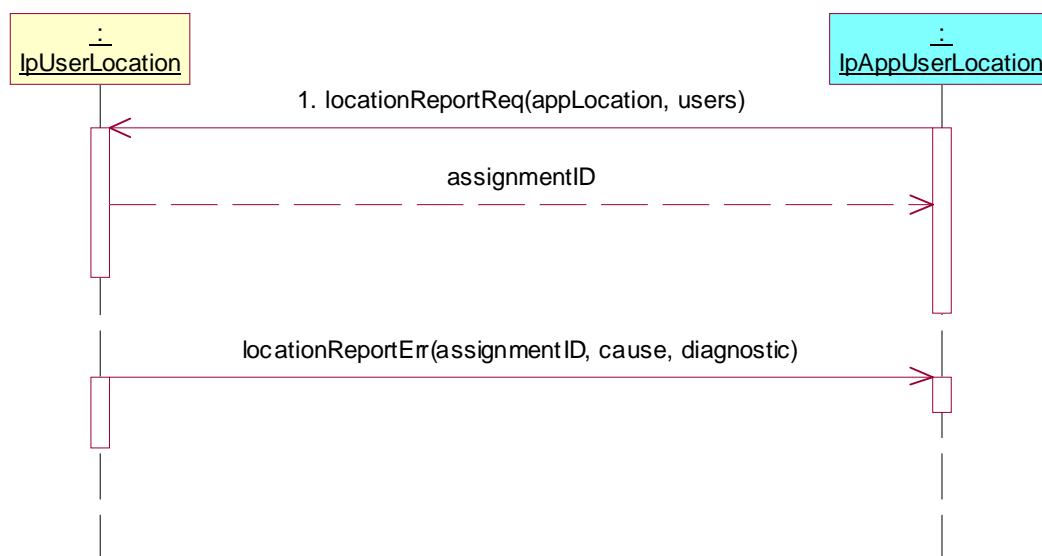
Precondition: IUT capable of invoking **locationReportReq()**

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.

Test Sequence:

1. 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



**Test M\_APP\_UL\_03**

Summary: advanced request of report of location, successful

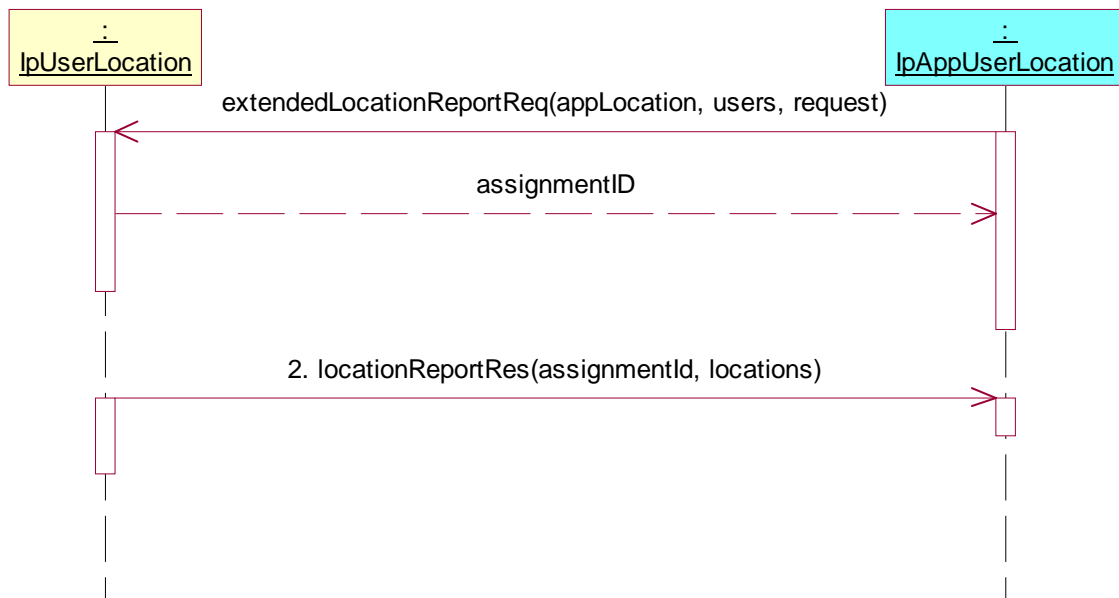
Reference: ES 201 915-6 [1], clause 8.1

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 IpUserLocation interface through selecting that service and signing the required service agreement.

Test Sequence:

1. 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



**Test M\_APP\_UL\_04**

Summary: advanced request of report of location, unsuccessful

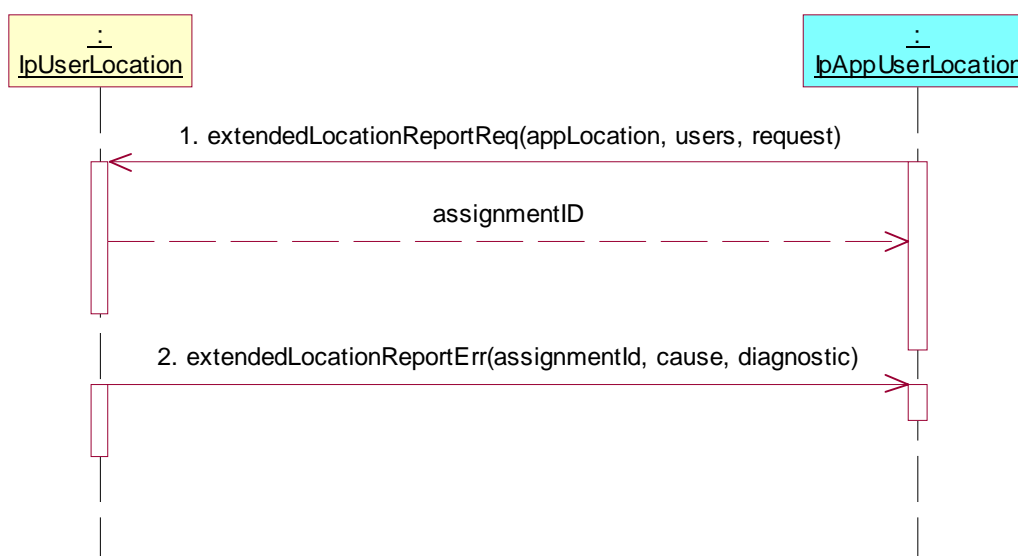
Reference: ES 201 915-6 [1], clause 8.1

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.

Test Sequence:

1. 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

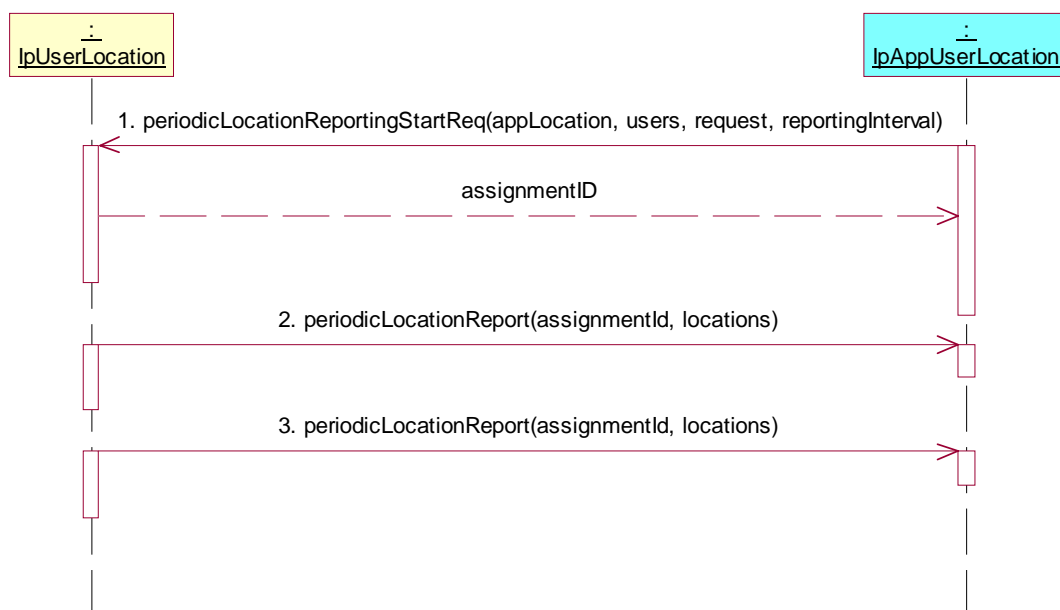
Precondition: IUT capable of invoking **periodicLocationReportingStartReq()**

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:

1. 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.



**Test M\_APP\_UL\_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 **periodicLocationReportingStop()**

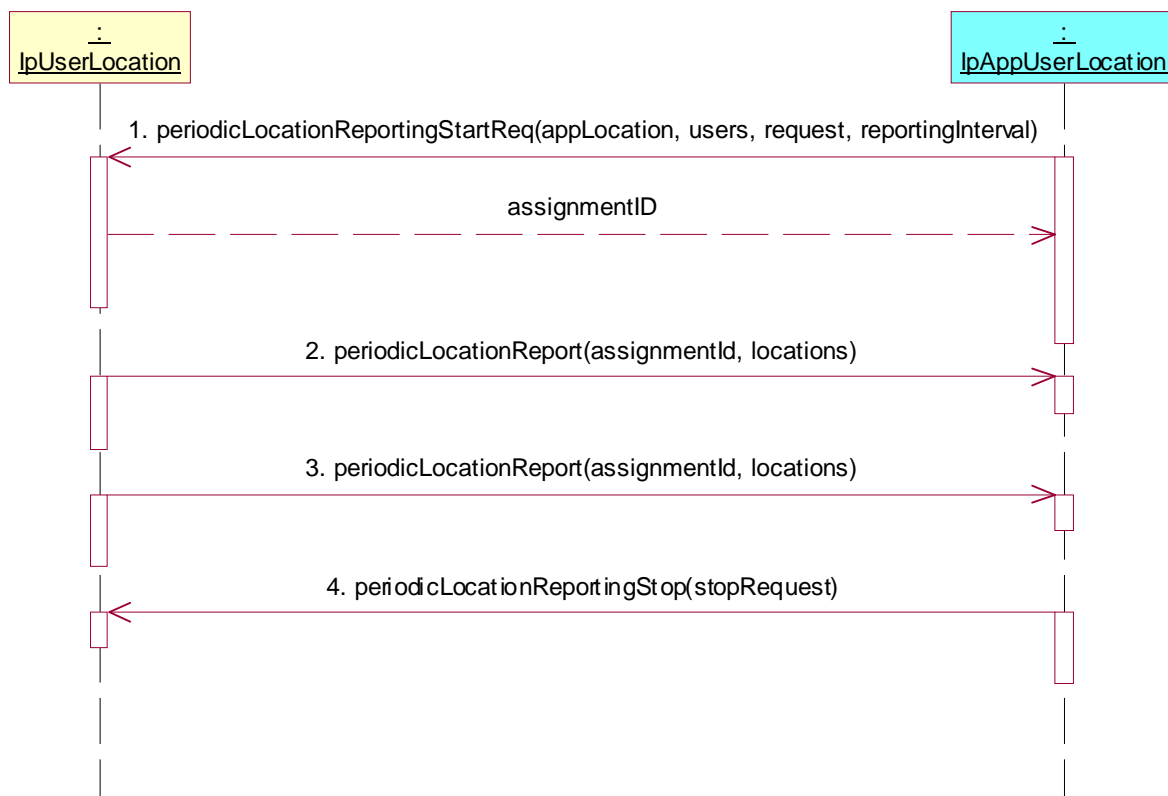
Preamble: Registration of the IUT (application) and the tester (User Location SCF) to the framework. The tester must have obtained a reference to an instance of the IpUserLocation interface through selecting that service and signing the required service agreement.

Test Sequence:

1. 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.

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



**Test M\_APP\_UL\_07**

Summary: request for periodic reports of location, unsuccessful

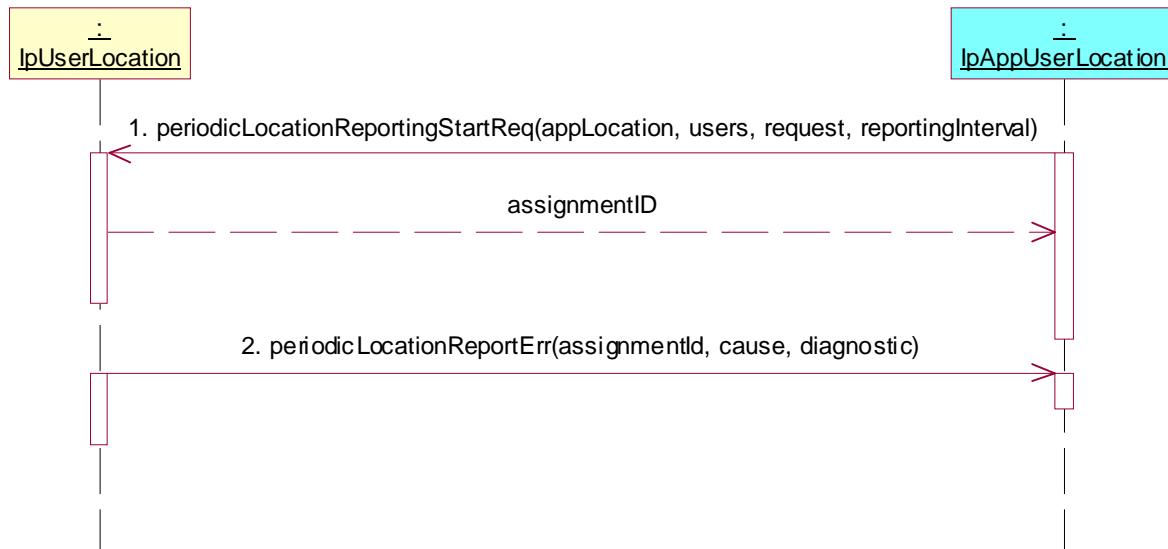
Reference: ES 201 915-6 [1], clause 8.1

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.

Test Sequence:

1. 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



### 5.3.1.2 IpAppTriggerredUserLocation interface

Precondition: **IpAppTriggerredUserLocation** interface implemented

#### Test M\_APP\_TUL\_01

Summary: request of report of location, successful

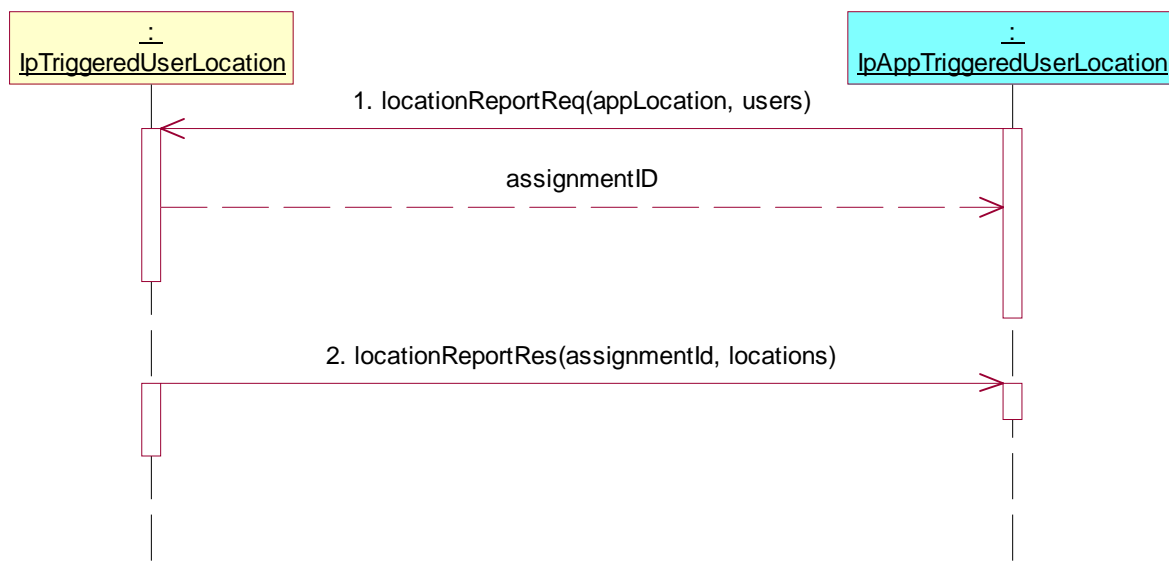
Reference: ES 201 915-6 [1], clause 8.1

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 IpTriggerredUserLocation interface through selecting that service and signing the required service agreement.

Test Sequence:

1. Triggered action: cause IUT to call **locationReportReq()** method on the tester's (SCF) **IpTriggerredUserLocation** 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

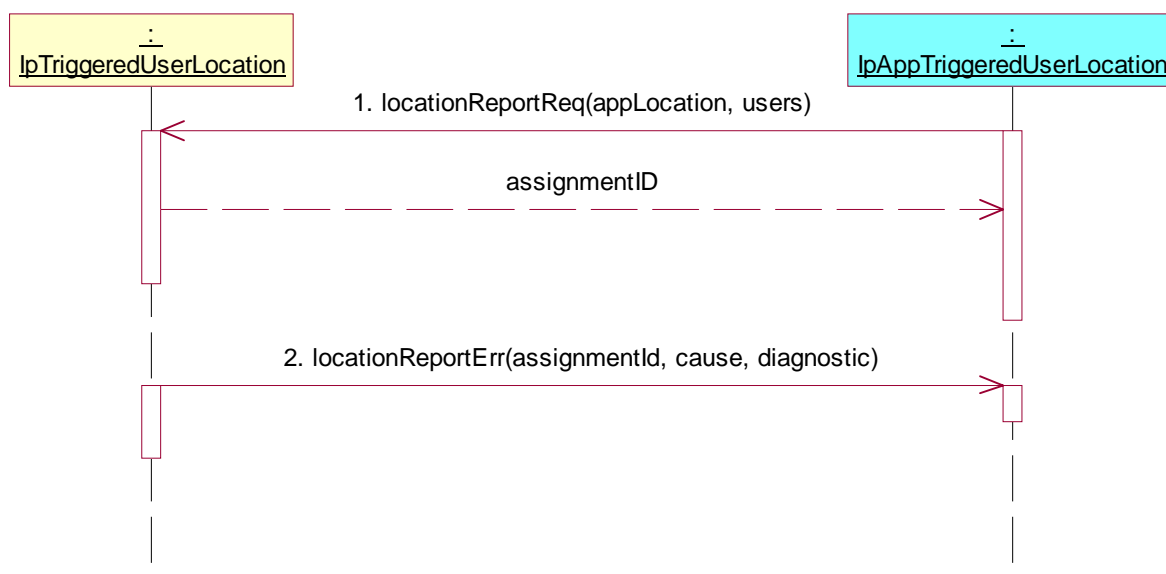
Precondition: IUT capable of invoking **locationReportReq()**

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.

Test Sequence:

1. 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

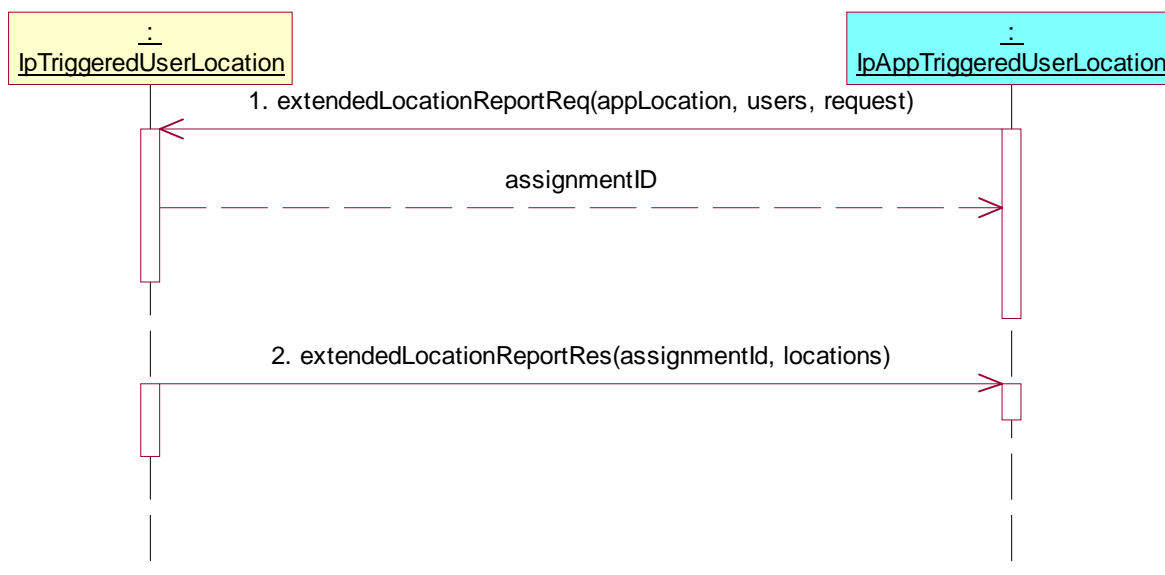
Reference: ES 201 915-6 [1], clause 8.1

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.

Test Sequence:

1. 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



**Test M\_APP\_TUL\_04**

Summary: advanced request of report of location, unsuccessful

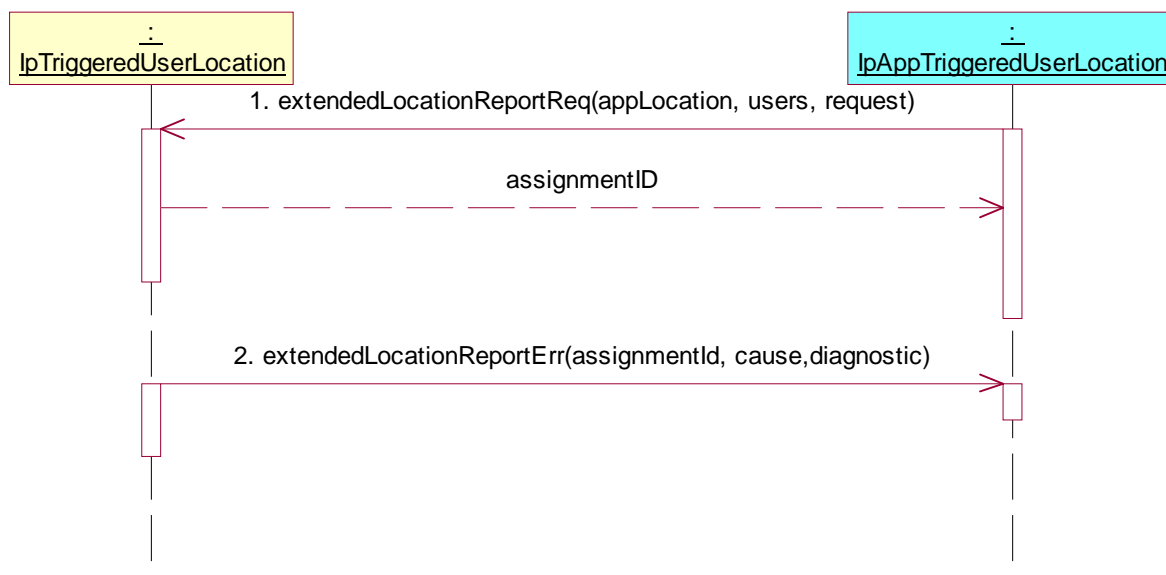
Reference: ES 201 915-6 [1], clause 8.1

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 IpTriggeredUserLocation interface through selecting that service and signing the required service agreement.

Test Sequence:

1. 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



**Test M\_APP\_TUL\_05**

Summary: request for periodic reports of location, successful

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

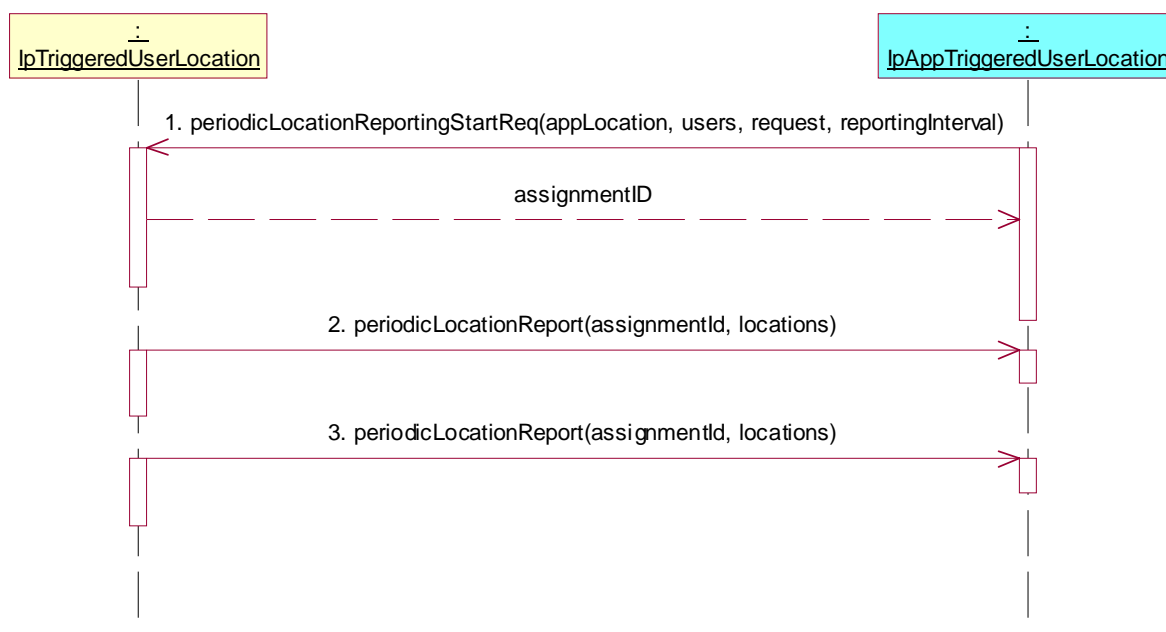
Precondition: IUT capable of invoking **periodicLocationReportingStartReq()**

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.

Test Sequence:

1. 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()**

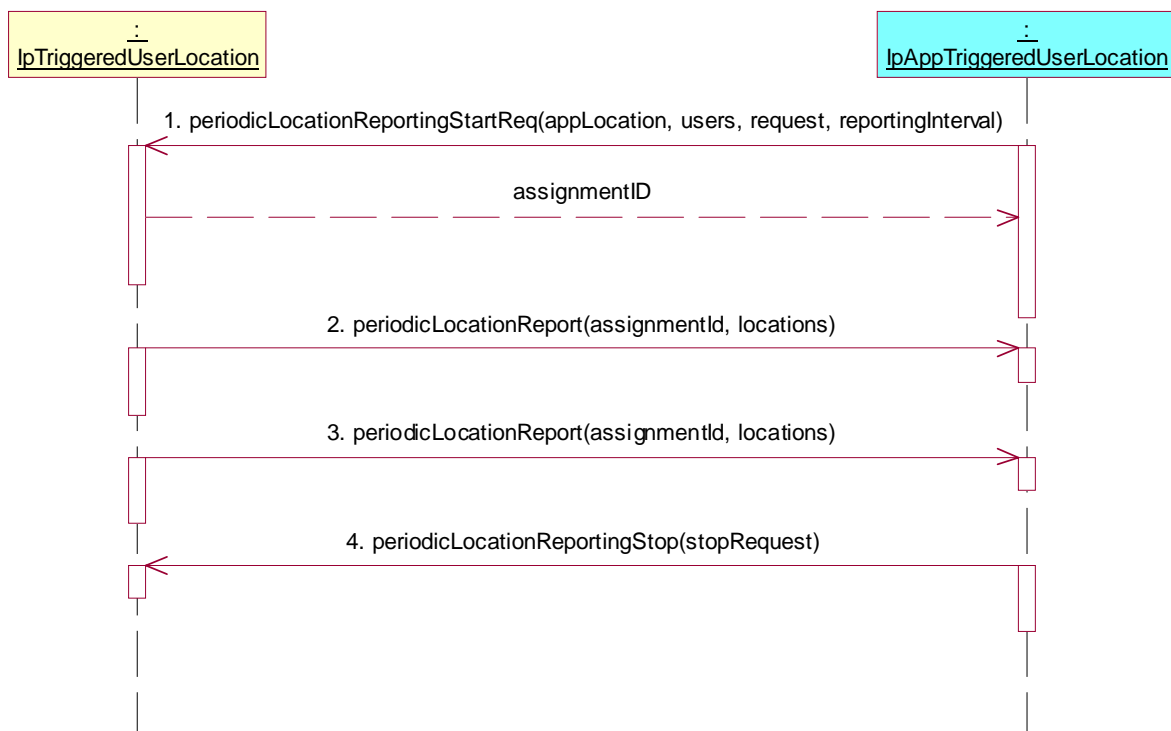
Preamble: Registration of the IUT (application) and the tester (User Location SCF) to the framework. The tester must have obtained a reference to an instance of the IpTriggeredUserLocation interface through selecting that service and signing the required service agreement.

Test Sequence:

1. 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.

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



**Test M\_APP\_TUL\_07**

Summary: request for periodic reports of location, unsuccessful

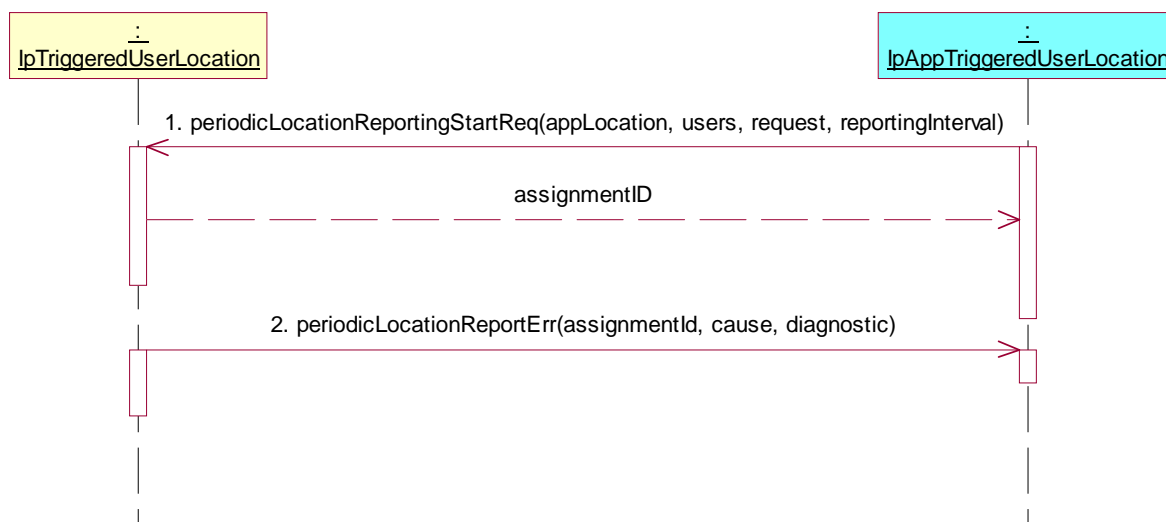
Reference: ES 201 915-6 [1], clause 8.1

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 IpTriggeredUserLocation interface through selecting that service and signing the required service agreement.

Test Sequence:

1. 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



**Test M\_APP\_TUL\_08**

Summary: request for triggered reports of location, successful

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

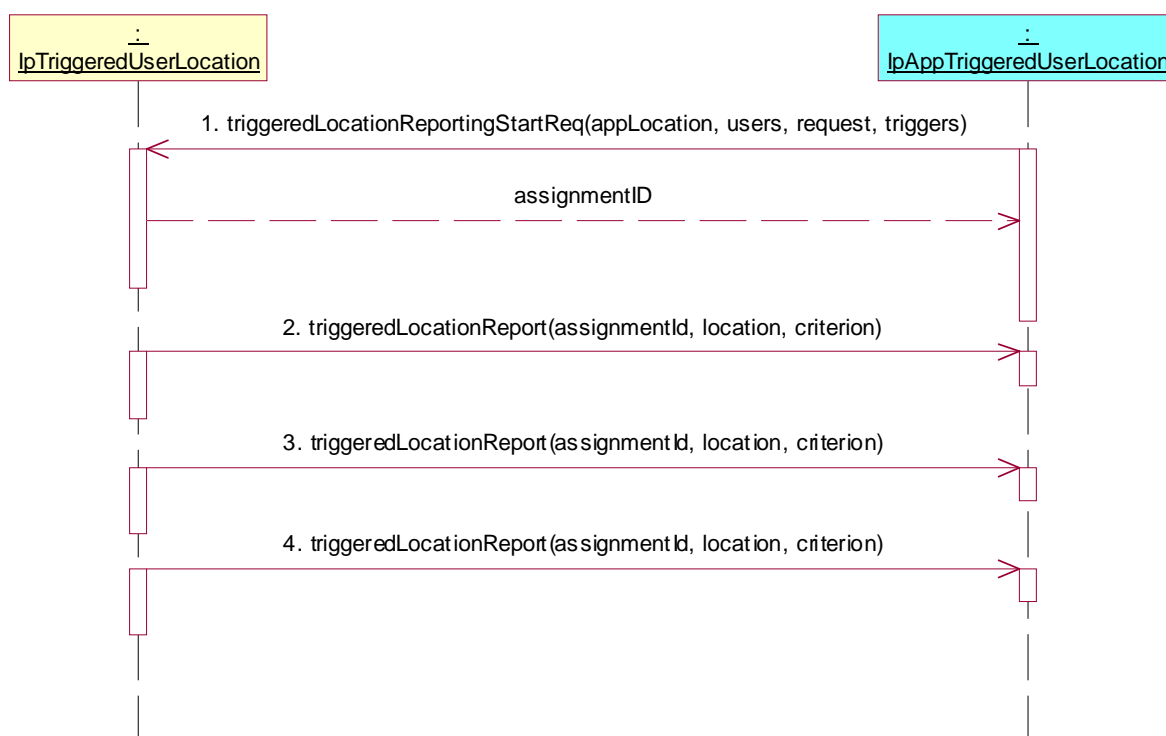
Precondition: IUT capable of invoking **triggeredLocationReportingStartReq()**

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.

Test Sequence:

1. 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.



**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()**

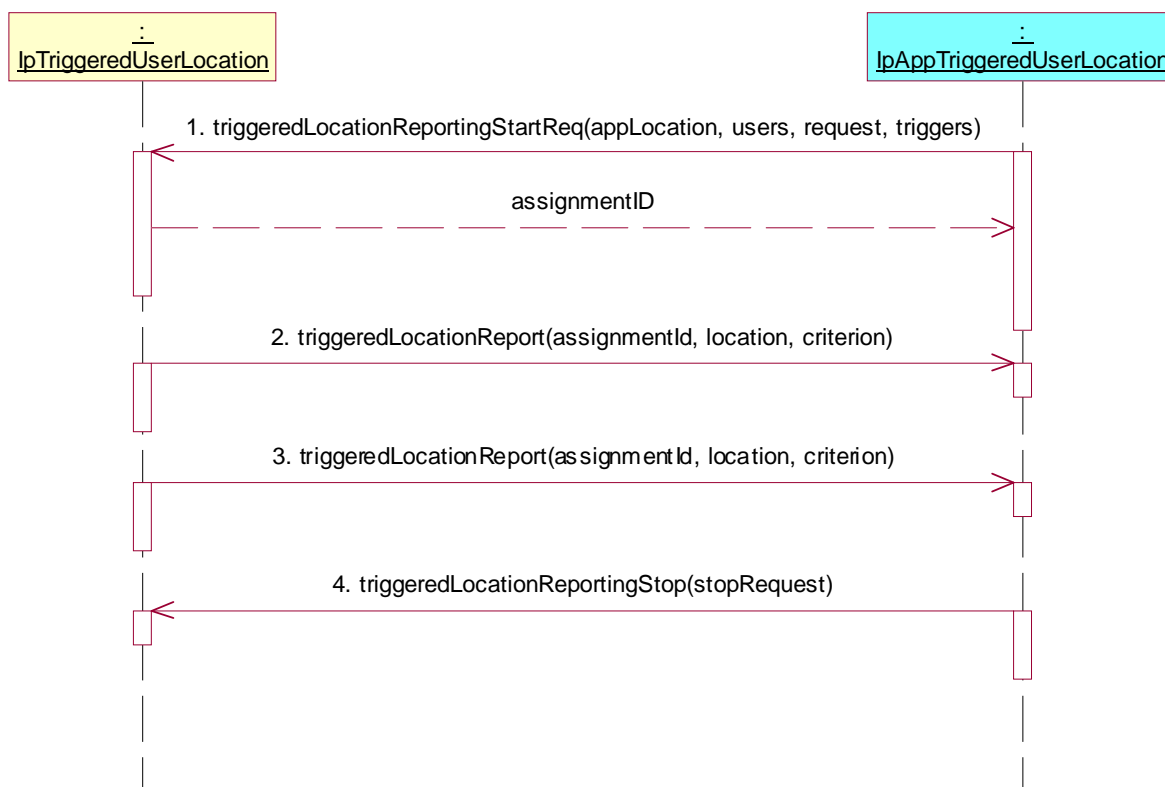
Preamble: Registration of the IUT (application) and the tester (User Location SCF) to the framework. The tester must have obtained a reference to an instance of the IpTriggeredUserLocation interface through selecting that service and signing the required service agreement.

Test Sequence:

1. 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



**Test M\_APP\_TUL\_10**

Summary: request for triggered reports of location, unsuccessful

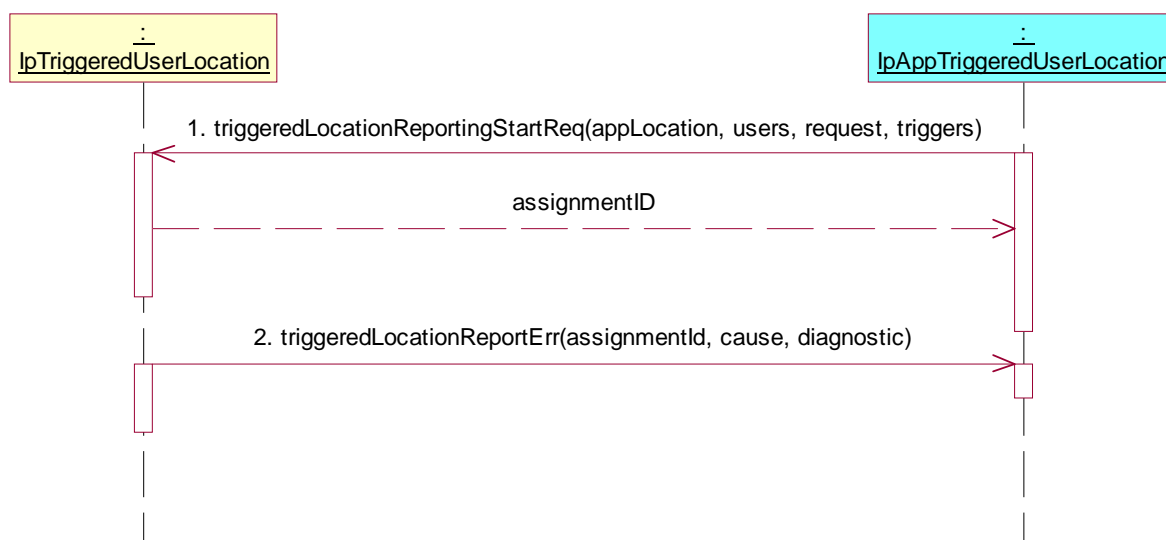
Reference: ES 201 915-6 [1], clause 8.1

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.

Test Sequence:

1. 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





## 5.3.2 User Location Camel

Precondition: **IpAppUserLocationCamel** interface implemented

### Test M\_APP\_ULC\_01

Summary: request of report of location, successful

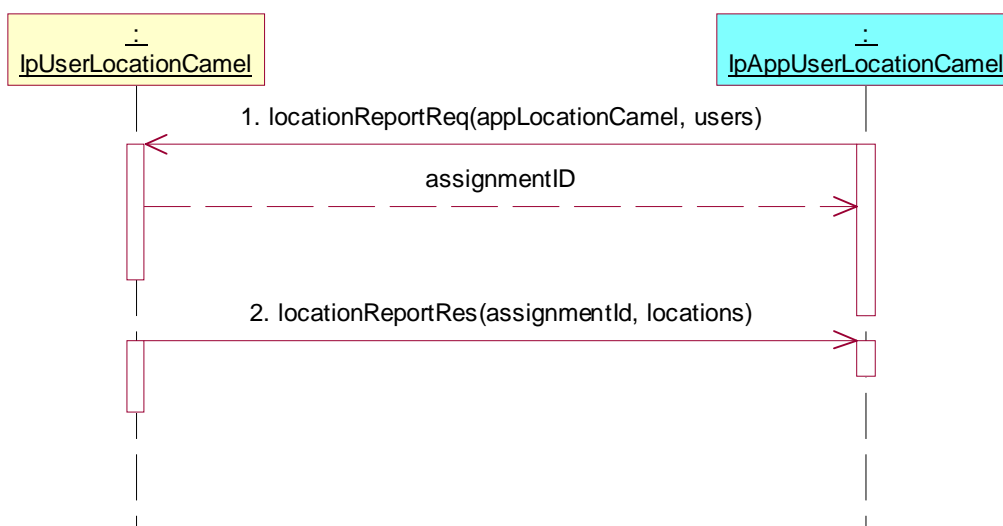
Reference: ES 201 915-6 [1], clause 8.1

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.

Test Sequence:

1. 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



**Test M\_APP\_ULC\_02**

Summary: request of report of location, unsuccessful

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

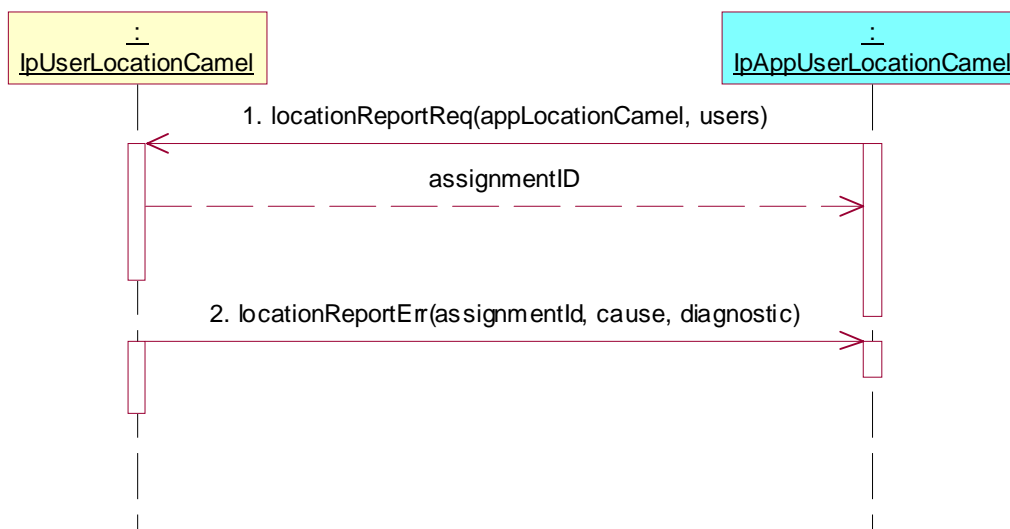
Precondition: IUT capable of invoking **locationReportReq()**

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 IpUserLocationCamel interface through selecting that service and signing the required service agreement.

Test Sequence:

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



**Test M\_APP\_ULC\_03**

Summary: request for periodic reports of location, successful

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

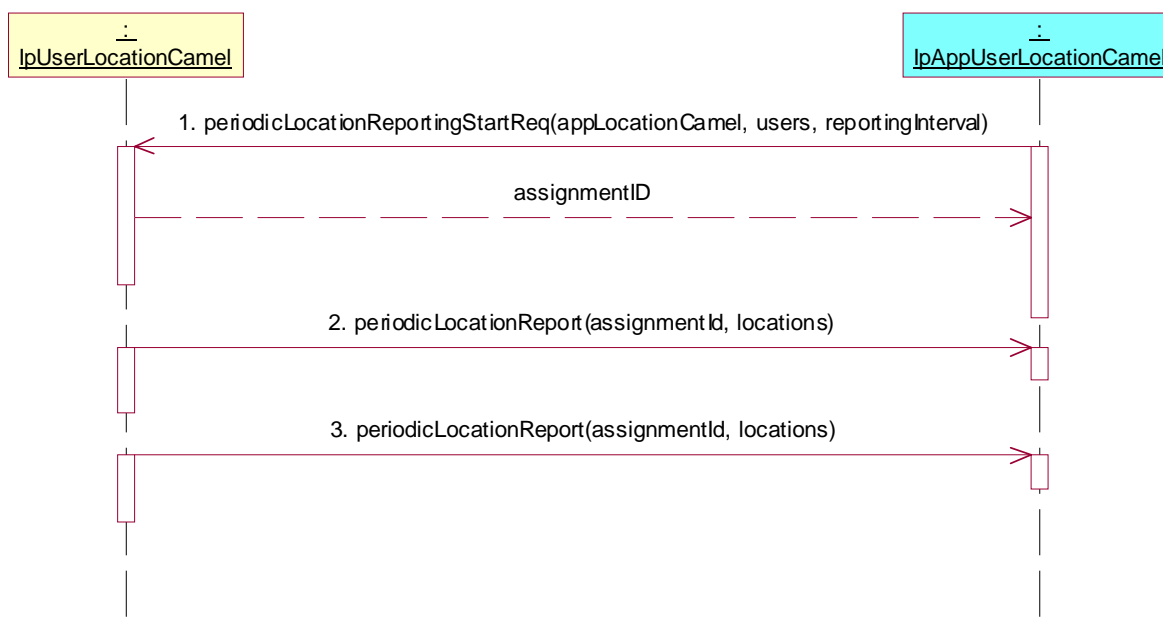
Precondition: IUT capable of invoking **periodicLocationReportingStartReq()**

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:

1. 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.



**Test M\_APP\_ULC\_04**

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()**

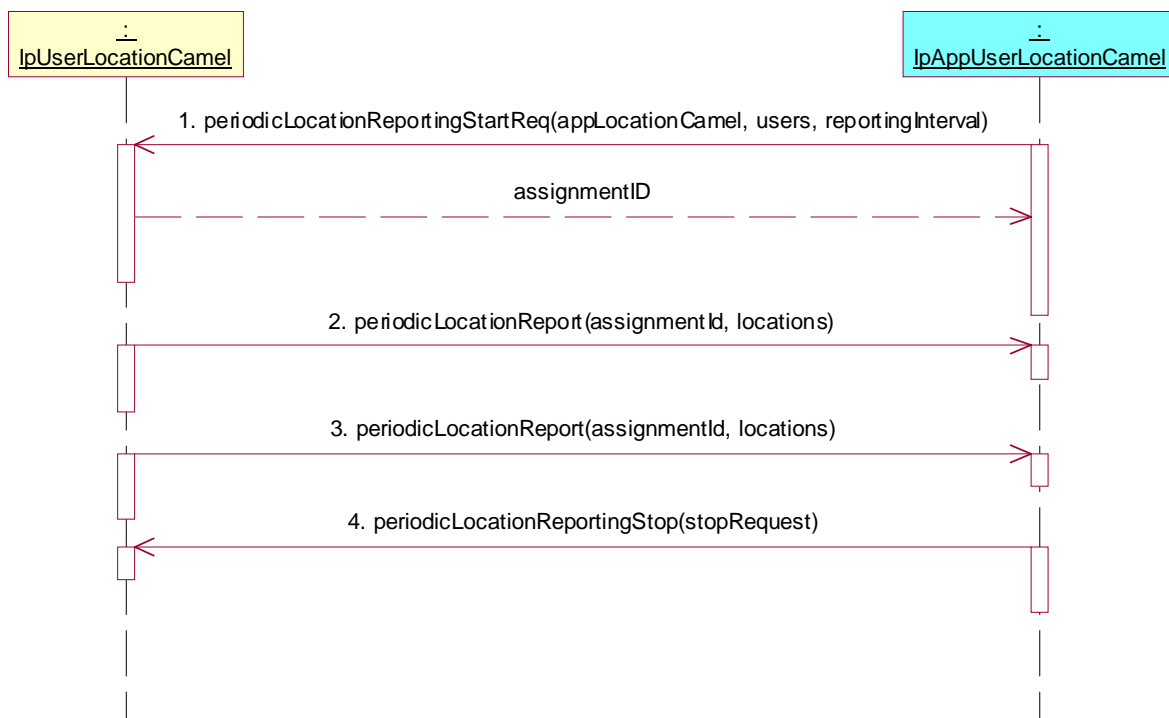
Preamble: Registration of the IUT (application) and the tester (User Location SCF) to the framework. The tester must have obtained a reference to an instance of the IpUserLocationCamel interface through selecting that service and signing the required service agreement.

Test Sequence:

1. 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.

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



**Test M\_APP\_ULC\_05**

Summary: request for periodic reports of location, unsuccessful

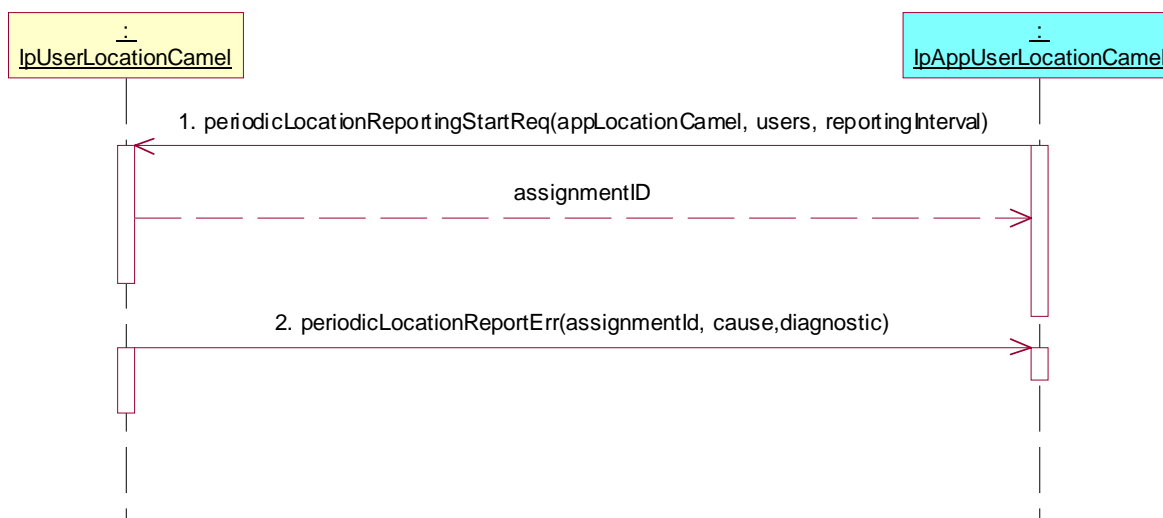
Reference: ES 201 915-6 [1], clause 8.1

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.

Test Sequence:

1. 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



**Test M\_APP\_ULC\_06**

Summary: request for triggered reports of location, successful

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

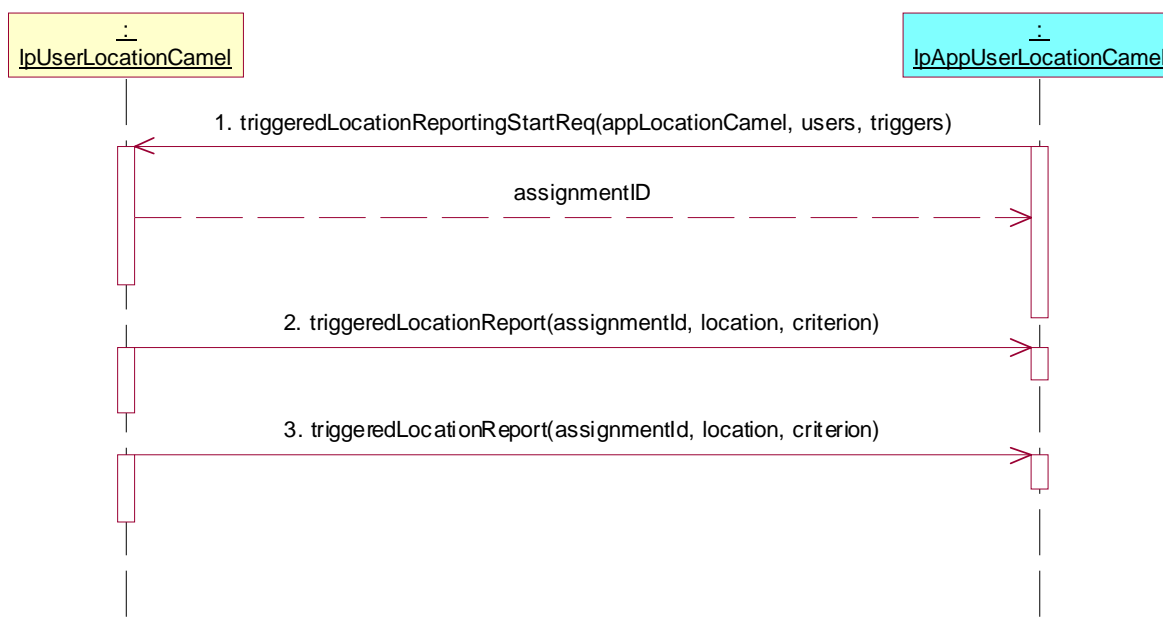
Precondition: IUT capable of invoking **triggeredLocationReportingStartReq()**

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.

Test Sequence:

1. 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.



**Test M\_APP\_ULC\_07**

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()**

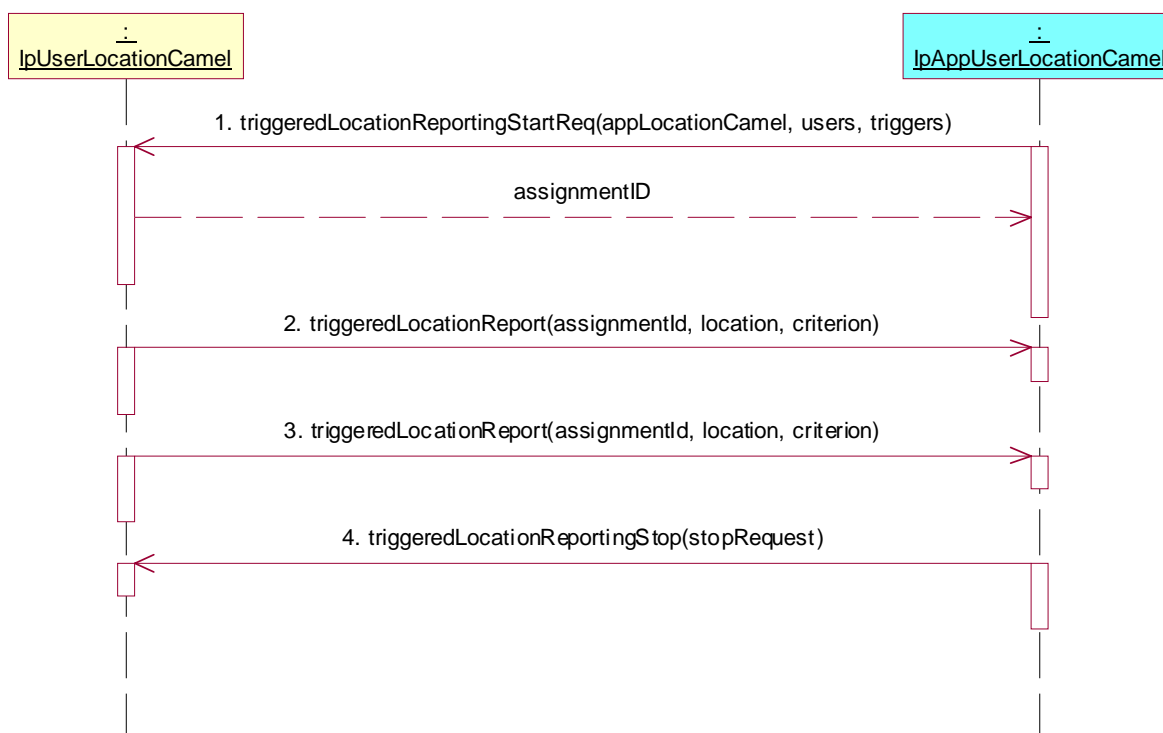
Preamble: Registration of the IUT (application) and the tester (User Location SCF) to the framework. The tester must have obtained a reference to an instance of the IpUserLocationCamel interface through selecting that service and signing the required service agreement.

Test Sequence:

1. 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.

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



**Test M\_APP\_ULC\_08**

Summary: request for triggered reports of location, unsuccessful

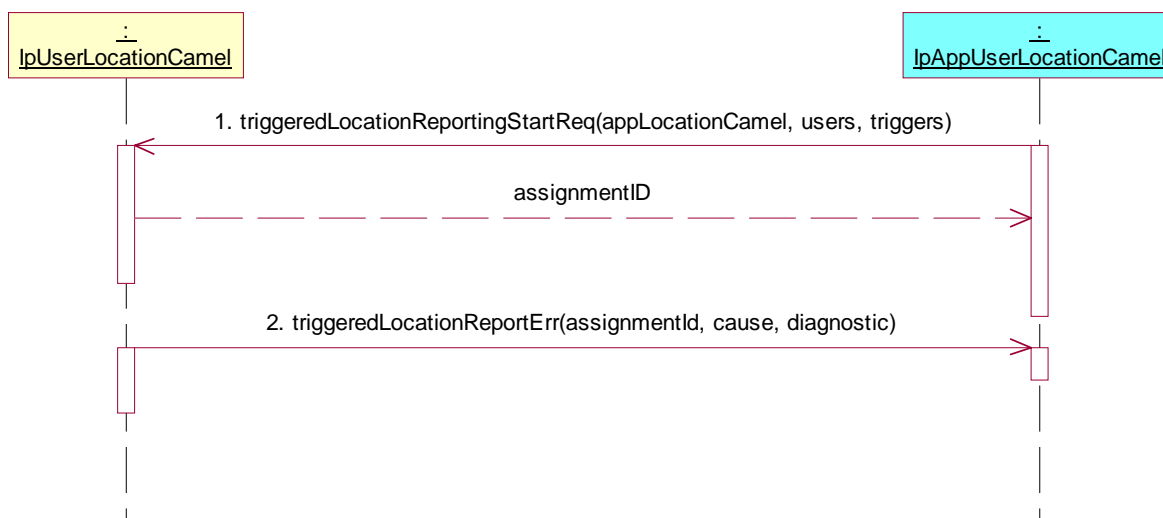
Reference: ES 201 915-6 [1], clause 8.1

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.

Test Sequence:

1. 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





### 5.3.3 User Location Emergency

Precondition: **IpAppUserLocationEmergency** interface implemented

#### Test M\_APP\_ULE\_01

Summary: request of report of location, successful

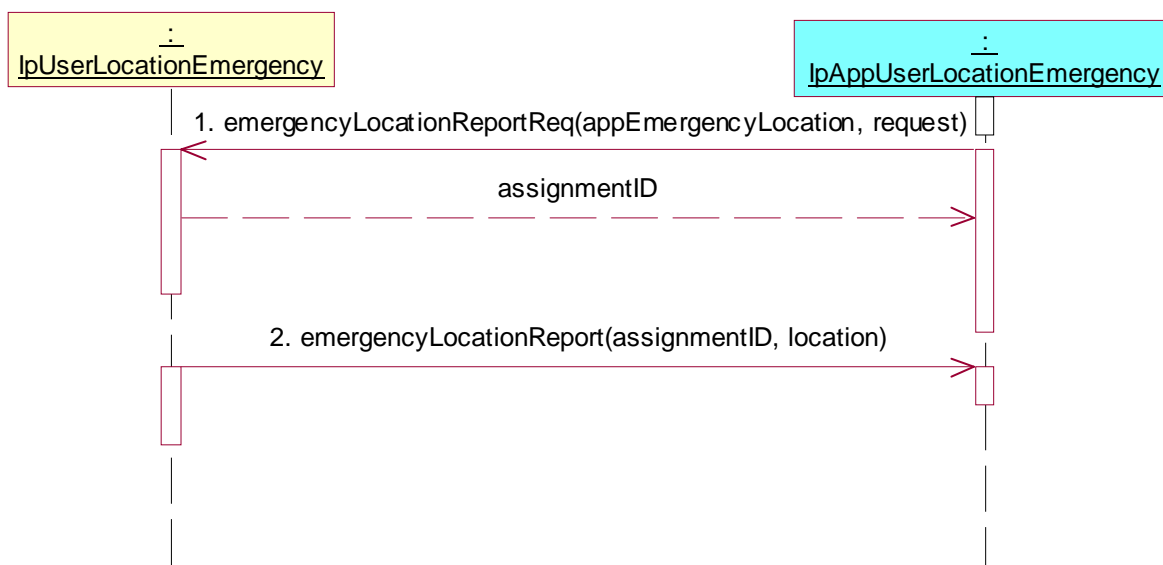
Reference: ES 201 915-6 [1], clause 8.1

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.

Test Sequence:

1. 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



**Test M\_APP\_ULE\_02**

Summary: request of report of location, unsuccessful

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

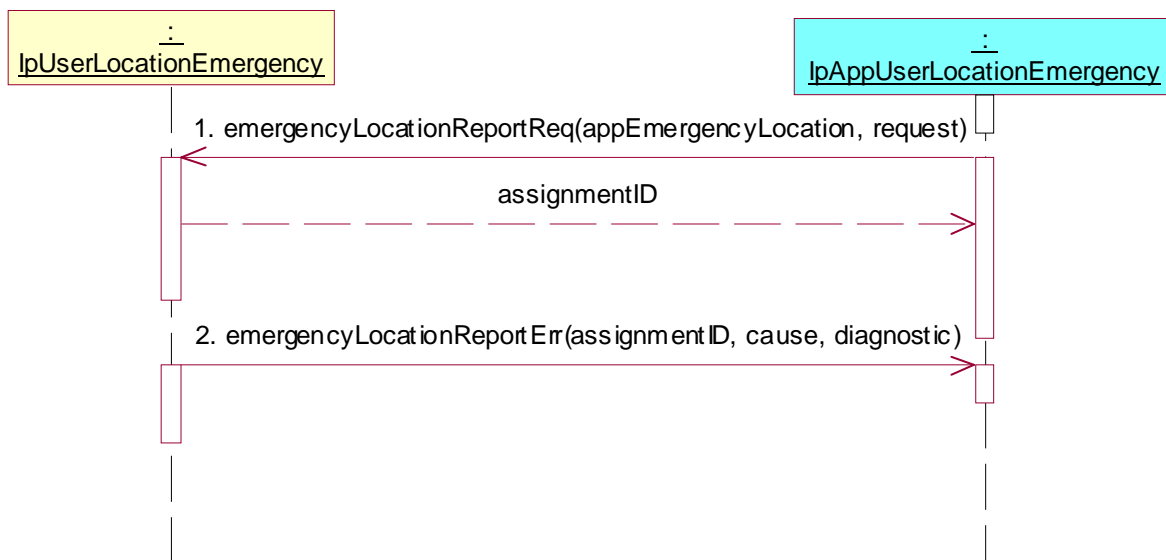
Precondition: IUT capable of invoking **emergencyLocationReportReq()**

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.

Test Sequence:

1. 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



**Test M\_APP\_ULE\_03**

Summary: subscribe to network initiated reports of location

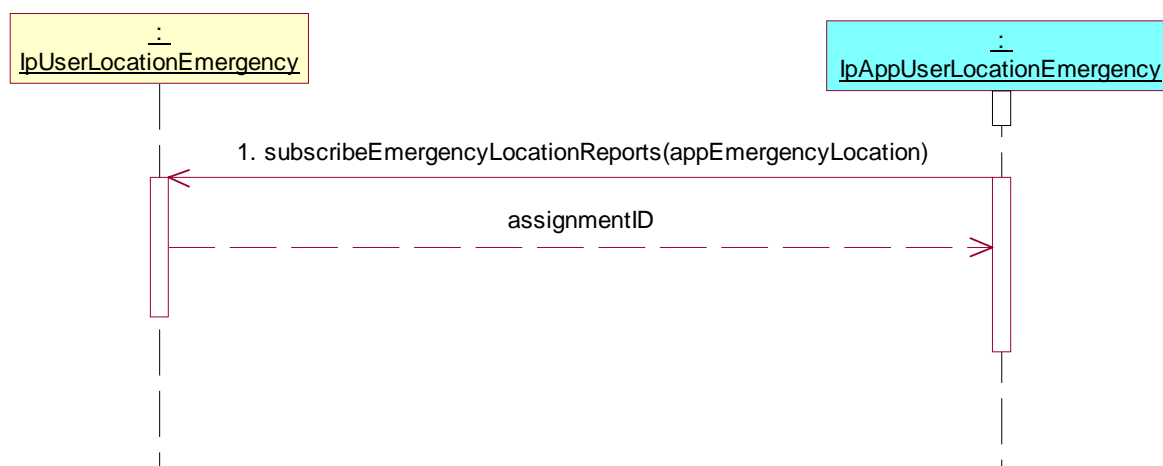
Reference: ES 201 915-6 [1], clause 8.1

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.

Test Sequence:

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



**Test M\_APP\_ULE\_04**

Summary: subscribe and unsubscribe to network initiated reports of location

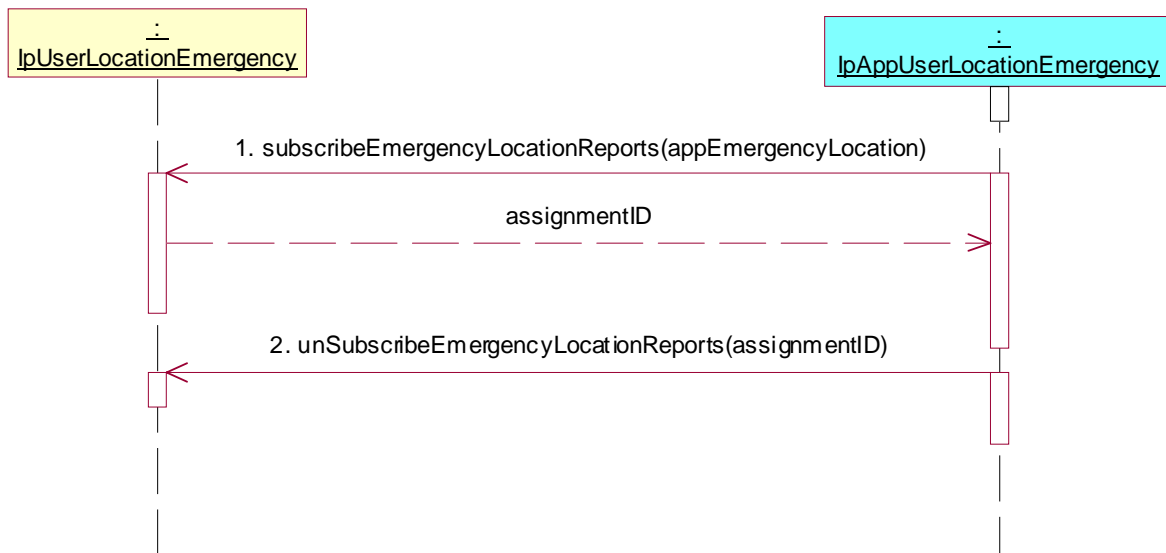
Reference: ES 201 915-6 [1], clause 8.1

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.

Test Sequence:

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



### 5.3.4 User Status

Precondition: **IpAppUserStatus** interface implemented

#### Test M\_APP\_US\_01

Summary: request of status report, successful

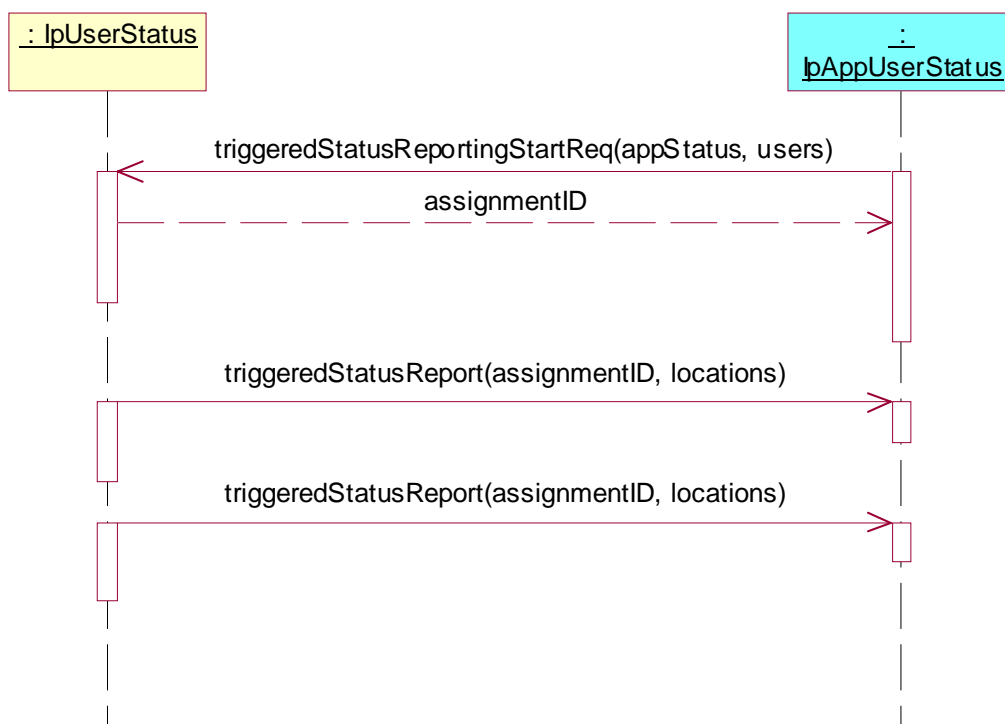
Reference: ES 201 915-6 [1], clause 8.1

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.

Test Sequence:

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

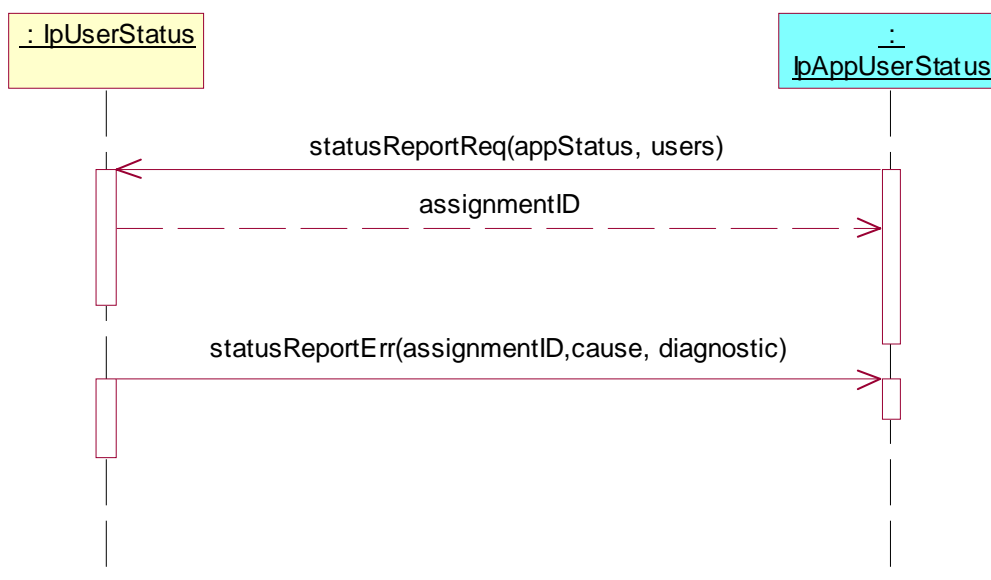
Reference: ES 201 915-6 [1], clause 8.1

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.

Test Sequence:

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



**Test M\_APP\_US\_03**

Summary: request for triggered status reports, successful

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

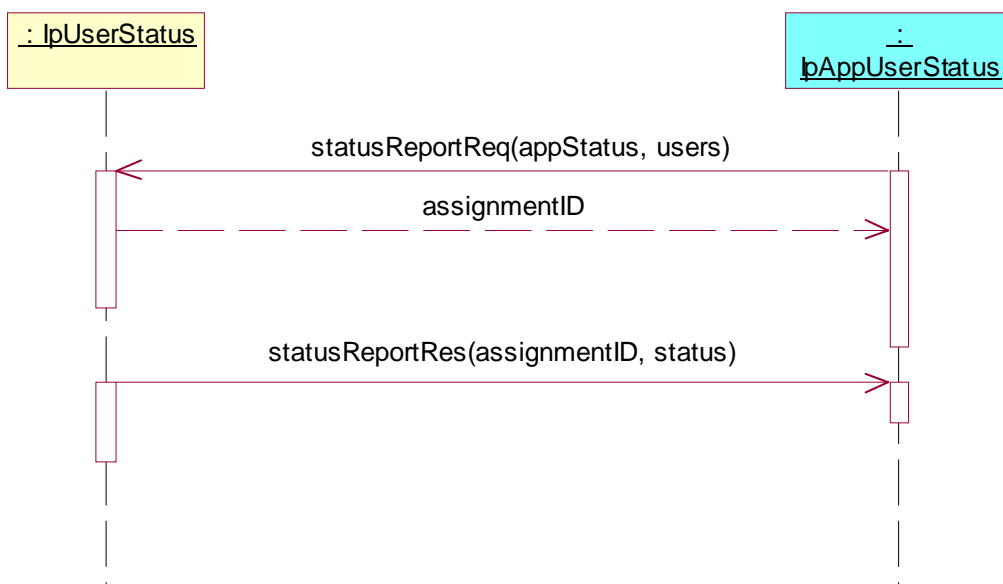
Precondition: IUT capable of invoking **triggeredStatusReportingStartReq()**

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:

1. 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()**

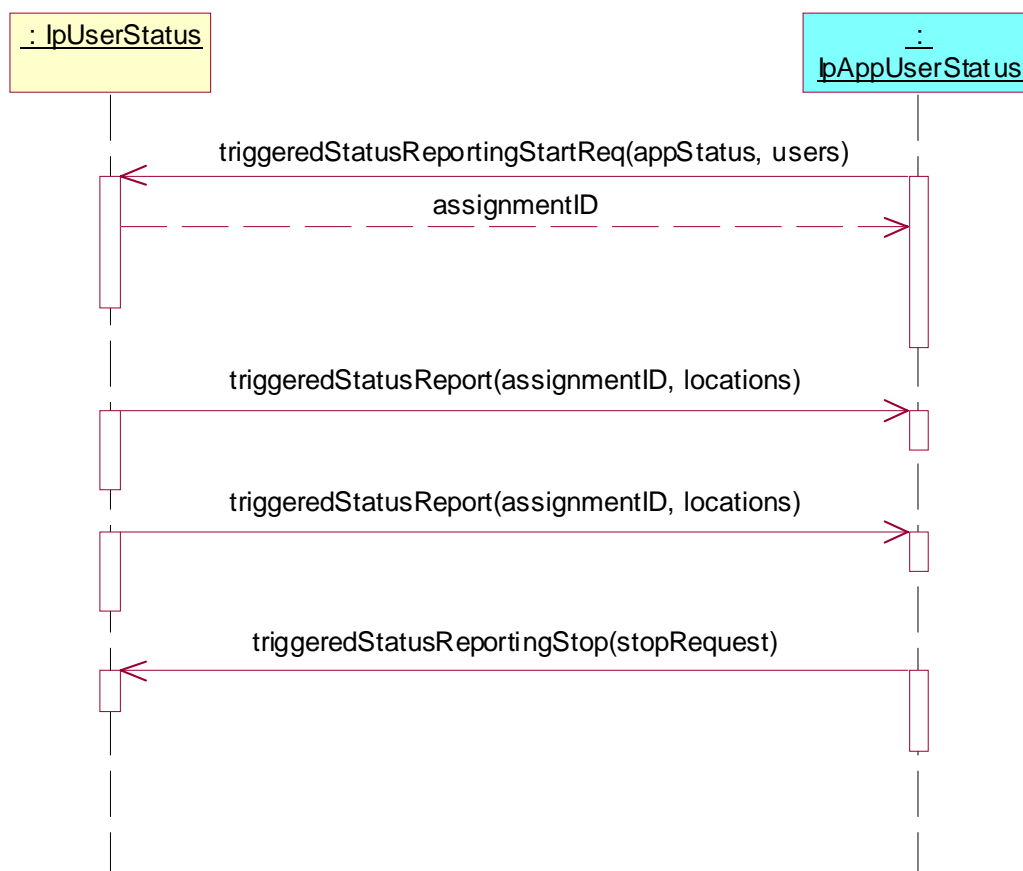
Preamble: Registration of the IUT (application) and the tester (User Location SCF) to the framework. The tester must have obtained a reference to an instance of the IpUserStatus interface through selecting that service and signing the required service agreement.

Test Sequence:

1. 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

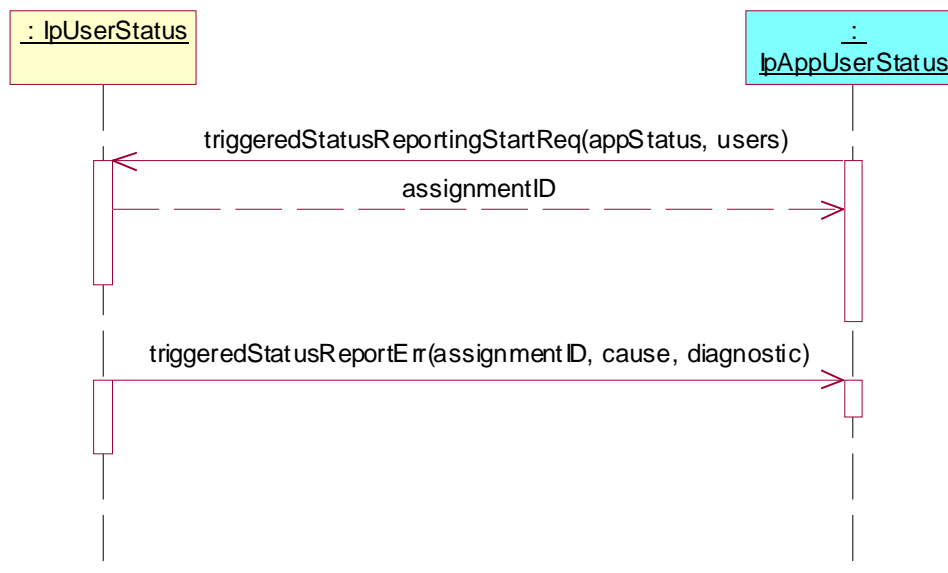
Reference: ES 201 915-6 [1], clause 8.1

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.

Test Sequence:

1. 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



---

## History

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