

ETSI TS 132 506 V10.0.0 (2011-05)

Technical Specification

**Digital cellular telecommunications system (Phase 2+);
Universal Mobile Telecommunications System (UMTS);
LTE;
Telecommunication management;
Self-Configuration of Network Elements
Integration Reference Point (IRP);
Solution Set (SS) definitions
(3GPP TS 32.506 version 10.0.0 Release 10)**



Reference

DTS/TSGS-0532506va00

Keywords

GSM, LTE, UMTS

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

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 2011.
All rights reserved.

DECT™, **PLUGTESTS™**, **UMTS™**, **TIPHON™**, the TIPHON logo and the ETSI logo are Trade Marks of ETSI registered for the benefit of its Members.

3GPP™ is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners.

LTE™ is a Trade Mark of ETSI currently being registered

for the benefit of its Members and of the 3GPP Organizational Partners.

GSM® and the GSM logo are Trade Marks registered and owned by the GSM Association.

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 Technical Specification (TS) has been produced by ETSI 3rd Generation Partnership Project (3GPP).

The present document may refer to technical specifications or reports using their 3GPP identities, UMTS identities or GSM identities. These should be interpreted as being references to the corresponding ETSI deliverables.

The cross reference between GSM, UMTS, 3GPP and ETSI identities can be found under <http://webapp.etsi.org/key/queryform.asp>.

Contents

Intellectual Property Rights	2
Foreword.....	2
Foreword.....	5
Introduction	5
1 Scope	6
2 References	6
3 Definitions and abbreviations.....	7
3.1 Definitions	7
3.2 Abbreviations	8
4 Solution Set definitions	9
Annex A (normative): CORBA Solution Set	10
A.1 Architectural features	10
A.1.1 Syntax for Distinguished Names	10
A.1.2 Notification Services	10
A.1.3 Push and Pull Style.....	10
A.1.4 Support multiple notifications in one push operation.....	10
A.2 Mapping	11
A.2.1 Operation and Notification mapping	11
A.2.2 Operation parameter mapping	11
A.2.3 Notification parameter mapping.....	14
A.3 Solution Set definitions	15
A.3.1 IDL definition structure.....	15
A.3.2 IDL specification (file name "SelfConfIRPConstDefs.idl").....	16
A.3.3 IDL specification (file name "SelfConfIRPSystem.idl").....	17
A.3.4 IDL specification (file name "SelfConfIRPNotifications.idl").....	20
Annex B (normative): XML definitions	22
B.1 Architectural Features	22
B.1.1 Syntax for Distinguished Names	22
B.1.2 Notification Services	22
B.1.3 IOC Definitions	22
B.2 Mapping	22
B.3 Solution Set definitions	22
B.3.1 XML definition structure.....	22
B.3.2 Graphical Representation	23
B.3.3 XML Schema 'scIRPNotif.xsd'.....	27
B.3.4 XML Schema 'scIRPIOCs.xsd'.....	30
Annex C (normative): SOAP Solution Set	32
C.1 Architectural features	32
C.1.1 Syntax for Distinguished Names	32
C.1.2 Notification Services	32
C.1.3 Supported W3C specifications	32
C.1.4 Prefixes and namespaces	32
C.2 Mapping	33
C.2.1 Operation and Notification mapping	33
C.2.2 Operation parameter mapping	33
C.2.2.1 Operation listScManagementCapabilities	34

C.2.2.1.1	Input parameters	34
C.2.2.1.2	Output parameters	34
C.2.2.1.3	Fault definition	34
C.2.2.2	Operation listScManagementProfiles	34
C.2.2.2.1	Input parameters	34
C.2.2.2.2	Output parameters	34
C.2.2.3	Operation createScManagementProfile	35
C.2.2.3.1	Input parameters	35
C.2.2.3.2	Output parameters	35
C.2.2.4	Operation deleteScManagementProfile	35
C.2.2.4.1	Input parameters	35
C.2.2.4.2	Output parameters	36
C.2.2.5	Operation listScProcesses	36
C.2.2.5.1	Input parameters	36
C.2.2.5.2	Output parameters	36
C.2.2.6	Operation resumeScProcess	36
C.2.2.6.1	Input parameters	36
C.2.2.6.2	Output parameters	36
C.2.2.7	Operation terminateSctProcess	37
C.2.2.7.1	Input parameters	37
C.2.2.7.2	Output parameters	37
C.2.2.8	Operation changeScManagementProfile	37
C.2.2.8.1	Input parameters	37
C.2.2.8.2	Output parameters	37
C.3	Solution Set definitions	38
C.3.1	WSDL definition structure	38
C.3.2	Graphical Representation	38
C.3.3	WSDL specification 'ScIRPSystem.wsdl'	39
Annex D (informative):	Change history	47
History		48

Foreword

This Technical Specification (TS) has been produced by the 3rd Generation Partnership Project (3GPP).

The contents of the present document are subject to continuing work within the TSG and may change following formal TSG approval. Should the TSG modify the contents of the present document, it will be re-released by the TSG with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

- x the first digit:
 - 1 presented to TSG for information;
 - 2 presented to TSG for approval;
 - 3 or greater indicates TSG approved document under change control.
- y the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.
- z the third digit is incremented when editorial only changes have been incorporated in the document.

Introduction

The present document is part of a TS-family covering the 3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; Telecommunication management, as identified below:

- 32.501: Self-Configuration of Network Elements; Concepts and Integration Reference Point (IRP) Requirements
- 32.502: Self-Configuration of Network Elements Integration Reference Point (IRP); Information Service (IS)
- 32.506: Self-Configuration of Network Elements Integration Reference Point (IRP): Solution Set (SS) definitions "**

The present document is part of a TS-family which describe the requirements and information model necessary for the Telecommunication Management (TM) of 3G systems. The TM principles and TM architecture are specified in 3GPP TS 32.101 [1] and 3GPP TS 32.102 [2].

For the purpose of Self-Configuration of Network Elements IRP, see TS 32.501 [3]

1 Scope

The present document specifies the Solution Set definitions for the IRP whose semantics are specified in Self-Configuration of Network Elements Integration Reference Point (IRP): Information Service (3GPP TS 32.502 [4]).

This Solution Set specification is related to 3GPP TS 32.502 V10.0.X.

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, edition number, version number, etc.) non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document*.

- [1] 3GPP TS 32.101: "Telecommunication management; Principles and high level requirements".
- [2] 3GPP TS 32.102: "Telecommunication management; Architecture".
- [3] 3GPP TS 32.501: " Technical Specification Group Services and System Aspects; Telecommunication management; Self-Configuration of Network Elements; Concepts and Integration Reference Point (IRP) Requirements ".
- [4] 3GPP TS 32.502: " Technical Specification Group Services and System Aspects; Telecommunication management; Self-Configuration of Network Elements Integration Reference Point (IRP); Information Service (IS)".
- [5] 3GPP TR 32.816: "Telecommunication management; Study on Management of Evolved Universal Terrestrial Radio Access Network (E-UTRAN) and Evolved Packet Core (EPC)".
- [6] OMG TC Document telecom/98-11-01: "OMG Notification Service".
<http://www.omg.org/technology/documents/>
- [7] 3GPP TS 32.531: "Telecommunication management; Software management; Concepts and Integration Reference Point (IRP) Requirements".
- [8] 3GPP TS 32.532: "Telecommunication management; Software management Integration Reference Point (IRP); Information Service (IS)".
- [9] 3GPP TS 32.536: "Telecommunication management; Software management Integration Reference Point (IRP); Solution Set definitions ".
- [10] W3C REC-xml-20001006: "Extensible Markup Language (XML) 1.0 (Second Edition)".
- [11] W3C REC-xmlschema-0-20010502: "XML Schema Part 0: Primer".
- [12] W3C REC-xmlschema-1-20010502: "XML Schema Part 1: Structures".
- [13] W3C REC-xmlschema-2-20010502: "XML Schema Part 2: Datatypes".
- [14] W3C REC-xml-names-19990114: "Namespaces in XML".
- [15] 3GPP TS 32.311: "Telecommunication management; Generic Integration Reference Point (IRP) management; Requirements"

- [16] 3GPP TS 32.312: "Telecommunication management; Generic Integration Reference Point (IRP) management; Information Service (IS)".
- [17] 3GPP TS 32.316: "Telecommunication management; Generic Integration Reference Point (IRP) management; Solution Set definitions".
- [18] 3GPP TS 32.150: "Telecommunication management; Integration Reference Point (IRP) Concept and definitions".
- [19] 3GPP TS 32.306: "Telecommunication management; Configuration Management (CM); Notification Integration Reference Point (IRP): Solution Set definitions".
- [20] W3C SOAP 1.1 specification (<http://www.w3.org/TR/2000/NOTE-SOAP-20000508/>)
- [21] W3C XPath 1.0 specification (<http://www.w3.org/TR/1999/REC-xpath-19991116>)
- [22] W3C WSDL 1.1 specification (<http://www.w3.org/TR/2001/NOTE-wsdl-20010315>)
- [23] W3C SOAP 1.2 specification (<http://www.w3.org/TR/soap12-part1/>)
- [24] 3GPP TS 32.300: "Telecommunication management; Configuration Management (CM); Name convention for Managed Objects"

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in 3GPP TS 32.101 [1], 3GPP TS 32.102 [2], 3GPP TS 32.150 [18] and 3GPP TS 32.501 [3] and the following apply:

IRP document version number string (or "IRPVersion"): See 3GPP TS 32.311 [15]

IRP: See 3GPP TS 32.150 [18].

IRPAgent: See 3GPP TS 32.150 [18].

IRPManager: See 3GPP TS 32.150 [18].

Self Configuration: The process which brings a network element into service requiring minimal human operator intervention or none at all.

XML file: file containing an XML document

XML document: composed of the succession of an optional XML declaration followed by a root XML element

NOTE: See [10]; in the scope of the present document.

XML declaration: it specifies the version of XML being used

NOTE: See [10].

XML element: has a type, is identified by a name, may have a set of XML attribute specifications and is either composed of the succession of an XML start-tag followed by the XML content of the XML element followed by an XML end-tag, or composed simply of an XML empty-element tag; each XML element may contain other XML elements

NOTE: See [10].

empty XML element: having an empty XML content; an empty XML element still possibly has a set of XML attribute specifications; an empty XML element is either composed of the succession of an XML start-tag directly followed by an XML end-tag, or composed simply of an XML empty-element tag

NOTE: See [10].

XML content (of an XML element): empty if the XML element is simply composed of an XML empty-element tag; otherwise the part, possibly empty, of the XML element between its XML start-tag and its XML end-tag

XML start-tag: the beginning of a non-empty XML element is marked by an XML start-tag containing the name and the set of XML attribute specifications of the XML element

NOTE: See [10].

XML end-tag: the end of a non-empty XML element is marked by an XML end-tag containing the name of the XML element

NOTE: See [10].

XML empty-element tag: composed simply of an empty-element tag containing the name and the set of XML attribute specifications of the XML element.

NOTE: See [10].

XML attribute specification: has a name and a value

NOTE: See [10].

DTD: defines structure and content constraints to be respected by an XML document to be valid with regard to this DTD

NOTE: See [10].

XML schema: more powerful than a DTD, an XML schema defines structure and content constraints to be respected by an XML document to conform with this XML schema; through the use of XML namespaces several XML schemas can be used together by a single XML document; an XML schema is itself also an XML document that shall conform with the XML schema for XML schemas

NOTE: See [11], [12] and [13].

XML namespace: enables qualifying element and attribute names used in XML documents by associating them with namespaces identified by different XML schemas

NOTE: See [14], in the scope of the present document.

XML complex type: defined in an XML schema; cannot be directly used in an XML document; can be the concrete type or the derivation base type for an XML element type or for another XML complex type; ultimately defines constraints for an XML element on its XML attribute specifications and/or its XML content

NOTE: See [11], [12] and [13].

XML element type: declared by an XML schema; can be directly used in an XML document; as the concrete type of an XML element, directly or indirectly defines constraints on its XML attribute specifications and/or its XML content; can also be the concrete type or the derivation base type for another XML element type

NOTE: See [11], [12] and [13].

3.2 Abbreviations

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

IS	Information Service
SC	Self Configuration
SS	Solution Set
WSDL	Web Service Description Language
XML	eXtensible Markup Language

4 Solution Set definitions

This specification defines the following 3GPP Self-Configuration of Network Elements IRP Solution Set definitions:

Annex A provides the CORBA Solution Set.

Annex B provides the XML Definitions.

Annex C provides the SOAP Solution Set.

Annex A (normative): CORBA Solution Set

This annex contains the CORBA Solution Set for the IRP whose semantics is specified in Self-Configuration of Network Elements IRP: Information Service (3GPP TS 32.502 [4]).

A.1 Architectural features

The overall architectural feature of Self-Configuration of Network Elements IRP is specified in 3GPP TS 32.502 [4].

A.1.1 Syntax for Distinguished Names

The format of a Distinguished Name is defined in 3GPP TS 32.300 [24].

The version of this IRP is represented as a string (see also clause 3 for versions).

A.1.2 Notification Services

Notifications are sent according to the Notification IRP: CORBA SS (see 3GPP TS 32.306 [19]).

The contents of the SCIRP notifications are defined in the present document.

A.1.3 Push and Pull Style

OMG Notification Service defines two styles of interaction. One is called push style. In this style, IRPAgent pushes notifications to IRPManager as soon as they are available. The other is called pull style. In this style, IRPAgent keeps the notifications till IRPManager requests for them.

This CORBA SS specifies that support of Push style is Mandatory (M) and that support of Pull style is Optional (O).

A.1.4 Support multiple notifications in one push operation

For efficiency reasons, IRPAgent may send multiple notifications using one single push operation. To pack multiple notifications into one push operation, IRPAgent may wait and not invoke the push operation as soon as notifications are available. To avoid IRPAgent to wait for an extended period of time that is objectionable to IRPManager, IRPAgent shall implement an IRPAgent wide timer configurable by administrator. On expiration of this timer, IRPAgent shall invoke push if there is at least one notification to be conveyed to IRPManager. This timer is re-started after each push invocation.

A.2 Mapping

A.2.1 Operation and Notification mapping

Software Management IRP: IS 3GPP TS (see 3GPP TS 32.502 [4]) defines semantics of operations and notifications visible across the Itf-N. Table A.2.1 indicates mapping of these operations and notifications to their equivalents defined in this SS.

Table A.2.1: Mapping from IS Notification/Operation to SS equivalents

IS Operation/ notification Self-Configuration IRP: IS 3GPP TS 32.502[4]	SS Method	Qualifier
listScManagementCapabilities	listScManagementCapabilities	M
listScManagementProfiles	listScManagementProfiles	M
createScManagementProfile	createScManagementProfile	M
deleteScManagementProfile	deleteScManagementProfile	M
listScProcesses	listScProcesses	M
resumeScProcess	resumeScProcess	M
swFallback	swFallback	M
terminateScManagementProcess	terminateScManagementProcess	M
changeScManagementProfile	changeScManagementProfile	O
notifyScManagementProfileCreation	notifyScManagementProfileCreation	M
notifyScManagementProfileDeletion	notifyScManagementProfileDeletion	M
notifyScManagementProcessCreation	notifyScManagementProcessCreation	M
notifyScManagementProcessStage	notifyScManagementProcessStage	M
notifyScManagementProcessDeletion	notifyScManagementProcessDeletion	M
notifyNewScManagementCapabilitiesAvailability	notifyNewScManagementCapabilitiesAvailability	O
notifyScManagementProfileChange	notifyScManagementProfileChange	O

A.2.2 Operation parameter mapping

Reference 3GPP TS 32.302 defines semantics of parameters carried in operations across the Itf-N. The following set of tables indicate the mapping of these parameters, as per operation, to their equivalents defined in this SS.

Table A.2.2-1: Mapping from IS listScManagementCapabilities parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
result	Exceptions: SelfConfIRPConstDefs::ListScManagementCapabilities, GenericIRPManagementSystem::ParameterNotSupported, GenericIRPManagementSystem::InvalidParameter, GenericIRPManagementSystem::ValueNotSupported, GenericIRPManagementSystem::OperationNotSupported	M

For all other parameters see TS 32.536 [9] mapping from IS **listSwMCapabilities** parameters to SS equivalents

Table A.2.2-2: Mapping from IS listScManagementProfiles parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
result	Exceptions: SelfConfIRPConstDefs::ListScManagementProfile, GenericIRPManagementSystem::ParameterNotSupported, GenericIRPManagementSystem::InvalidParameter, GenericIRPManagementSystem::ValueNotSupported, GenericIRPManagementSystem::OperationNotSupported	M

For all other parameters see TS 32.536 [9] mapping from IS **listSwMProfiles** parameters to SS equivalents

Table A.2.2-3: Mapping from IS createScManagementProfile parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
result	Exceptions: SelfConfIRPConstDefs::CreateScManagementProfile, GenericIRPManagementSystem::ParameterNotSupported, GenericIRPManagementSystem::InvalidParameter, GenericIRPManagementSystem::ValueNotSupported, GenericIRPManagementSystem::OperationNotSupported	M

For all other parameters see TS 32.536 [9] mapping from IS **createSwMProfile** parameters to SS equivalents

Table A.2.2-4: Mapping from IS deleteScManagementProfile parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
result	Exceptions: SelfConfIRPConstDefs::DeleteScManagementProfile, GenericIRPManagementSystem::ParameterNotSupported, GenericIRPManagementSystem::InvalidParameter, GenericIRPManagementSystem::ValueNotSupported, GenericIRPManagementSystem::OperationNotSupported	M

For all other parameters see TS 32.536 [9] mapping from IS **deleteSwMProfile** parameters to SS equivalents

Table A.2.2-5: Mapping from IS listScManagementProcesses parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
Result	Exceptions: SelfConfIRPConstDefs::ListScManagementProcesses, GenericIRPManagementSystem::ParameterNotSupported, GenericIRPManagementSystem::InvalidParameter, GenericIRPManagementSystem::ValueNotSupported, GenericIRPManagementSystem::OperationNotSupported	M

For all other parameters see TS 32.536 [9] mapping from IS **listSwMProcesses** parameters to SS equivalents

Table A.2.2-6: Mapping from IS resumeScManagementProcess parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
Result	Exceptions: SelfConfIRPConstDefs::ResumeScManagementProcess, GenericIRPManagementSystem::ParameterNotSupported, GenericIRPManagementSystem::InvalidParameter, GenericIRPManagementSystem::ValueNotSupported, GenericIRPManagementSystem::OperationNotSupported	M

For all other parameters see TS 32.536 [9] mapping from IS **resumeSwMProcess** parameters to SS equivalents

Table A.2.2-7: Mapping from IS swFallback parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
Result	Exceptions: SelfConfIRPConstDefs::SwFallback, GenericIRPManagementSystem::ParameterNotSupported, GenericIRPManagementSystem::InvalidParameter, GenericIRPManagementSystem::ValueNotSupported, GenericIRPManagementSystem::OperationNotSupported	M

For all other parameters see TS 32.536 [9] mapping from IS **swFallback** parameters to SS equivalents

Table A.2.2-8: Mapping from IS terminateScManagementProcess parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
Result	Exceptions: SelfConfIRPConstDefs::TerminateScManagementProcess, GenericIRPManagementSystem::ParameterNotSupported, GenericIRPManagementSystem::InvalidParameter, GenericIRPManagementSystem::ValueNotSupported, GenericIRPManagementSystem::OperationNotSupported	M

For all other parameters see TS 32.536 [9] mapping from IS **terminateSwMProcess** parameters to SS equivalents

Table A.2.2-9: Mapping from IS `changeScManagementProfile` parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
Result	Exceptions: SelfConfIRPConstDefs::CreateScManagementProfile, GenericIRPManagementSystem::ParameterNotSupported, GenericIRPManagementSystem::InvalidParameter, GenericIRPManagementSystem::ValueNotSupported, GenericIRPManagementSystem::OperationNotSupported	M

For all other parameters see TS 32.536 [9] mapping from IS **changeSwMProfile** parameters to SS equivalents

A.2.3 Notification parameter mapping

Reference 3GPP TS 32.502 [4] defines semantics of parameters carried in notifications. The following tables indicate the mapping of these parameters to their SS equivalents."

The following tables indicate the mapping of these parameters to their OMG CORBA Structured Event (defined in OMG Notification Service [6]) equivalents. The composition of OMG Structured Event, as defined in the OMG Notification Service [6], is:

```

Header
  Fixed Header
    domain_name
    type_name
    event_name
  Variable Header
Body
  filterable_body_fields
  remaining_body

```

The following tables list all OMG Structured Event attributes in the second column. The first column identifies the Software Management IRP: IS [8] defined notification parameters.

Table A.2.3.1: Mapping for notifyScManagementProfileCreation

See TS 32.536 [9] mapping from IS **notifySwMProfileCreation**

Table A.2.3.2: Mapping for notifyScManagementProfileDeletion

See TS 32.536 [9] mapping from IS **notifySwMProfileDeletion**

Table A.2.3.3: Mapping for notifyScProcessCreation

See TS 32.536 [9] mapping from IS **notifySwMProcessCreation**

Table A.2.3.4: Mapping for notifyScProcessStage

See TS 32.536 [9] mapping from IS **notifySwMProcessStage**

Table A.2.3.5: Mapping for notifyScProcessDeletion

See TS 32.536 [9] mapping from IS **notifySwMProcessDeletion**

Table A.2.3.6: Mapping for notifyNewScManagementCapabilityAvailability

IS Parameters	<SS> Parameters	Qualifier	Comment
id	SwMIRPConstDefs::Id	M	
nEInformation	SwMIRPConstDefs::NEInformation	M	
stepsAndOfferedStopPointList	SwMIRPConstDefs::StepsAndOfferedStopPointList	M	
offeredFinalAdministrativeStateInformation	SwMIRPConstDefs::OfferedFinalAdministrativeStateInformation	M	
swVersionToBeInstalledOfferList	SwMIRPConstDefs::SwVersionToBeInstalledOfferListConditional	C	

Table A.2.3.7: Mapping for notifyScManagementProfileChange

See TS 32.536 [9] mapping from IS **notifySwMProfileChange**

A.3 Solution Set definitions

A.3.1 IDL definition structure

Clause A.3.2 defines the constants and types used by the SC Management IRP.

Clause A.3.3 defines the operations which are performed by the SC Management IRP agent.

Clause A.3.4 defines the notifications which are emitted by the SC Management IRP agent.

A.3.2 IDL specification (file name "SelfConfIRPConstDefs.idl")

```
// File: SelfConfIRPConstDefs.idl
#ifndef _SELF_CONF_IRP_CONST_DEFS_IDL_
#define _SELF_CONF_IRP_CONST_DEFS_IDL_

#include <KernelCmConstDefs.idl>
#include <NotificationIRPConstDefs.idl>
#include <SwMIRPConstDefs.idl>

// This statement must appear after all include statements
#pragma prefix "3gppsa5.org"

/* ## Module: SelfConfIRPConstDefs */

module SelfConfIRPConstDefs
{

/*****
/* definition of types used in operations for Self-Configuration : */
*****/

/* currently all types used in operations imported from 32.536 */

/*****
/* definition of types in notifications for self-configuration : */
*****/

    enum TriggerForDeletion { IRP_AGENT_TERMINATION, IRP_MANAGER_TERMINATION,
    SELF_CONFIGURATION_SUCCESFULLY_CONCLUDED };

interface AttributeNameValue
{
    const string ID = "ID";
    const string NE_INFORMATION = "NE_INFORMATION";
    const string STEPS_AND_OFFERED_STOP_POINT_LIST = "STEPS_AND_OFFERED_STOP_POINT_LIST";
    const string OFFERED_FINAL_ADMINISTRATIVE_STATE_INFORMATION =
"OFFERED_FINAL_ADMINISTRATIVE_STATE_INFORMATION";
    const string SW_VERSION_TO_BE_INSTALLED_OFFER_LIST = "SW_VERSION_TO_BE_INSTALLED_OFFER_LIST";
};

};

#endif // _SELF_CONF_IRP_CONST_DEFS_IDL_
```

A.3.3 IDL specification (file name "SelfConfIRPSystem.idl")

```

//File: SelfConfIRPSystem.idl
#ifndef _SELF_CONF_IRP_SYSTEM_IDL_
#define _SELF_CONF_IRP_SYSTEM_IDL_

#include <SelfConfIRPConstDefs.idl>
#include <SwMIRPConstDefs.idl>
#include <GenericIRPManagementSystem.idl>

// This statement must appear after all include statements
#pragma prefix "3gppsa5.org"

/* ## Module: SelfConfIRPSystem */

module SelfConfIRPSystem
{
    /*
    If the system fails to complete an operation, then it can provide a reason
    to qualify the exception. The semantics carried in this reason are outside
    the scope of the present document.
    */
    exception ListScManagementCapabilities { string reason; };
    exception ListScManagementProfiles { string reason; };
    exception CreateScManagementProfile { string reason; };
    exception DeleteScManagementProfile { string reason; };
    exception ListScProcesses { string reason; };
    exception ResumeScProcess { string reason; };
    exception SwFallback { string reason; };
    exception TerminateScProcess { string reason; };
    exception ChangeScManagementProfile { string reason; };

    interface SelfConfIRPOperations_1
    {
        /* for the purpose of this operation see 3GPP TS 32.502 */
        SwMIRPConstDefs::Result listScManagementCapabilities
        (
            in SwMIRPConstDefs::NEInformationOpt nEInformation,
            out SwMIRPConstDefs::CapabilitiesList capabilitiesList
        )
        raises
        (
            ListScManagementCapabilities,
            GenericIRPManagementSystem::ParameterNotSupported,
            GenericIRPManagementSystem::InvalidParameter,
            GenericIRPManagementSystem::ValueNotSupported,
            GenericIRPManagementSystem::OperationNotSupported
        );

        /* for the purpose of this operation see 3GPP TS 32.502 */
        SwMIRPConstDefs::Result listScProfiles
        (
            in SwMIRPConstDefs::NEInformationOpt nEInformation,
            out SwMIRPConstDefs::ProfileList profileList
        )
        raises
        (
            ListScProfiles,
            GenericIRPManagementSystem::ParameterNotSupported,
            GenericIRPManagementSystem::InvalidParameter,
            GenericIRPManagementSystem::ValueNotSupported,
            GenericIRPManagementSystem::OperationNotSupported
        );

        /* for the purpose of this operation see 3GPP TS 32.502 */
        SwMIRPConstDefs::Result createScManagementProfile
        (
            in SwMIRPConstDefs::IdOpt id,
            in SwMIRPConstDefs::NEInformation nEInformation,
            in SwMIRPConstDefs::SwVersionToBeInstalledOpt swVersionToBeInstalled,
            in SwMIRPConstDefs::StepsAndSelectedStopPointList stepsAndSelectedStopPointList,
            in SwMIRPConstDefs::SelectedFinalAdministrativeState selectedFinalAdministrativeState
        )
    }
}

```

```
)
raises
(
  CreateScManagementProfile,
  GenericIRPManagementSystem::ParameterNotSupported,
  GenericIRPManagementSystem::InvalidParameter,
  GenericIRPManagementSystem::ValueNotSupported,
  GenericIRPManagementSystem::OperationNotSupported
);

/* for the purpose of this operation see 3GPP TS 32.502 */
SwMIRPConstDefs::Result deleteScManagementProfile
(
  in SwMIRPConstDefs::Id id
)
raises
(
  DeleteScManagementProfile,
  GenericIRPManagementSystem::ParameterNotSupported,
  GenericIRPManagementSystem::InvalidParameter,
  GenericIRPManagementSystem::ValueNotSupported,
  GenericIRPManagementSystem::OperationNotSupported
);

/* for the purpose of this operation see 3GPP TS 32.502 */
SwMIRPConstDefs::Result listScProcesses
(
  in SwMIRPConstDefs::NEIdentificationOpt nEIdentification,
  out SwMIRPConstDefs::ProcessList processList
)
raises
(
  ListScProcesses,
  GenericIRPManagementSystem::ParameterNotSupported,
  GenericIRPManagementSystem::InvalidParameter,
  GenericIRPManagementSystem::ValueNotSupported,
  GenericIRPManagementSystem::OperationNotSupported
);

/* for the purpose of this operation see 3GPP TS 32.502 */
SwMIRPConstDefs::Result resumeScProcess
(
  in SwMIRPConstDefs::Id id
)
raises
(
  ResumeScProcess,
  GenericIRPManagementSystem::ParameterNotSupported,
  GenericIRPManagementSystem::InvalidParameter,
  GenericIRPManagementSystem::ValueNotSupported,
  GenericIRPManagementSystem::OperationNotSupported
);

/* for the purpose of this operation see 3GPP TS 32.502 */
SwMIRPConstDefs::Result swFallback
(
  in SwMIRPConstDefs::Filter filter,
  out SwMIRPConstDefs::NEList nEList
)
raises
(
  SwFallback,
  GenericIRPManagementSystem::ParameterNotSupported,
  GenericIRPManagementSystem::InvalidParameter,
  GenericIRPManagementSystem::ValueNotSupported,
  GenericIRPManagementSystem::OperationNotSupported
);

/* for the purpose of this operation see 3GPP TS 32.502 */
SwMIRPConstDefs::Result terminateScProcess
(
  in SwMIRPConstDefs::Id id
)
```

```
    raises
    (
        TerminateScProcess,
        GenericIRPManagementSystem::ParameterNotSupported,
        GenericIRPManagementSystem::InvalidParameter,
        GenericIRPManagementSystem::ValueNotSupported,
        GenericIRPManagementSystem::OperationNotSupported
    );
};

interface SelfConfIRPOperations_2
{
    /* for the purpose of this operation see 3GPP TS 32.502 */
    SwMIRPConstDefs::Result changeScManagementProfile
    (
        in SwMIRPConstDefs::Id id,
        in SwMIRPConstDefs::NEInformation nEInformation,
        in SwMConfIRPConstDefs::SwVersionToBeInstalledOpt swVersionToBeInstalled,
        in SwMConfIRPConstDefs::StepsAndSelectedStopPointList stepsAndSelectedStopPointList,
        in SwMIRPConstDefs::SelectedFinalAdministrativeState selectedFinalAdministrativeState
    )
    raises
    (
        ChangeScManagementProfile,
        GenericIRPManagementSystem::ParameterNotSupported,
        GenericIRPManagementSystem::InvalidParameter,
        GenericIRPManagementSystem::ValueNotSupported,
        GenericIRPManagementSystem::OperationNotSupported
    );
};
};

#endif // _SELF_CONF_IRP_SYSTEM_IDL_
```

A.3.4 IDL specification (file name "SelfConfIRPNotifications.idl")

```
//File: SelfConfIRPNotifications.idl
#ifndef _SELF_CONF_IRP_NOTIFICATIONS_IDL_
#define _SELF_CONF_IRP_NOTIFICATIONS_IDL_

#include <SelfConfIRPConstDefs.idl>
#include <SwMIRPConstDefs.idl>
#include <NotificationIRPNotifications.idl>
#include <SwMIRPNotifications.idl>

// This statement must appear after all include statements
#pragma prefix "3gppsa5.org"

/* ## Module: SelfConfIRPNotifications
This contains the specification of notifications of Software Management.
=====
*/
module SelfConfIRPNotifications
{

    /* Constant definitions for the notifyScManagementProfileCreation notification */

    interface NotifyScManagementProfileCreation: SwMIRPNotifications::NotifySwMProfileCreation
    {
        const string EVENT_TYPE = "notifyScManagementProfileCreation";
    };

    /* Constant definitions for the notifyScManagementProfileDeletion notification */

    interface NotifyScManagementProfileDeletion: SwMIRPNotifications::NotifySwMProfileDeletion
    {
        const string EVENT_TYPE = "notifyScManagementProfileDeletion";
    };

    /* Constant definitions for the notifyScProcessCreation notification */

    interface NotifyScProcessCreation: SwMIRPNotifications::NotifySwMProcessCreation
    {
        const string EVENT_TYPE = "notifyScProcessCreation";
    };

    /* Constant definitions for the notifyScProcessStage notification */

    interface NotifyScProcessStage: SwMIRPNotifications::NotifySwMProcessStage
    {
        const string EVENT_TYPE = "notifyScProcessStage";
    };

    /* Constant definitions for the notifyScProcessDeletion notification */

    interface NotifyScProcessDeletion: SwMIRPNotifications::NotifyScProcessDeletion
    {
        const string EVENT_TYPE = "notifyScProcessDeletion";
    };

    /* Constant definitions for the notifyNewScManagementCapabilityAvailability notification */

    interface NotifyNewScManagementCapabiliyAvailability: NotificationIRPNotifications::Notify
    {
        const string EVENT_TYPE = "notifyNewScManagementCapabilityAvailability";

        /**
         * This constant defines the name of the id property,
         * which is transported in the filterable_body_fields.
        */
    };
};

```

```
* The data type for the value of this property is
* SwMIRPConstDefs::Id.
*/
const string ID =
    SelfConfIRPConstDefs::AttributeNameValue::ID;

/**
* This constant defines the name of the nEInformation property,
* which is transported in the filterable_body_fields.
* The data type for the value of this property is
* SwMIRPConstDefs::NEInformation.
*/
const string ID =
    SelfConfIRPConstDefs::AttributeNameValue::NE_INFORMATION;

/**
* This constant defines the name of the stepsAndOfferedStopPointList property,
* which is transported in the remaining_body.
* The data type for the value of this property is
* SwMIRPConstDefs::StepsAndOfferedStopPointList.
*/
const string ID =
    SelfConfIRPConstDefs::AttributeNameValue::STEPS_AND_OFFERED_STOP_POINT_LIST;

/**
* This constant defines the name of the offeredFinalAdministrativeStateInformation property,
* which is transported in the remaining_body.
* The data type for the value of this property is
* SwMIRPConstDefs::OfferedFinalAdministrativeStateInformation.
*/
const string ID =
    SelfConfIRPConstDefs::AttributeNameValue::OFFERED_FINAL_ADMINISTRATIVE_STATE_INFORMATION;

/**
* This constant defines the name of the swVersionToBeInstalledOfferList property,
* which is transported in the remaining_body.
* The data type for the value of this property is
* SwMIRPConstDefs::SwVersionToBeInstalledOfferListOpt.
*/
const string ID =
    SelfConfIRPConstDefs::AttributeNameValue::SW_VERSION_TO_BE_INSTALLED_OFFER_LIST;
};

/* Constant definitions for the notifyScManagementProfileChange notification */
interface NotifyScManagementProfileChange: SwMIRPNotifications::NotifySwMProfileChange
{
    const string EVENT_TYPE = "notifyScManagementProfileChange";
};

};

#endif // _SELF_CONF_IRP_NOTIFICATIONS_IDL_
```

Annex B (normative): XML definitions

This annex contains the XML Definitions for the SC Management IRP for the IRP whose semantics is specified in Trace Management IRP: Information Service (3GPP TS 32.502 [4]).

This XML Definitions specification defines the XML syntax of the SC Management IRP XML Data File.

B.1 Architectural Features

The overall architectural feature of SC Management IRP is specified in 3G TS 32.502 [4]. This clause specifies features that are specific to the XML definitions.

B.1.1 Syntax for Distinguished Names

The syntax of a Distinguished Name is defined in 3GPP TS 32.300 [24].

B.1.2 Notification Services

This annex defines the XML syntax of SC Management IRP notifications that is to be used for the SC Management IRP SOAP Solution Set and in conjunction with Notification Log IRP XML Definitions for Notification Log IRP XML Data File and the NL IRP XML Notification Format.

B.1.3 IOC Definitions

This annex defines the XML syntax for the IOC definitions of the SC Management IRP IS [4], which are used by the XML definitions for the SC Management IRP notifications and the SC Management IRP IS operations.

B.2 Mapping

Not present in the current version of this specification.

B.3 Solution Set definitions

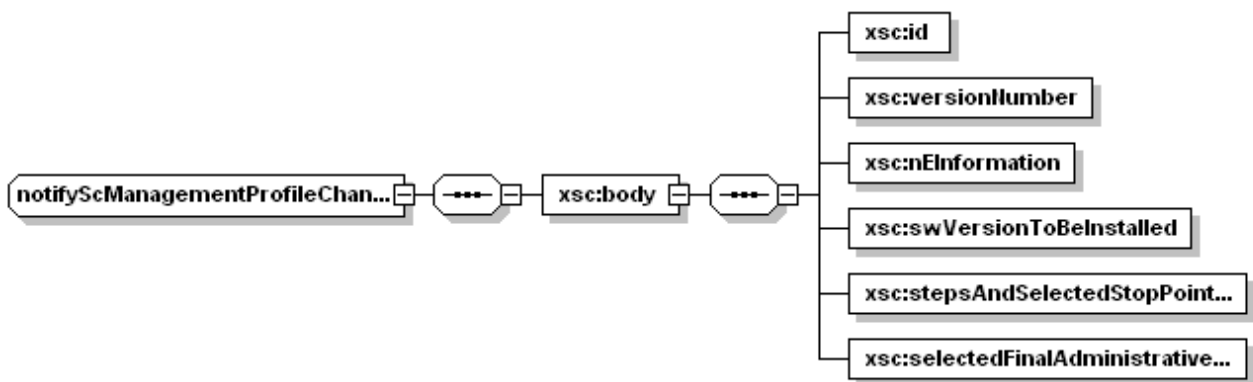
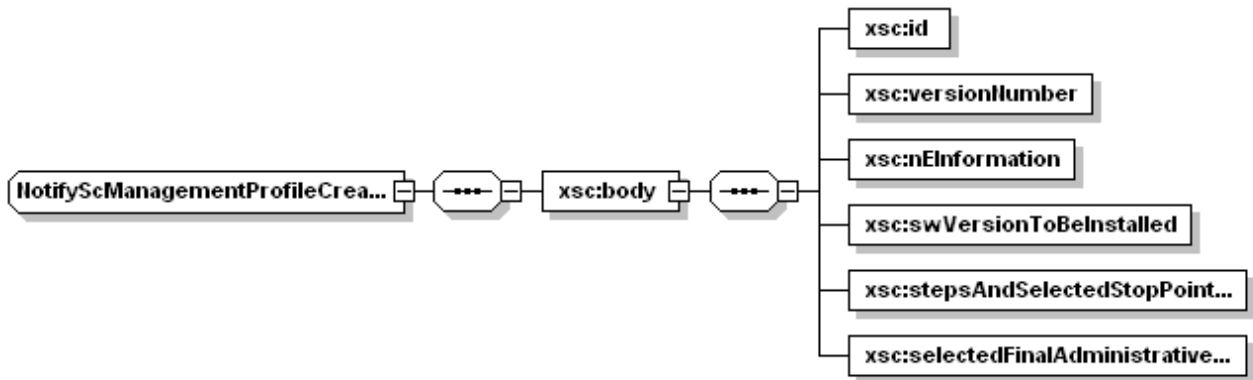
B.3.1 XML definition structure

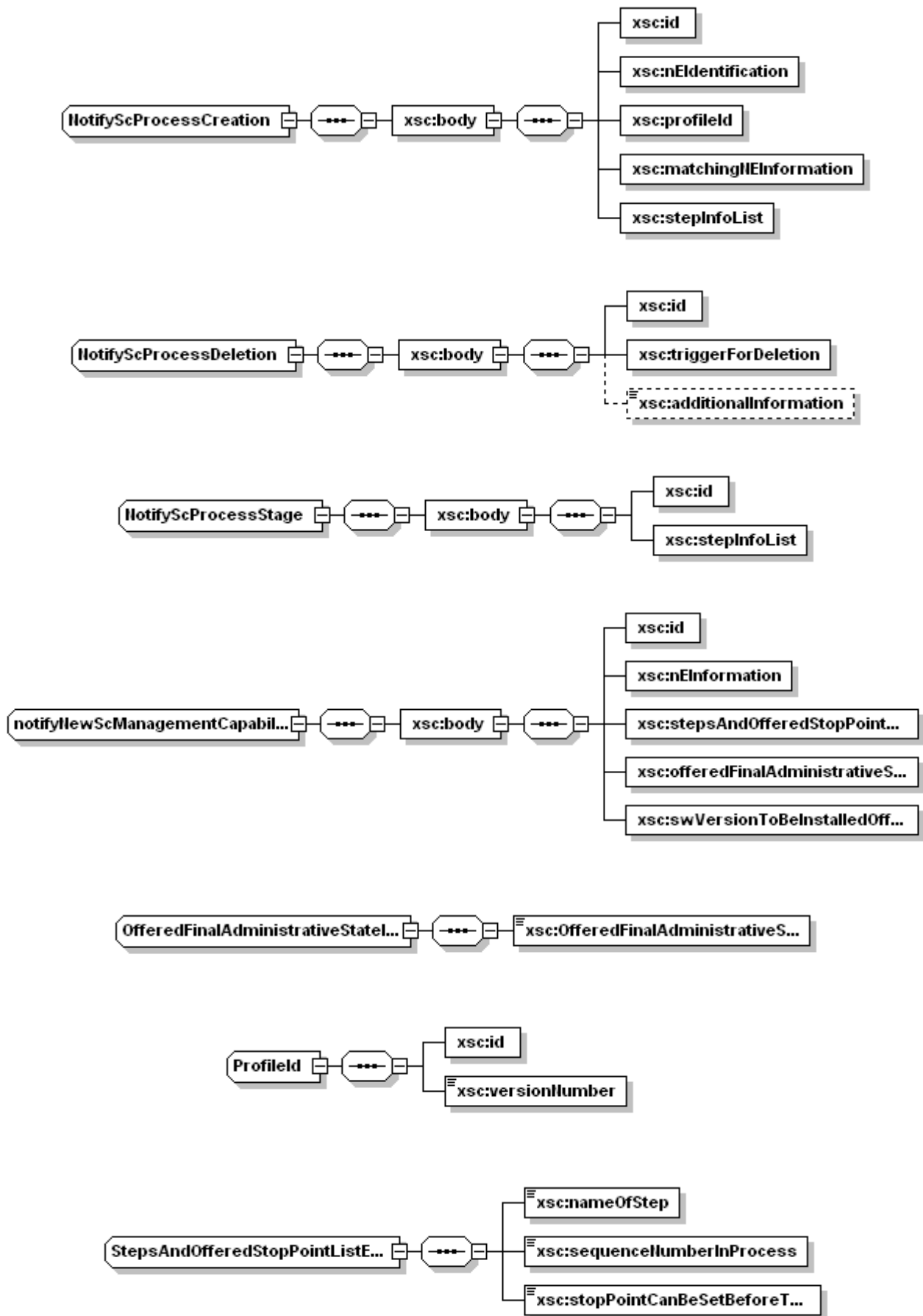
Clause B.3.2 provides a graphical representation of the XML elements.

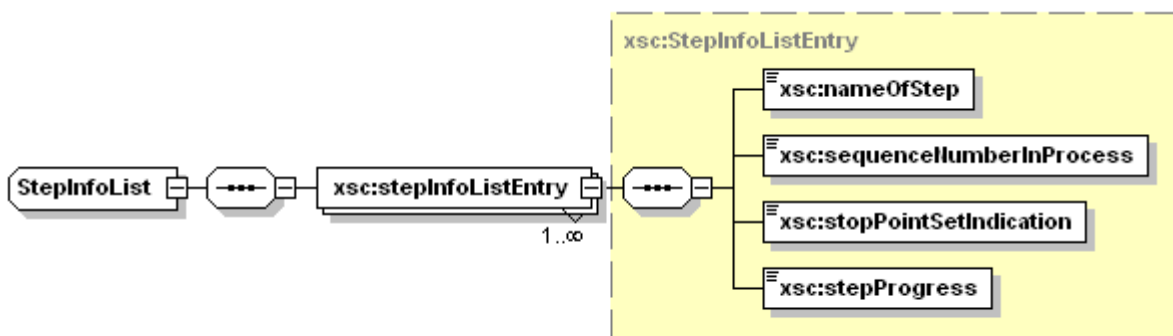
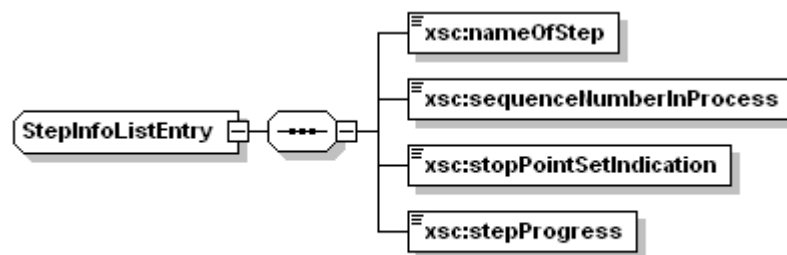
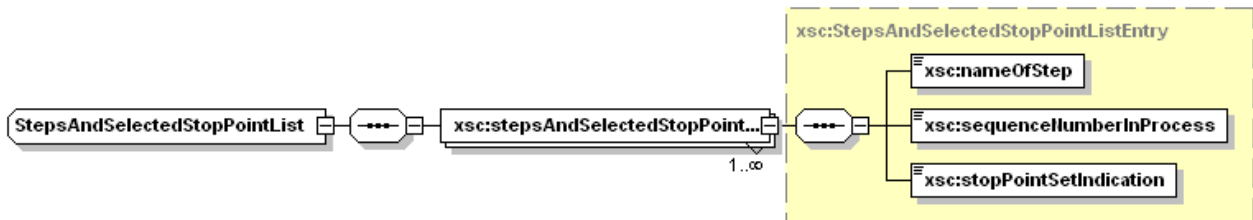
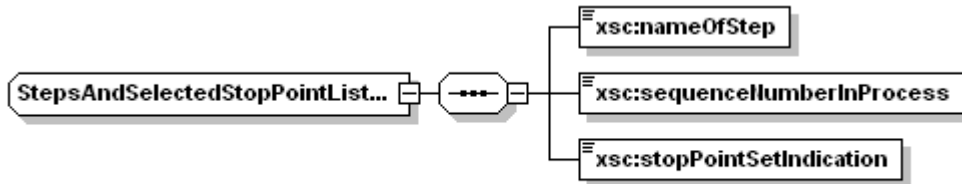
Clause B.3.3 provides XML definitions of SC Management IRP notifications as defined in [4]. These definitions are to be used for the Trace Management IRP SOAP Solution Set.

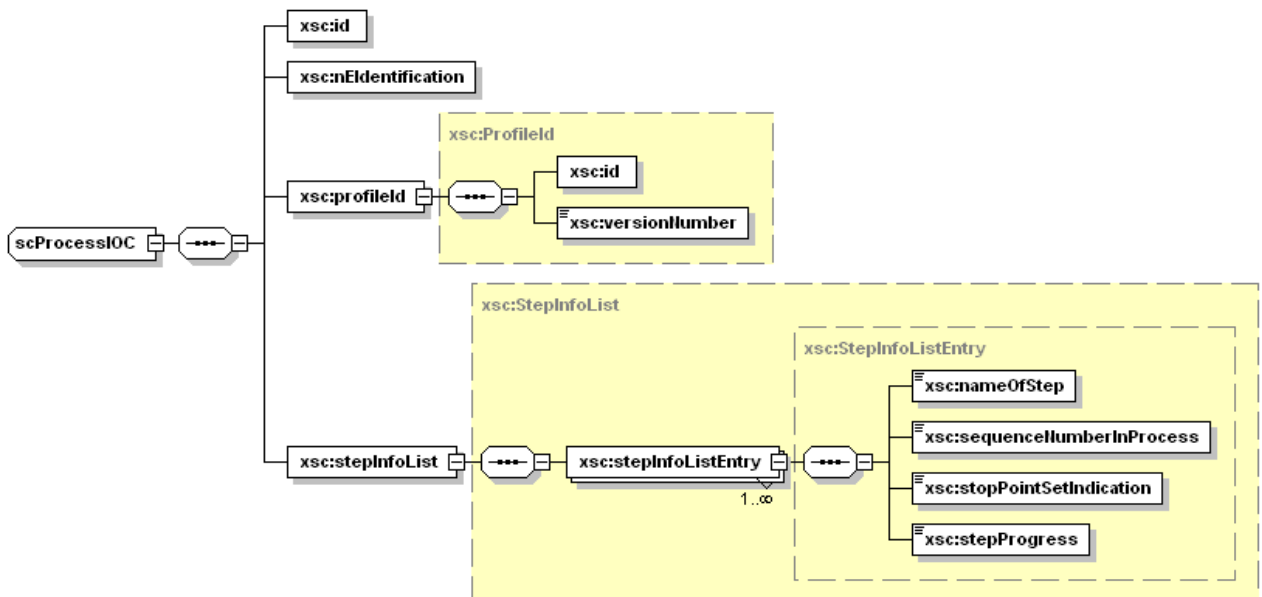
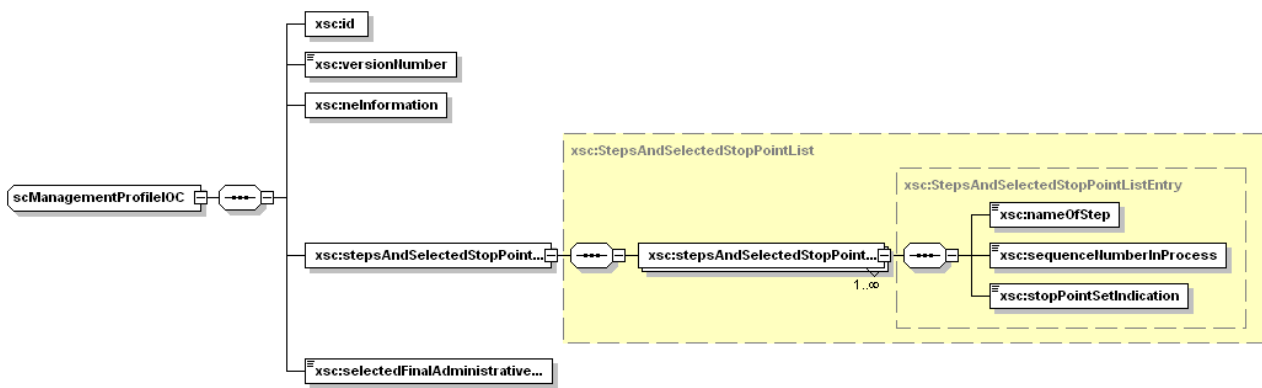
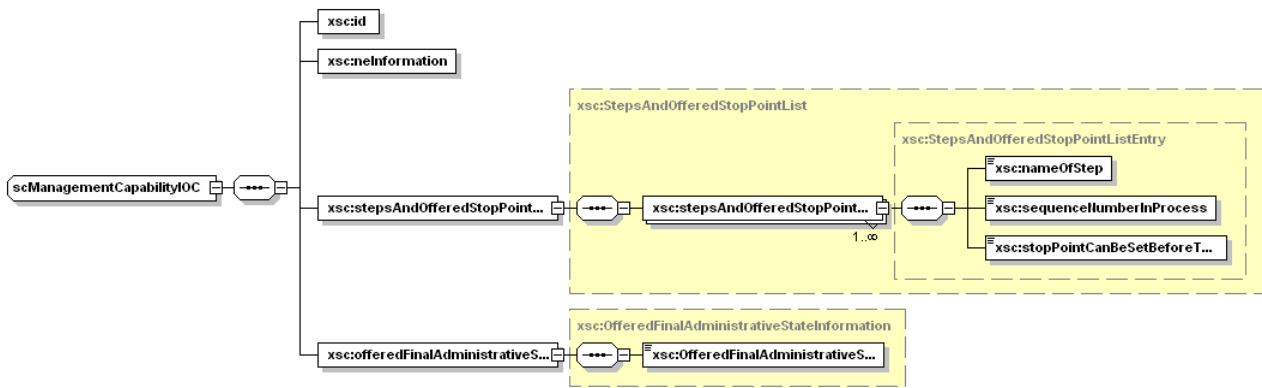
Clause B.3.4 provides XML definitions of SC Management IOC as defined in [4].

B.3.2 Graphical Representation









B.3.3 XML Schema 'sCIRPNotif.xsd'

```

<?xml version="1.0" encoding="UTF-8"?>
<!--
  3GPP TS 32.506 SCIRP Notification XML Schema
  sCIRPNotif.xsd
-->
<schema targetNamespace="http://www.3gpp.org/ftp/specs/archive/32_series/32.506#scnIRPNotif"
xmlns:xsc="http://www.3gpp.org/ftp/specs/archive/32_series/32.506#scnIRPNotif"
xmlns:swm="http://www.3gpp.org/ftp/specs/archive/32_series/32.536#swManagementIRPNotif"
xmlns:xe="http://www.3gpp.org/ftp/specs/archive/32_series/32.306#notification"
xmlns="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified"
attributeFormDefault="unqualified">
  <import namespace="xe:Notification" schemaLocation="notification.xsd"/>
  <complexType name="NotifyScManagementProfileCreation">
    <complexContent>
      <extension base="xe:Notification">
        <sequence>
          <element name="body">
            <complexType>
              <sequence>
                <element name="id" type="swm:Id"/>
                <element name="versionNumber" type="swm:VersionNumber"/>
                <element name="nEInformation" type="swm:NEInformation"/>
                <element name="swVersionToBeInstalled"
type="swm:SwVersionToBeInstalledConditional"/>
                <element name="stepsAndSelectedStopPointList"
type="swm:StepsAndSelectedStopPointList"/>
                <element name="selectedFinalAdministrativeState"
type="swm:FinalAdministrativeStateValue"/>
              </sequence>
            </complexType>
          </element>
        </sequence>
      </extension>
    </complexContent>
  </complexType>
  <complexType name="NotifyScManagementProfileDeletion">
    <complexContent>
      <extension base="xe:Notification">
        <sequence>
          <element name="body">
            <complexType>
              <sequence>
                <element name="id" type="swm:Id"/>
              </sequence>
            </complexType>
          </element>
        </sequence>
      </extension>
    </complexContent>
  </complexType>
  <complexType name="NotifyScProcessCreation">
    <complexContent>
      <extension base="xe:Notification">
        <sequence>
          <element name="body">
            <complexType>
              <sequence>
                <element name="id" type="swm:Id"/>
                <element name="nEIdentification" type="swm:Id"/>
                <element name="profileId" type="swm:ProfileId"/>
                <element name="matchingNEInformation" type="swm:NEInformation"/>
                <element name="stepInfoList" type="swm:StepInfoList"/>
              </sequence>
            </complexType>
          </element>
        </sequence>
      </extension>
    </complexContent>
  </complexType>
  <complexType name="NotifyScProcessStage">
    <complexContent>
      <extension base="xe:Notification">
        <sequence>
          <element name="body">

```

```

        <complexType>
          <sequence>
            <element name="id" type="swm:Id"/>
            <element name="stepInfoList" type="swm:StepInfoList"/>
          </sequence>
        </complexType>
      </element>
    </sequence>
  </extension>
</complexContent>
</complexType>
<complexType name="NotifyScProcessDeletion">
  <complexContent>
    <extension base="xe:Notification">
      <sequence>
        <element name="body">
          <complexType>
            <sequence>
              <element name="id" type="swm:Id"/>
              <element name="triggerForDeletion" type="swm:TriggerForDeletion"/>
              <element name="additionalInformation" type="string" minOccurs="0"/>
            </sequence>
          </complexType>
        </element>
      </sequence>
    </extension>
  </complexContent>
</complexType>
<complexType name="notifyNewScManagementCapabilityAvailability">
  <complexContent>
    <extension base="xe:Notification">
      <sequence>
        <element name="body">
          <complexType>
            <sequence>
              <element name="id" type="swm:Id"/>
              <element name="nEInformation" type="swm:NEInformation"/>
              <element name="stepsAndOfferedStopPointList"
type="swm:StepsAndOfferedStopPointList"/>
              <element name="offeredFinalAdministrativeStateInformation"
type="swm:OfferedFinalAdministrativeStateInformation"/>
              <element name="swVersionToBeInstalledOfferList"
type="swm:swVersionToBeInstalledOfferList"/>
            </sequence>
          </complexType>
        </element>
      </sequence>
    </extension>
  </complexContent>
</complexType>
<complexType name="notifyScManagementProfileChange">
  <complexContent>
    <extension base="xe:Notification">
      <sequence>
        <element name="body">
          <complexType>
            <sequence>
              <element name="id" type="swm:Id"/>
              <element name="versionNumber" type="swm:VersionNumber"/>
              <element name="nEInformation" type="swm:NEInformation"/>
              <element name="swVersionToBeInstalled"
type="swm:SwVersionToBeInstalledConditional"/>
              <element name="stepsAndSelectedStopPointList"
type="swm:StepsAndSelectedStopPointList"/>
              <element name="selectedFinalAdministrativeState"
type="swm:FinalAdministrativeStateValue"/>
            </sequence>
          </complexType>
        </element>
      </sequence>
    </extension>
  </complexContent>
</complexType>
<element name="notifyScManagementProfileCreation" type="swm:NotifyScManagementProfileCreation"/>
<element name="notifyScManagementProfileDeletion" type="swm:NotifyScManagementProfileDeletion"/>
<element name="notifyScManagementProcessCreation" type="swm:NotifyScManagementProcessCreation"/>
<element name="notifyScProcessStage" type="swm:NotifyScProcessStage"/>
<element name="notifyScManagementProcessDeletion" type="swm:NotifyScManagementProcessDeletion"/>

```

```
<element name="notifyNewScManagementCapabilityAvailability"  
type="swm:notifyNewScManagementCapabilityAvailability"/>  
<element name="notifyScManagementProfileChange" type="swm:NotifyScManagementProfileChange"/>  
</schema>
```

B.3.4 XML Schema 'scIRPIOCs.xsd'

```

<?xml version="1.0" encoding="UTF-8"?>
<!--
3GPP TS 32.506 Self Configuration IRP IOC XML Schema
scIRPIOCs.xsd
-->
<schema xmlns="http://www.w3.org/2001/XMLSchema"
xmlns:xsc="http://www.3gpp.org/ftp/specs/archive/32_series/32.506#scIRPIOCs"
xmlns:xn="http://www.3gpp.org/ftp/specs/archive/32_series/32.536#swManagementIRPIOCs"
xmlns:ns1="http://www.3gpp.org/ftp/specs/archive/32_series/32.536#swManagementIRPIOCs"
targetNamespace="http://www.3gpp.org/ftp/specs/archive/32_series/32.506#scIRPIOCs"
elementFormDefault="qualified" attributeFormDefault="unqualified">
  <import namespace="http://www.3gpp.org/ftp/specs/archive/32_series/32.536#swManagementIRPIOCs"
schemaLocation="swManagementIRPIOCs.xsd"/>
  <simpleType name="id">
    <restriction/>
  </simpleType>
  <simpleType name="neInformation">
    <restriction/>
  </simpleType>
  <simpleType name="VersionNumber">
    <restriction base="unsignedShort"/>
  </simpleType>
  <complexType name="ProfileId">
    <sequence>
      <element name="id" type="xsc:Id"/>
      <element name="versionNumber" type="xsc:VersionNumber"/>
    </sequence>
  </complexType>
  <simpleType name="FinalAdministrativeStateValue">
    <restriction base="string">
      <enumeration value="LOCKED"/>
      <enumeration value="UNLOCKED"/>
      <enumeration value="DETERMINED_BY_CONFIGURATION_DATA"/>
    </restriction>
  </simpleType>
  <complexType name="OfferedFinalAdministrativeStateInformation">
    <sequence>
      <element name="OfferedFinalAdministrativeStateValue"
type="xsc:FinalAdministrativeStateValue"/>
    </sequence>
  </complexType>
  <complexType name="StepsAndOfferedStopPointListEntry">
    <sequence>
      <element name="nameOfStep" type="xsc:NameOfStep"/>
      <element name="sequenceNumberInProgress" type="xsc:SequenceNumberInProgress"/>
      <element name="stopPointCanBeSetBeforeThisStep"
type="xsc:StopPointCanBeSetBeforeThisStep"/>
    </sequence>
  </complexType>
  <complexType name="StepsAndOfferedStopPointList">
    <sequence>
      <element name="stepsAndOfferedStopPointListEntry"
type="xsc:StepsAndOfferedStopPointListEntry" maxOccurs="unbounded"/>
    </sequence>
  </complexType>
  <simpleType name="StopPointCanBeSetBeforeThisStep">
    <restriction base="boolean"/>
  </simpleType>
  <complexType name="StepsAndSelectedStopPointListEntry">
    <sequence>
      <element name="nameOfStep" type="xsc:NameOfStep"/>
      <element name="sequenceNumberInProgress" type="xsc:SequenceNumberInProgress"/>
      <element name="stopPointSetIndication" type="xsc:StopPointSetIndication"/>
    </sequence>
  </complexType>
  <complexType name="StepsAndSelectedStopPointList">
    <sequence>
      <element name="stepsAndSelectedStopPointListEntry"
type="xsc:StepsAndSelectedStopPointListEntry" maxOccurs="unbounded"/>
    </sequence>
  </complexType>
  <simpleType name="StopPointSetIndication">
    <restriction base="string">
      <enumeration value="STOP_POINT_IS_SET_BEFORE_THIS_STEP"/>
    </restriction>
  </simpleType>

```

```

        <enumeration value="STOP_POINT_IS_NOT_SET"/>
    </restriction>
</simpleType>
<simpleType name="SequenceNumberInProgress">
    <restriction base="unsignedShort"/>
</simpleType>
<simpleType name="NameOfStep">
    <restriction base="string">
        <enumeration value="SW_DOWNLOAD"/>
        <enumeration value="SW_INSTALLATION"/>
        <enumeration value="SW_ACTIVATION"/>
        <enumeration value="PREPARE_BASIC_CONFIGURATION_AND_OAMLINK"/>
        <enumeration value="RETRIEVE_CONFIGURATION_DATA"/>
        <enumeration value="SETUP_PRECONFIGURED_SIGNALLING_LINKS"/>
        <enumeration value="SET_FINAL_STATE_OF_NE "/>
    </restriction>
</simpleType>
<simpleType name="StepProgress">
    <restriction base="string">
        <enumeration value="NOT_YET_STARTED"/>
        <enumeration value="RUNNING"/>
        <enumeration value="COMPLETED"/>
        <enumeration value="AWAITING_RESUME"/>
        <enumeration value="FAILURE"/>
        <enumeration value="TERMINATED"/>
    </restriction>
</simpleType>
<complexType name="StepInfoListEntry">
    <sequence>
        <element name="nameOfStep" type="xsc:NameOfStep"/>
        <element name="sequenceNumberInProgress" type="xsc:SequenceNumberInProgress"/>
        <element name="stopPointSetIndication" type="xsc:StopPointSetIndication"/>
        <element name="stepProgress" type="xsc:StepProgress"/>
    </sequence>
</complexType>
<complexType name="StepInfoList">
    <sequence>
        <element name="stepInfoListEntry" type="xsc:StepInfoListEntry" maxOccurs="unbounded"/>
    </sequence>
</complexType>
<complexType name="scManagementCapabilityIOC">
    <sequence>
        <element name="id" type="xsc:Id"/>
        <element name="neInformation" type="xsc:NEInformation"/>
        <element name="stepsAndOfferedStopPointList" type="xsc:StepsAndOfferedStopPointList"/>
        <element name="offeredFinalAdministrativeStateInformation"
type="xsc:OfferedFinalAdministrativeStateInformation"/>
    </sequence>
</complexType>
<complexType name="scManagementProfileIOC">
    <sequence>
        <element name="id" type="xsc:Id"/>
        <element name="versionNumber" type="xsc:VersionNumber"/>
        <element name="neInformation" type="xsc:NEInformation"/>
        <element name="stepsAndSelectedStopPointList" type="xsc:StepsAndSelectedStopPointList"/>
        <element name="selectedFinalAdministrativeState"
type="xsc:FinalAdministrativeStateValue"/>
    </sequence>
</complexType>
<complexType name="scProcessIOC">
    <sequence>
        <element name="id" type="xsc:Id"/>
        <element name="neIdentification" type="xsc:NEInformation"/>
        <element name="profileId" type="xsc:ProfileId"/>
        <element name="stepInfoList" type="xsc:StepInfoList"/>
    </sequence>
</complexType>
</schema>

```


Annex C (normative): SOAP Solution Set

This annex specifies the SOAP Solution Set for the IRP whose semantics are specified in Self-Configuration of Network Elements IRP: Information Service (3GPP TS 32.502 [4]).

C.1 Architectural features

The overall architectural feature of the Self-Configuration of Network Elements IRP is specified in 3GPP TS 32.502 [4]. This clause specifies features that are specific to the SOAP solution set.

C.1.1 Syntax for Distinguished Names

The syntax of a Distinguished Name is defined in 3GPP TS 32.300 [24].

C.1.2 Notification Services

The Self-Configuration of Network Elements IRP SOAP SS uses the Notification IRP SOAP SS of 3GPP TS 32.306[19].

C.1.3 Supported W3C specifications

The SOAP 1.1 specification [20] and WSDL 1.1 specification [22] are supported.

The SOAP 1.2 specification [23] is supported optionally.

This specification uses "document" style in WSDL file.

This specification uses "literal" encoding style in WSDL file.

The filter language used in the SS is the XPath Language (see W3C XPath 1.0 specification [21]). IRPAgents may throw a FilterComplexityLimit fault when a given filter is too complex.

Relevant definitions are imported from the Self-Configuration of Network Elements XML definitions of Annex B

C.1.4 Prefixes and namespaces

This specification uses a number of namespace prefixes throughout that are listed in Table C.1.4.

Table C.1.4: Prefixes and Namespaces used in this specification

PREFIX	NAMESPACE
(no prefix)	http://schemas.xmlsoap.org/wsdl/
soap	http://schemas.xmlsoap.org/wsdl/soap/
SciRPSSystem	http://www.3gpp.org/ftp/specs/archive/32_series/32.506#SciRPSSystem
SciRPData	http://www.3gpp.org/ftp/specs/archive/32_series/32.506#SciRPData
xsc	http://www.3gpp.org/ftp/specs/archive/32_series/32.506#scIRPIOCs
genericRPSSystem	http://www.3gpp.org/ftp/specs/archive/32_series/32.316#GenericRPSSystem
ntfIRPntfSystem	http://www.3gpp.org/ftp/specs/archive/32_series/32.306#NotificationIRPntfSystem

C.2 Mapping

C.2.1 Operation and Notification mapping

The Self-Configuration of Network Elements IRP IS (3GPP TS 32.502 [4]) defines the operations and their semantics.

Table C.2.1 maps the operations defined in the Self-Configuration of Network Elements IRP IS to their equivalent port type and binding operations in this Solution Set (SS).

Table C.2.1 also maps the notifications of the Self-Configuration of Network Elements IRP IS, as well as inherited operations.

Table C.2.1 also qualifies if an operation is Mandatory (M) or Optional (O).

Table C.2.1: Mapping from IS Operation to SS Equivalents

IS Operation in 3GPP TS 32.502 [4]	SS: Operation for WSDL port type and WSDL binding	SS: Port of SCIRPService	Qualifier
listScManagementCapabilities	listScManagementCapabilities (note 1)	SCIRPOperation1	M
listScManagementProfiles	listScManagementProfiles (note 1)	SCIRPOperation1	M
createScManagementProfile	createScManagementProfile (note1)	SCIRPOperation1	M
deleteScManagementProfile	deleteScManagementProfile (note1)	SCIRPOperation1	M
listScProcesses	listScProcesses (note1)	SCIRPOperation1	M
resumeScProcess	resumeScProcess (note1)	SCIRPOperation1	M
terminateScProcess	terminateScProcess (note1)	SCIRPOperation1	M
changeScManagementProfile	changeScManagementProfile (note1)	SCIRPOperation1	O
notifyScManagementProfileCreation	notify (note 2)	NotificationIRPNtfPort	M
notifyScManagementProfileDeletion	notify (note 2)	NotificationIRPNtfPort	M
notifyScProcessCreation	notify (note 2)	NotificationIRPNtfPort	M
notifyScProcessStage	notify (note 2)	NotificationIRPNtfPort	M
notifyScProcessDeletion	notify (note 2)	NotificationIRPNtfPort	M
notifyNewScManagementCapabilityAvailability	notify (note 2)	NotificationIRPNtfPort	M
notifyScManagementProfileChange	notify (note 2)	NotificationIRPNtfPort	O
getIRPVersion (note 3)	See TS 32.316 [17]	GenericIRPPort	M
getOperationProfile (note 3)	See TS 32.316 [17]	GenericIRPPort	O
getNotificationProfile (note 3)	See TS 32.316 [17]	GenericIRPPort	O
NOTE 1: The operation is under the port type SCIRPSystem:SCIRPPortType and under the binding SCIRPSystem:SCIRPBinding.			
NOTE 2: The IS equivalent maps to an XML definition specified in Annex B, and this being an input parameter to the operation notify under the port type ntfIRPNtfSystem:NotificationIRPNtf and under the binding ntfIRPNtfSystem:NotificationIRPNtf of 3GPP TS 32.306 [19]. This binding is linked to a port of the SCIRPService as indicated in the table above..			
NOTE 3: The IS operation is inherited from the ManagedGenericIRP IOC specified in 3GPP TS 32.312 [16]. This inheritance is by the SelfConfigurationIRP IOC of 3GPP TS 32.502 [4] inheriting from the ManagedGenericIRP IOC. The corresponding binding is linked to a port of the SCIRPService as indicated in the table above.			

C.2.2 Operation parameter mapping

The SCIRP IS (3GPP TS 32.502 [4]) defines semantics of parameters carried in the operations. The tables below show the mapping of these parameters, as per operation, to their equivalents defined in this SS.

C.2.2.1 Operation listScManagementCapabilities

C.2.2.1.1 Input parameters

Table C.2.2.1.1: Mapping from IS listScManagementCapabilities input parameters to SS equivalents

IS Operation parameter	SS WSDL type sub-element used in corresponding input message under corresponding port type operation as indicated in Table C.2.1	Qualifier
nEInformation	nEInformation	M

C.2.2.1.2 Output parameters

Table C.2.2.1.2: Mapping from IS listScManagementCapabilities output parameters to SS equivalents

IS Operation parameter	SS WSDL type sub-element used in corresponding input message under corresponding port type operation as indicated in Table C.2.1	Qualifier
capabilitiesList	capabilitiesList	M
Result	Result	M

C.2.2.1.3 Fault definition

Table C.2.2.1.3: Mapping from IS listScManagementCapabilities exceptions to SS equivalents

Assertion name	SS WSDL type sub-element used in corresponding input message under corresponding port type operation as indicated in Table C.2.1	Qualifier
operation_failed	operation_failed	M

C.2.2.2 Operation listScManagementProfiles

C.2.2.2.1 Input parameters

Table C.2.2.2.1: Mapping from IS listScManagementProfiles input parameters to SS equivalents

IS Operation parameter	SS WSDL type sub-element used in corresponding input message under corresponding port type operation as indicated in Table C.2.1	Qualifier
nEInformation	nEInformation	M

C.2.2.2.2 Output parameters

Table C.2.2.2.2: Mapping from IS listScManagementProfiles output parameters to SS equivalents

IS Operation parameter	SS WSDL type sub-element used in corresponding input message under corresponding port type operation as indicated in Table C.2.1	Qualifier
ScManagementProfileList	ScManagementProfileList	M
result	result	M

C.2.2.3 Operation createScManagementProfile

C.2.2.3.1 Input parameters

Table C.2.2.3.1: Mapping from IS createScManagementProfile input parameters to SS equivalents

IS Operation parameter	SS WSDL type sub-element used in corresponding input message under corresponding port type operation as indicated in Table C.2.1	Qualifier
id	id	O
nEInformation	nEInformation	M
swVersionToBeInstalled	swVersionToBeInstalled	M
stepsAndSelectedStopPointList	stepsAndSelectedStopPointList	M
selectedFinalAdministrativeState	selectedFinalAdministrativeState	M

C.2.2.3.2 Output parameters

Table C.2.2.3.2: Mapping from IS createScManagementProfile output parameters to SS equivalents

IS Operation parameter	SS WSDL type sub-element used in corresponding input message under corresponding port type operation as indicated in Table C.2.1	Qualifier
id	id	CM
result	result	M

C.2.2.4 Operation deleteScManagementProfile

C.2.2.4.1 Input parameters

Table C.2.2.4.1: Mapping from IS deleteScManagementProfile input parameters to SS equivalents

IS Operation parameter	SS WSDL type sub-element used in corresponding input message under corresponding port type operation as indicated in Table C.2.1	Qualifier
id	id	M

C.2.2.4.2 Output parameters

Table C.2.2.4.2: Mapping from IS `deleteScManagementProfile` output parameters to SS equivalents

IS Operation parameter	SS WSDL type sub-element used in corresponding input message under corresponding port type operation as indicated in Table C.2.1	Qualifier
result	result	M

C.2.2.5 Operation listScProcesses

C.2.2.5.1 Input parameters

Table C.2.2.5.1: Mapping from IS `listScProcesses` input parameters to SS equivalents

IS Operation parameter	SS WSDL type sub-element used in corresponding input message under corresponding port type operation as indicated in Table C.2.1	Qualifier
nEIdentification	nEIdentification	O

C.2.2.5.2 Output parameters

Table C.2.2.5.2: Mapping from IS `listScProcesses` output parameters to SS equivalents

IS Operation parameter	SS WSDL type sub-element used in corresponding input message under corresponding port type operation as indicated in Table C.2.1	Qualifier
ScManagementprocessList	ScManagementprocessList	M
result	result	M

C.2.2.6 Operation resumeScProcess

C.2.2.6.1 Input parameters

Table C.2.2.6.1: Mapping from IS `resumeScProcess` input parameters to SS equivalents

IS Operation parameter	SS WSDL type sub-element used in corresponding input message under corresponding port type operation as indicated in Table C.2.1	Qualifier
id	id	M
startStepName	startStepName	M

C.2.2.6.2 Output parameters

Table C.2.2.6.2: Mapping from IS `resumeScProcess` output parameters to SS equivalents

IS Operation parameter	SS WSDL type sub-element used in corresponding input message under corresponding port type operation as indicated in Table C.2.1	Qualifier
result	result	M

C.2.2.7 Operation terminateSctProcess

C.2.2.7.1 Input parameters

Table C.2.2.7.1: Mapping from IS terminateSctProcess input parameters to SS equivalents

IS Operation parameter	SS WSDL type sub-element used in corresponding input message under corresponding port type operation as indicated in Table C.2.1	Qualifier
id	id	M

C.2.2.7.2 Output parameters

Table C.2.2.7.2: Mapping from IS terminateSctProcess output parameters to SS equivalents

IS Operation parameter	SS WSDL type sub-element used in corresponding input message under corresponding port type operation as indicated in Table C.2.1	Qualifier
result	result	M

C.2.2.8 Operation changeScManagementProfile

C.2.2.8.1 Input parameters

Table C.2.2.8.1: Mapping from IS changeScManagementProfile input parameters to SS equivalents

IS Operation parameter	SS WSDL type sub-element used in corresponding input message under corresponding port type operation as indicated in Table C.2.1	Qualifier
id	id	M
nEInformation	nEInformation	M
swVersionToBeInstalled	swVersionToBeInstalled	M
stepsAndSelectedStopPointList	stepsAndSelectedStopPointList	M
selectedFinalAdministrativeState	selectedFinalAdministrativeState	M

C.2.2.8.2 Output parameters

Table C.2.2.8.2: Mapping from IS changeScManagementProfile input parameters to SS equivalents

IS Operation parameter	SS WSDL type sub-element used in corresponding input message under corresponding port type operation as indicated in Table C.2.1	Qualifier
result	result	M
versionNumber	versionNumber	M
conflictingProfileId	conflictingProfileId	C *)

C.3 Solution Set definitions

C.3.1 WSDL definition structure

Clause C.3.2 provides a graphical representation of the Self-Configuration of Network Elements IRP service.

Clause C.3.3 defines the services which are supported the Self-Configuration of Network Elements IRP agent.

C.3.2 Graphical Representation

The WSDL structure is depicted in Figure C.3.2 below, depicting port type, binding and service. The port type contains port type operations, which again contains input, output and fault messages. The binding contains binding operations, which have the same name as the port type operations. The binding connects to a port inside the service.

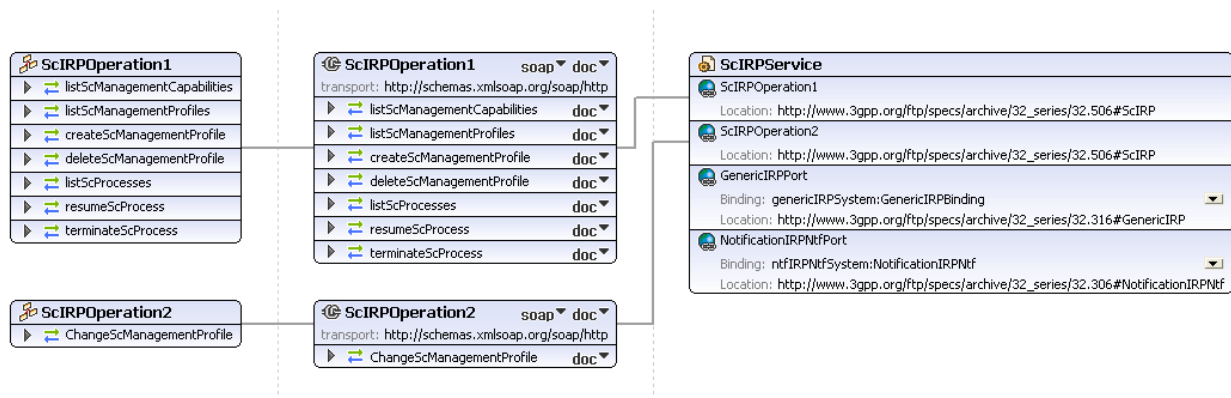


Figure C.3.2: Self-Configuration of Network Elements IRP SOAP Solution Set WSDL structure

C.3.3 WSDL specification 'ScIRPSystem.wsdl'

```

<?xml version="1.0" encoding="UTF-8"?>
<!--
  3GPP TS 32.506 Self Configuration IRP SOAP Solution Set
-->
<definitions xmlns="http://schemas.xmlsoap.org/wsdl/"
  xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
  xmlns:ScIRPSystem="http://www.3gpp.org/ftp/specs/archive/32_series/32.506#ScIRPSystem"
  xmlns:ScIRPData="http://www.3gpp.org/ftp/specs/archive/32_series/32.506#ScIRPData"
  xmlns:xsc="http://www.3gpp.org/ftp/specs/archive/32_series/32.506#ScIRPIOCs"
  xmlns:xn="http://www.3gpp.org/ftp/specs/archive/32_series/32.626#genericNrm"
  xmlns:genericIRPSystem="http://www.3gpp.org/ftp/specs/archive/32_series/32.316#GenericIRP
  System"
  xmlns:ntfIRPntfSystem="http://www.3gpp.org/ftp/specs/archive/32_series/32.306#Notificatio
  nIRPntfSystem" xmlns:ns="http://www.w3.org/2001/XMLSchema"
  xmlns:ns1="http://schemas.xmlsoap.org/soap/encoding/"
  targetNamespace="http://www.3gpp.org/ftp/specs/archive/32_series/32.506#ScIRPSystem">
  <import
    namespace="http://www.3gpp.org/ftp/specs/archive/32_series/32.306#NotificationIRPntfSyste
    m"/>
    <import
      namespace="http://www.3gpp.org/ftp/specs/archive/32_series/32.316#GenericIRPSystem"/>
    <types>
      <schema xmlns="http://www.w3.org/2001/XMLSchema"
        targetNamespace="http://www.3gpp.org/ftp/specs/archive/32_series/32.506#ScIRPData">
        <import
          namespace="http://www.3gpp.org/ftp/specs/archive/32_series/32.506#ScIRPIOCs"/>
          <import
            namespace="http://www.3gpp.org/ftp/specs/archive/32_series/32.626#genericNrm"/>
            <!-- The following types are defined for the SC IRP operations -->
            <simpleType name="NEInformation">
              <restriction base="string"/>
            </simpleType>
            <simpleType name="Id">
              <restriction base="xn:dn"/>
            </simpleType>
            <complexType name="capability">
              <sequence>
                <element name="Id" type="ScIRPData:Id"/>
                <element name="NEInformation" type="ScIRPData:NEInformation"/>
                <element name="StepsAndOfferedStopPointList"
                  type="xsc:StepsAndOfferedStopPointList"/>
                <element name="OfferedFinalAdministrativeStateInformation"
                  type="xsc:OfferedFinalAdministrativeStateInformation"/>
              </sequence>
            </complexType>
            <complexType name="capabilitiesList">
              <sequence>
                <element name="capability" type="ScIRPData:capability"
                  maxOccurs="unbounded"/>
              </sequence>
            </complexType>
            <complexType name="ScProfile">
              <sequence>
                <element name="Id" type="ScIRPData:Id"/>
                <element name="VersionNumber" type="xsc:VersionNumber"/>
                <element name="NEInformation" type="ScIRPData:NEInformation"/>
                <element name="StepsAndSelectedStopPointList"
                  type="xsc:StepsAndSelectedStopPointList"/>
                <element name="FinalAdministrativeStateValue"
                  type="xsc:FinalAdministrativeStateValue"/>
                <element name="swVersionToBeInstalled" type="xsc:VersionNumber"
                  minOccurs="0"/>
              </sequence>
            </complexType>
            <complexType name="ScProfileList">
              <sequence>

```



```

        <element name="ScProfile" type=" ScIRPData:ScProfile"
maxOccurs="unbounded"/>
    </sequence>
</complexType>
<complexType name="ScProcess">
    <sequence>
        <element name="Id" type="ScIRPData:Id"/>
        <element name="NEInformation" type="ScIRPData:NEInformation"/>
        <element name="ProfileId" type="xsc:ProfileId"/>
        <element name="StepInfoList" type="xsc:StepInfoList"/>
    </sequence>
</complexType>
<complexType name="ScProcessList">
    <sequence>
        <element name="ScProcess" type="ScIRPData:ScProcess" maxOccurs="unbounded"/>
    </sequence>
</complexType>
<complexType name="startStepName">
    <sequence>
        <element name="startStepName" type="string"/>
    </sequence>
</complexType>
<!--listScManagementCapabilities Request-->
<element name="listScManagementCapabilitiesRequest">
    <complexType>
        <sequence>
            <element name="nEInformation" type="ScIRPData:NEInformation"/>
        </sequence>
    </complexType>
</element>
<!--listScManagementCapabilities Response -->
<element name="listScManagementCapabilitiesResponse">
    <complexType>
        <sequence>
            <element name="capabilitiesList" type="ScIRPData:capabilitiesList"/>
            <element name="Result">
                <simpleType>
                    <restriction base="string">
                        <enumeration value="success"/>
                        <enumeration value="failure"/>
                        <enumeration value="stepNameNotMatch"/>
                        <enumeration value="nEInformationIntersection"/>
                    </restriction>
                </simpleType>
            </element>
        </sequence>
    </complexType>
</element>
<!--listScManagementCapabilities Fault -->
<element name="listScManagementCapabilitiesFault">
    <simpleType>
        <restriction base="string">
            <enumeration value="operationfailed"/>
        </restriction>
    </simpleType>
</element>
<!--listScManagementProfiles Request -->
<element name="listScManagementProfilesRequest">
    <complexType>
        <sequence>
            <element name="nEInformation" type="ScIRPData:NEInformation"/>
        </sequence>
    </complexType>
</element>
<!--listScManagementProfiles Response -->
<element name="listScManagementProfilesResponse">
    <complexType>
        <sequence>
            <element name="ScProfileList" type="ScIRPData:ScProfileList"/>
            <element name="Result">

```

```

        <simpleType>
          <restriction base="string">
            <enumeration value="success"/>
            <enumeration value="failure"/>
            <enumeration value="stepNameNotMatch"/>
            <enumeration value="nEInformationIntersection"/>
          </restriction>
        </simpleType>
      </element>
    </sequence>
  </complexType>
</element>
<!--createScManagementProfile Request -->
<element name="createScManagementProfileRequest">
  <complexType>
    <sequence>
      <element name="id" type="ScIRPData:Id" minOccurs="0"/>
      <element name="nEInformation" type="ScIRPData:NEInformation"/>
      <element name="swVersionToBeInstalled" type="xsc:VersionNumber"/>
      <element name="stepsAndSelectedStopPointList"
type="xsc:StepsAndSelectedStopPointList"/>
      <element name="selectedFinalAdministrativeState"
type="xsc:FinalAdministrativeStateValue"/>
    </sequence>
  </complexType>
</element>
<!--createScManagementProfile Response -->
<element name="createScManagementProfileResponse">
  <complexType>
    <sequence>
      <element name="id" type="ScIRPData:Id" minOccurs="0"/>
      <element name="Result">
        <simpleType>
          <restriction base="string">
            <enumeration value="Success"/>
            <enumeration value="Failure"/>
            <enumeration value="nEInformationIntersection"/>
            <enumeration value="notAllowedBecauseOfOngoingSwmActivity"/>
          </restriction>
        </simpleType>
      </element>
    </sequence>
  </complexType>
</element>
<!--deleteScManagementProfile Request -->
<element name="deleteScManagementProfileRequest">
  <complexType>
    <sequence>
      <element name="id" type="ScIRPData:Id"/>
    </sequence>
  </complexType>
</element>
<!--deleteScManagementProfile Response -->
<element name="deleteScManagementProfileResponse">
  <complexType>
    <sequence>
      <element name="Result">
        <simpleType>
          <restriction base="string">
            <enumeration value="success"/>
            <enumeration value="failure"/>
            <enumeration value="stepNameNotMatch"/>
            <enumeration value="nEInformationIntersection"/>
          </restriction>
        </simpleType>
      </element>
    </sequence>
  </complexType>
</element>
<!--listScProcesses Request -->

```

```

    <element name="listScProcessesRequest">
      <complexType>
        <sequence>
          <element name="nEIdentification" type="ScIRPData:NEInformation"
minOccurs="0"/>
        </sequence>
      </complexType>
    </element>
    <!--listScProcesses Response -->
    <element name="listScProcessesResponse">
      <complexType>
        <sequence>
          <element name="ScProcessList" type="ScIRPData:ScProcessList"/>
          <element name="Result">
            <simpleType>
              <restriction base="string">
                <enumeration value="success"/>
                <enumeration value="failure"/>
                <enumeration value="stepNameNotMatch"/>
                <enumeration value="nEInformationIntersection"/>
              </restriction>
            </simpleType>
          </element>
        </sequence>
      </complexType>
    </element>
    <!--resumeScProcess Request -->
    <element name="resumeScProcessRequest">
      <complexType>
        <sequence>
          <element name="id" type="ScIRPData:Id"/>
          <element name="startStepName" type="ScIRPData:startStepName"/>
        </sequence>
      </complexType>
    </element>
    <!--resumeScProcess Response -->
    <element name="resumeScProcessResponse">
      <complexType>
        <sequence>
          <element name="Result">
            <simpleType>
              <restriction base="string">
                <enumeration value="success"/>
                <enumeration value="failure"/>
                <enumeration value="stepNameNotMatch"/>
                <enumeration value="nEInformationIntersection"/>
              </restriction>
            </simpleType>
          </element>
        </sequence>
      </complexType>
    </element>
    <!--terminateScProcess Request -->
    <element name="terminateScProcessRequest">
      <complexType>
        <sequence>
          <element name="id" type="ScIRPData:Id"/>
        </sequence>
      </complexType>
    </element>
    <!--terminateScProcess Response -->
    <element name="terminateScProcessResponse">
      <complexType>
        <sequence>
          <element name="Result">
            <simpleType>
              <restriction base="string">
                <enumeration value="success"/>
                <enumeration value="failure"/>
                <enumeration value="stepNameNotMatch"/>
              </restriction>
            </simpleType>
          </element>
        </sequence>
      </complexType>
    </element>

```

```

        <enumeration value="nEInformationIntersection"/>
      </restriction>
    </simpleType>
  </element>
</sequence>
</complexType>
</element>
<!--ChangeScManagementProfile Request -->
<element name="ChangeScManagementProfileRequest">
  <complexType>
    <sequence>
      <element name="id" type="ScIRPData:Id"/>
      <element name="nEInformation" type="ScIRPData:NEInformation"/>
      <element name="swVersionToBeInstalled" type="xsc:VersionNumber"/>
      <element name="stepsAndSelectedStopPointList"
type="xsc:StepsAndSelectedStopPointList"/>
      <element name="selectedFinalAdministrativeState"
type="xsc:FinalAdministrativeStateValue"/>
    </sequence>
  </complexType>
</element>
<!--ChangeScManagementProfile Response -->
<element name="ChangeScManagementProfileResponse">
  <complexType>
    <sequence>
      <element name="Result">
        <simpleType>
          <restriction base="string">
            <enumeration value="Success"/>
            <enumeration value="Failure"/>
            <enumeration value="nEInformationIntersection"/>
            <enumeration value="notAllowedBecauseOfOngoingSwmActivity"/>
          </restriction>
        </simpleType>
      </element>
      <element name="versionNumber" type="xsc:VersionNumber"/>
      <element name="conflictingProfileId" minOccurs="0">
        <simpleType>
          <restriction base="string">
            <enumeration value="swMprofileId"/>
            <enumeration value="swMprofileAId"/>
            <enumeration value="swMprofileBId"/>
            <enumeration value="empty"/>
          </restriction>
        </simpleType>
      </element>
    </sequence>
  </complexType>
</element>
</schema>
</types>
<message name="listScManagementCapabilitiesRequest">
  <part name="parameter" element="ScIRPData:listScManagementCapabilitiesRequest"/>
</message>
<message name="listScManagementCapabilitiesResponse">
  <part name="parameter" element="ScIRPData:listScManagementCapabilitiesResponse"/>
</message>
<message name="listScManagementCapabilitiesFault">
  <part name="parameter" element="ScIRPData:listScManagementCapabilitiesFault"/>
</message>
<message name="listScManagementProfilesRequest">
  <part name="parameter" element="ScIRPData:listScManagementProfilesRequest"/>
</message>
<message name="listScManagementProfilesResponse">
  <part name="parameter" element="ScIRPData:listScManagementProfilesResponse"/>
</message>
<message name="createScManagementProfileRequest">
  <part name="parameter" element="ScIRPData:createScManagementProfileRequest"/>
</message>
<message name="createScManagementProfileResponse">

```

```

    <part name="parameter" element="ScIRPData:createScManagementProfileResponse"/>
  </message>
  <message name="deleteScManagementProfileRequest">
    <part name="parameter" element="ScIRPData:deleteScManagementProfileRequest"/>
  </message>
  <message name="deleteScManagementProfileResponse">
    <part name="parameter" element="ScIRPData:deleteScManagementProfileResponse"/>
  </message>
  <message name="listScProcessesRequest">
    <part name="parameter" element="ScIRPData:listScProcessesRequest"/>
  </message>
  <message name="listScProcessesResponse">
    <part name="parameter" element="ScIRPData:listScProcessesResponse"/>
  </message>
  <message name="resumeScProcessRequest">
    <part name="parameter" element="ScIRPData:resumeScProcessRequest"/>
  </message>
  <message name="resumeScProcessResponse">
    <part name="parameter" element="ScIRPData:resumeScProcessResponse"/>
  </message>
  <message name="terminateScProcessRequest">
    <part name="parameter" element="ScIRPData:terminateScProcessRequest"/>
  </message>
  <message name="terminateScProcessResponse">
    <part name="parameter" element="ScIRPData:terminateScProcessResponse"/>
  </message>
  <message name="ChangeScManagementProfileRequest">
    <part name="parameter" element="ScIRPData:ChangeScManagementProfileRequest"/>
  </message>
  <message name="ChangeScManagementProfileResponse">
    <part name="parameter" element="ScIRPData:ChangeScManagementProfileResponse"/>
  </message>
  <portType name="ScIRPOperation1">
    <operation name="listScManagementCapabilities">
      <input message="ScIRPSystem:listScManagementCapabilitiesRequest"/>
      <output message="ScIRPSystem:listScManagementCapabilitiesResponse"/>
      <fault name="listScManagementCapabilitiesFault"
message="ScIRPSystem:listScManagementCapabilitiesFault"/>
    </operation>
    <operation name="listScManagementProfiles">
      <input message="ScIRPSystem:listScManagementProfilesRequest"/>
      <output message="ScIRPSystem:listScManagementProfilesResponse"/>
    </operation>
    <operation name="createScManagementProfile">
      <input message="ScIRPSystem:createScManagementProfileRequest"/>
      <output message="ScIRPSystem:createScManagementProfileResponse"/>
    </operation>
    <operation name="deleteScManagementProfile">
      <input message="ScIRPSystem:deleteScManagementProfileRequest"/>
      <output message="ScIRPSystem:deleteScManagementProfileResponse"/>
    </operation>
    <operation name="listScProcesses">
      <input message="ScIRPSystem:listScProcessesRequest"/>
      <output message="ScIRPSystem:listScProcessesResponse"/>
    </operation>
    <operation name="resumeScProcess">
      <input message="ScIRPSystem:resumeScProcessRequest"/>
      <output message="ScIRPSystem:resumeScProcessResponse"/>
    </operation>
    <operation name="terminateScProcess">
      <input message="ScIRPSystem:terminateScProcessRequest"/>
      <output message="ScIRPSystem:terminateScProcessResponse"/>
    </operation>
  </portType>
  <portType name="ScIRPOperation2">
    <operation name="ChangeScManagementProfile">
      <input message="ScIRPSystem:ChangeScManagementProfileRequest"/>
      <output message="ScIRPSystem:ChangeScManagementProfileResponse"/>
    </operation>
  </portType>

```

```

<binding name="ScIRPOperation1" type="ScIRPSystem:ScIRPOperation1">
  <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
  <operation name="listScManagementCapabilities">
    <soap:operation
soapAction="http://www.3gpp.org/ftp/specs/archive/32_series/32.506#listScManagementCapabi
lities" style="document"/>
    <input>
      <soap:body use="literal"/>
    </input>
    <output>
      <soap:body use="literal"/>
    </output>
    <fault name="listScManagementCapabilitiesFault"/>
  </operation>
  <operation name="listScManagementProfiles">
    <soap:operation
soapAction="http://www.3gpp.org/ftp/specs/archive/32_series/32.506#listScManagementProfil
es" style="document"/>
    <input>
      <soap:body use="literal"/>
    </input>
    <output>
      <soap:body use="literal"/>
    </output>
  </operation>
  <operation name="createScManagementProfile">
    <soap:operation
soapAction="http://www.3gpp.org/ftp/specs/archive/32_series/32.506#createScManagementProf
ile" style="document"/>
    <input>
      <soap:body use="literal"/>
    </input>
    <output>
      <soap:body use="literal"/>
    </output>
  </operation>
  <operation name="deleteScManagementProfile">
    <soap:operation
soapAction="http://www.3gpp.org/ftp/specs/archive/32_series/32.506#deleteScManagementProf
ile" style="document"/>
    <input>
      <soap:body use="literal"/>
    </input>
    <output>
      <soap:body use="literal"/>
    </output>
  </operation>
  <operation name="listScProcesses">
    <soap:operation
soapAction="http://www.3gpp.org/ftp/specs/archive/32_series/32.506#listScProcesses"
style="document"/>
    <input>
      <soap:body use="literal"/>
    </input>
    <output>
      <soap:body use="literal"/>
    </output>
  </operation>
  <operation name="resumeScProcess">
    <soap:operation
soapAction="http://www.3gpp.org/ftp/specs/archive/32_series/32.506#resumeScProcess"
style="document"/>
    <input>
      <soap:body use="literal"/>
    </input>
    <output>
      <soap:body use="literal"/>
    </output>
  </operation>
  <operation name="terminateScProcess">

```

```

        <soap:operation
soapAction="http://www.3gpp.org/ftp/specs/archive/32_series/32.506#terminateScProcess"
style="document"/>
        <input>
            <soap:body use="literal"/>
        </input>
        <output>
            <soap:body use="literal"/>
        </output>
    </operation>
</binding>
<binding name="ScIRPOperation2" type="ScIRPSystem:ScIRPOperation2">
    <soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>
    <operation name="ChangeScManagementProfile">
        <soap:operation
soapAction="http://www.3gpp.org/ftp/specs/archive/32_series/32.506#ChangeScManagementProf
ile" style="document"/>
        <input>
            <soap:body use="literal"/>
        </input>
        <output>
            <soap:body use="literal"/>
        </output>
    </operation>
</binding>
<service name="ScIRPService">
    <port name="ScIRPOperation1" binding="ScIRPSystem:ScIRPOperation1">
        <soap:address
location="http://www.3gpp.org/ftp/specs/archive/32_series/32.506#ScIRP"/>
    </port>
    <port name="GenericIRPPort" binding="genericIRPSystem:GenericIRPBinding">
        <soap:address
location="http://www.3gpp.org/ftp/specs/archive/32_series/32.316#GenericIRP"/>
    </port>
    <port name="NotificationIRPNtfPort" binding="ntfIRPNtfSystem:NotificationIRPNtf">
        <soap:address
location="http://www.3gpp.org/ftp/specs/archive/32_series/32.306#NotificationIRPNtf"/>
    </port>
    <port name="ScIRPOperation2" binding="ScIRPSystem:ScIRPOperation2">
        <soap:address
location="http://www.3gpp.org/ftp/specs/archive/32_series/32.506#ScIRP"/>
    </port>
</service>
</definitions>

```

Annex D (informative): Change history

Change history							
Date	TSG #	TSG Doc.	CR	Rev	Subject/Comment	Old	New
2010-09	SA#49	SP-100509	--	--	Presentation to SA for Information and Approval	---	1.0.0
2010-10	--	--	--	--	Publication	1.0.0	10.0.0

History

Document history		
V10.0.0	May 2011	Publication