3GPP TS 32.536 V16.0.0 (2020-07)

Technical Specification

3rd Generation Partnership Project;

Technical Specification Group Services and System Aspects;

Telecommunication management;

Software management Integration Reference Point (IRP);

Solution Set (SS) definitions

(Release 16)

* *

The present document has been developed within the 3rd Generation Partnership Project (3GPP TM) and may be further elaborated for the purposes of 3GPP.   
The present document has not been subject to any approval process by the 3GPPOrganizational Partners and shall not be implemented.   
This Specification is provided for future development work within 3GPPonly. The Organizational Partners accept no liability for any use of this Specification.  
Specifications and reports for implementation of the 3GPP TM system should be obtained via the 3GPP Organizational Partners' Publications Offices.

Keywords

Software management,

CORBA, XML, SOAP

***3GPP***

Postal address

3GPP support office address

650 Route des Lucioles - Sophia Antipolis

Valbonne - FRANCE

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

Internet

http://www.3gpp.org

***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.

© 2020, 3GPP Organizational Partners (ARIB, ATIS, CCSA, ETSI, TSDSI, TTA, TTC).

All rights reserved.

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

GSM® and the GSM logo are registered and owned by the GSM Association

Contents

Foreword [5](#__RefHeading___Toc479148664)

Introduction [5](#__RefHeading___Toc479148665)

1 Scope [6](#__RefHeading___Toc479148666)

2 References [6](#__RefHeading___Toc479148667)

3 Definitions and abbreviations [7](#__RefHeading___Toc479148668)

3.1 Definitions [7](#__RefHeading___Toc479148669)

3.2 Abbreviations [7](#__RefHeading___Toc479148670)

4 Solution Set definitions [7](#__RefHeading___Toc479148671)

Annex A (normative): CORBA Solution Set [8](#__RefHeading___Toc479148672)

A.1 Architectural Features [8](#__RefHeading___Toc479148673)

A.1.1 Syntax for Distinguished Names and Versions [8](#__RefHeading___Toc479148674)

A.2 Mapping [8](#__RefHeading___Toc479148675)

A.2.1 Operation and Notification mapping [8](#__RefHeading___Toc479148676)

A.2.2 Operation parameter mapping [9](#__RefHeading___Toc479148677)

A.2.3 Notification parameter mapping [12](#__RefHeading___Toc479148678)

A.3 Solution Set definitions [14](#__RefHeading___Toc479148679)

A.3.1 IDL definition structure [14](#__RefHeading___Toc479148680)

A.3.2 IDL specification “SwMIRPConstDefs.idl” [15](#__RefHeading___Toc479148681)

A.3.3 IDL specification “SwMIRPSystem.idl” [21](#__RefHeading___Toc479148682)

A.3.4 IDL specification “SwMIRPNotifications.idl” [25](#__RefHeading___Toc479148683)

Annex B (normative): XML definitions [31](#__RefHeading___Toc479148684)

B.1 Architectural features [31](#__RefHeading___Toc479148685)

B.1.1 Syntax for Distinguished Names [31](#__RefHeading___Toc479148686)

B.2 Mapping [31](#__RefHeading___Toc479148687)

B.3 Solution Set definitions [31](#__RefHeading___Toc479148688)

B.3.1 XML definition structure [31](#__RefHeading___Toc479148689)

B.3.2 Graphical Representation [31](#__RefHeading___Toc479148690)

B.3.3 XML Schema “swManagementIRPNotif.xsd” [36](#__RefHeading___Toc479148691)

B.3.4 XML Schema “swManagementIRPIOCs.xsd” [40](#__RefHeading___Toc479148692)

Annex C (normative): SOAP Solution Set [43](#__RefHeading___Toc479148693)

C.1 Architectural Features [43](#__RefHeading___Toc479148694)

C.1.1 Syntax for Distinguished Names and versions [43](#__RefHeading___Toc479148695)

C.1.2 General [43](#__RefHeading___Toc479148696)

C.2 Mapping [43](#__RefHeading___Toc479148697)

C.2.1 Operation and Notification mapping [43](#__RefHeading___Toc479148698)

C.2.2 Operation parameter mapping [44](#__RefHeading___Toc479148699)

C.2.2.1 Operation listSwMCapabilities [44](#__RefHeading___Toc479148700)

C.2.2.1.1 Input parameters [44](#__RefHeading___Toc479148701)

C.2.2.1.2 Output parameters [45](#__RefHeading___Toc479148702)

C.2.2.1.3 Fault definition [45](#__RefHeading___Toc479148703)

C.2.2.2 Operation listSwMProfiles [45](#__RefHeading___Toc479148704)

C.2.2.2.1 Input parameters [45](#__RefHeading___Toc479148705)

C.2.2.2.2 Output parameters [45](#__RefHeading___Toc479148706)

C.2.2.3 Operation createSwMProfile [45](#__RefHeading___Toc479148707)

C.2.2.3.1 Input parameters [45](#__RefHeading___Toc479148708)

C.2.2.3.2 Output parameters [46](#__RefHeading___Toc479148709)

C.2.2.4 Operation deleteSwMProfile [46](#__RefHeading___Toc479148710)

C.2.2.4.1 Input parameters [46](#__RefHeading___Toc479148711)

C.2.2.4.2 Output parameters [46](#__RefHeading___Toc479148712)

C.2.2.5 Operation listSwMProcesses [46](#__RefHeading___Toc479148713)

C.2.2.5.1 Input parameters [46](#__RefHeading___Toc479148714)

C.2.2.5.2 Output parameters [46](#__RefHeading___Toc479148715)

C.2.2.6 Operation resumeSwMProcess [47](#__RefHeading___Toc479148716)

C.2.2.6.1 Input parameters [47](#__RefHeading___Toc479148717)

C.2.2.6.2 Output parameters [47](#__RefHeading___Toc479148718)

C.2.2.7 Operation swFallback [47](#__RefHeading___Toc479148719)

C.2.2.7.1 Input parameters [47](#__RefHeading___Toc479148720)

C.2.2.7.2 Output parameters [47](#__RefHeading___Toc479148721)

C.2.2.8 Operation terminateSwMProcess [47](#__RefHeading___Toc479148722)

C.2.2.8.1 Input parameters [47](#__RefHeading___Toc479148723)

C.2.2.8.2 Output parameters [48](#__RefHeading___Toc479148724)

C.2.2.9 Operation changeSwMProfile [48](#__RefHeading___Toc479148725)

C.2.2.9.1 Input parameters [48](#__RefHeading___Toc479148726)

C.2.2.9.2 Output parameters [48](#__RefHeading___Toc479148727)

C.2.2.10 Operation downloadNESw [48](#__RefHeading___Toc479148728)

C.2.2.10.1 Input parameters [48](#__RefHeading___Toc479148729)

C.2.2.10.2 Output parameters [49](#__RefHeading___Toc479148730)

C.2.2.10.3 Fault definition [49](#__RefHeading___Toc479148731)

C.2.2.11 Operation activateNESw [49](#__RefHeading___Toc479148732)

C.2.2.11.1 Input parameters [49](#__RefHeading___Toc479148733)

C.2.2.11.2 Output parameters [49](#__RefHeading___Toc479148734)

C.2.2.11.3 Fault definition [49](#__RefHeading___Toc479148735)

C.2.2.12 Operation installNESw [50](#__RefHeading___Toc479148736)

C.2.2.12.1 Input parameters [50](#__RefHeading___Toc479148737)

C.2.2.12.2 Output parameters [50](#__RefHeading___Toc479148738)

C.2.2.12.3 Fault definition [50](#__RefHeading___Toc479148739)

C.2.2.13 Operation listNaswmProcesses [50](#__RefHeading___Toc479148740)

C.2.2.13.1 Input parameters [50](#__RefHeading___Toc479148741)

C.2.2.13.2 Output parameters [51](#__RefHeading___Toc479148742)

C.2.2.13.3 Fault definition [51](#__RefHeading___Toc479148743)

C.2.2.14 Operation cancelNaswmProcesses [51](#__RefHeading___Toc479148744)

C.2.2.14.1 Input parameters [51](#__RefHeading___Toc479148745)

C.2.2.14.2 Output parameters [51](#__RefHeading___Toc479148746)

C.2.2.14.3 Fault definition [51](#__RefHeading___Toc479148747)

C.3 Solution Set definitions [52](#__RefHeading___Toc479148748)

C.3.1 WSDL definition structure [52](#__RefHeading___Toc479148749)

C.3.2 Graphical Representation [52](#__RefHeading___Toc479148750)

C.3.3 WSDL specification “SWMIRPSystem.wsdl” [53](#__RefHeading___Toc479148751)

Annex D (informative): Change history [66](#__RefHeading___Toc479148752)

# Foreword

This Technical Specification 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:

32.531: Telecommunication management; Software management; Concepts and Integration Reference Point (IRP) Requirements

32.532: Telecommunication management; Software management Integration Reference Point (IRP); Information Service (IS)

**32.536: Telecommunication management; Software management Integration Reference Point (IRP); Solution Set (SS) definitions**

Software Management, in general, provides the operator with the ability to manage effectively the software entities residing in the 3G network. Software Management function is useful especially when there is a need to manage a large number of software residing in the managed elements widely distributed geographically. The main focus is the management of new software releases and correction patches [2].

The software management operations are initiated by the operator or by functions in the Operations Systems (OSs) either on single NE or across several NEs of the 3G network.

# 1 Scope

The present document specifies the Solution Set definitions for the IRP whose semantics are specified in Software Management IRP Information Service (3GPP TS 32.532 [5]).

This Solution Set definitions specification is related to 3GPP TS 32.532 V14.0.X [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, edition number, version number, etc.) or 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 TR 21.905: "Vocabulary for 3GPP Specifications".

[2] 3GPP TS 32.101: "Telecommunication management; Principles and high level requirements".

[3] 3GPP TS 32.102: "Telecommunication management; Architecture".

[4] 3GPP TS 32.531: "Telecommunication management; Software management; Concepts and Integration Reference Point (IRP) Requirements".

[5] 3GPP TS 32.532: "Telecommunication management; Software management Integration Reference Point (IRP); Information Service (IS)".

[6] OMG TC Document telecom/98-11-01: "OMG Notification Service". <http://www.omg.org/technology/documents/>

[7] 3GPP TS 32.336: “Telecommunication management; Notification Log (NL) Integration Reference Point (IRP): Solution Set (SS) definitions”.

[8] 3GPP TS 32.331: “Telecommunication management; Notification Log (NL) Integration Reference Point (IRP): Requirements”.

[9] 3GPP TS 32.150: "Telecommunication management; Integration Reference Point (IRP) Concept and definitions".

[10] 3GPP TS 32.306: "Telecommunication management; Configuration Management (CM); Notification Integration Reference Point (IRP): Solution Set (SS) definitions".

[11] W3C SOAP 1.1 specification (<http://www.w3.org/TR/2000/NOTE-SOAP-20000508/>).

[12] W3C XPath 1.0 specification (<http://www.w3.org/TR/1999/REC-xpath-19991116>)

[13] W3C WSDL 1.1 specification (<http://www.w3.org/TR/2001/NOTE-wsdl-20010315>)

[14] W3C SOAP 1.2 specification (<http://www.w3.org/TR/soap12-part1/>)

[15] 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 TS 32.101 [2], TS 32.102 [3], 3GPP TS 32.331 [8], 3GPP TS 32.150 [9] and TR 21.905 [1] and the following apply. A term defined in the present document takes precedence over the definition of the same term, if any, in TS 32.532 [5], TS 32.531 [4], 3GPP TS 32.331 [8], 3GPP TS 32.150 [9], TS 32.101 [2], TS 32.102 [3] and TS 21.905 [1], in that order.

## 3.2 Abbreviations

For the purposes of the present document, the abbreviations given in TR 21.905 [1], TS 32.531 [4], 3GPP TS 32.331 [8], 3GPP TS 32.150 [9] and the following apply. An abbreviation defined in the present document takes precedence over the definition of the same abbreviation, if any, in TS 32.532 [5], TS 32.531 [4], 3GPP TS 32.331 [8], 3GPP TS 32.150 [9], TS 32.101 [2], TS 32.102 [3] and TS 21.905 [1], in that order.

IRP Integration Reference Point

IS Information Service

NL Notification Log

NRM Network Resource Model

SwM Software Management

UML Unified Modelling Language

WSDL Web Service Description Language

XML eXtensible Markup Language

# 4 Solution Set definitions

This specification defines the following 3GPP Software management IRP Solution Set definitions:

- 3GPP Software management IRP CORBA SS (Annex A)

- 3GPP Software management IRP XML definitions (Annex B)

- 3GPP Software management IRP SOAP Solution Set (Annex C)

Annex A (normative):   
CORBA Solution Set

This annex contains the CORBA Solution Set for the IRP whose semantics is specified in Software management IRP: Information Service (TS 32.532 [5]).

# A.1 Architectural Features

The overall architectural feature of Software management IRP is specified in 3GPP TS 32.532 [5].

This clause specifies features that are specific to the CORBA SS.

## A.1.1 Syntax for Distinguished Names and Versions

The syntax of a Distinguished Name is defined in 3GPP TS 32.300 [15].

# A.2 Mapping

## A.2.1 Operation and Notification mapping

Software Management IRP: IS 3GPP TS (see 3GPP TS 32.532 [5]) 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  Software Management IRP: IS 3GPP TS 32.532[5] | SS Method | Qualifier |
| listSwMCapabilities | listSwMCapabilities | M |
| listSwMProfiles | listSwMProfiles | M |
| createSwMProfile | createSwMProfile | M |
| deleteSwMProfile | deleteSwMProfile | M |
| listSwMProcesses | listSwMProcesses | M |
| resumeSwMProcess | resumeSwMProcess | M |
| swFallback | swFallback | M |
| terminateSwMProcess | terminateSwMProcess | M |
| changeSwMProfile | changeSwMProfile | O |
| downloadNESw | downloadNESw | M |
| installNESw | installNESw | O |
| activateNESw | activateNESw | M |
| listNaswmProcesses | listNaswmProcesses | CM |
| cancelNaswmProcesses | cancelNaswmProcesses | CM |
| notifySwMProfileCreation | notifySwMProfileCreation | M |
| notifySwMProfileDeletion | notifySwMProfileDeletion | M |
| notifySwMProcessCreation | notifySwMProcessCreation | M |
| notifySwMProcessStage | notifySwMProcessStage | M |
| notifySwMProcessDeletion | notifySwMProcessDeletion | M |
| notifyNewSwAvailability | notifyNewSwAvailability | M |
| notifySwMProfileChange | notifySwMProfileChange | O |
| notifyDownloadNESwStatusChanged | notifyDownloadNESwStatusChanged | M |
| notifyInstallNESwStatusChanged | notifyInstallNESwStatusChanged | O |
| notifyActivateNESwStatusChanged | notifyActivateNESwStatusChanged | M |

## A.2.2 Operation parameter mapping

Reference 3GPP TS 32.532 [5] 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 listSwMCapabilities parameters to SS equivalents

|  |  |  |
| --- | --- | --- |
| IS Operation parameter | SS Method parameter | Qualifier |
| nEInformation | SwMIRPConstDefs::NEInformation | M |
| capabilitiesList | SwMIRPConstDefs::SwMCapabilitiesList | M |
| Result | Exceptions:  SwMIRPConstDefs::ListSwMCapabilities,  GenericIRPManagementSystem::ParameterNotSupported,  GenericIRPManagementSystem::InvalidParameter,  GenericIRPManagementSystem::ValueNotSupported,  GenericIRPManagementSystem::OperationNotSupported | M |

Table A.2.2.2: Mapping from IS listSwMProfiles parameters to SS equivalents

|  |  |  |
| --- | --- | --- |
| IS Operation parameter | SS Method parameter | Qualifier |
| nEInformation | SwMIRPConstDefs::NEInformation | M |
| profileList | SwMIRPConstDefs::SwMProfileList | M |
| result | Exceptions:  SwMIRPConstDefs::ListSwMProfile,  GenericIRPManagementSystem::ParameterNotSupported,  GenericIRPManagementSystem::InvalidParameter,  GenericIRPManagementSystem::ValueNotSupported,  GenericIRPManagementSystem::OperationNotSupported | M |

Table A.2.2.3: Mapping from IS createSwMProfile parameters to SS equivalents

|  |  |  |
| --- | --- | --- |
| IS Operation parameter | SS Method parameter | Qualifier |
| id | SwMIRPConstDefs::IdOpt | O |
| nEInformation | SwMIRPConstDefs::NEInformation | M |
| swVersionToBeInstalled | SwMIRPConstDefs::SwVersionToBeInstalledConditional | CM |
| stepsAndSelectedStopPointList | SwMIRPConstDefs::StepsAndSelectedStopPointList | M |
| selectedFinalAdministrativeState | SwMIRPConstDefs::SelectedFinalAdministrativeState | M |
| result | Exceptions:  SwMIRPConstDefs::CreateSwMProfile,  GenericIRPManagementSystem::ParameterNotSupported,  GenericIRPManagementSystem::InvalidParameter,  GenericIRPManagementSystem::ValueNotSupported,  GenericIRPManagementSystem::OperationNotSupported | M |

Table A.2.2.4: Mapping from IS deleteSwMProfile parameters to SS equivalents

|  |  |  |
| --- | --- | --- |
| IS Operation parameter | SS Method parameter | Qualifier |
| id | SwMIRPConstDefs::Id | M |
| result | Exceptions:  SwMIRPConstDefs::DeleteSwMProfile,  GenericIRPManagementSystem::ParameterNotSupported,  GenericIRPManagementSystem::InvalidParameter,  GenericIRPManagementSystem::ValueNotSupported,  GenericIRPManagementSystem::OperationNotSupported | M |

Table A.2.2.5: Mapping from IS listSwMProcesses parameters to SS equivalents

|  |  |  |
| --- | --- | --- |
| IS Operation parameter | SS Method parameter | Qualifier |
| nEIdentification | SwMIRPConstDefs::NEIdentificationOpt | O |
| processList | SwMIRPConstDefs::ProcessList | M |
| result | Exceptions:  SwMIRPConstDefs::ListSwMProcesses,  GenericIRPManagementSystem::ParameterNotSupported,  GenericIRPManagementSystem::InvalidParameter,  GenericIRPManagementSystem::ValueNotSupported,  GenericIRPManagementSystem::OperationNotSupported | M |

Table A.2.2.6: Mapping from IS resumeSwMProcess parameters to SS equivalents

|  |  |  |
| --- | --- | --- |
| IS Operation parameter | SS Method parameter | Qualifier |
| id | SwMIRPConstDefs::Id | M |
| startStepName | SwMIRPConstDefs::NameOfStep | M |
| result | Exceptions:  SwMIRPConstDefs::ResumeSwMProcess,  GenericIRPManagementSystem::ParameterNotSupported,  GenericIRPManagementSystem::InvalidParameter,  GenericIRPManagementSystem::ValueNotSupported,  GenericIRPManagementSystem::OperationNotSupported | M |

Table A.2.2.7: Mapping from IS swFallback parameters to SS equivalents

|  |  |  |
| --- | --- | --- |
| IS Operation parameter | SS Method parameter | Qualifier |
| filter | SwMIRPConstDefs::Filter | M |
| nEList | SwMIRPConstDefs::NEList | M |
| result | Exceptions:  SwMIRPConstDefs::SwFallback,  GenericIRPManagementSystem::ParameterNotSupported,  GenericIRPManagementSystem::InvalidParameter,  GenericIRPManagementSystem::ValueNotSupported,  GenericIRPManagementSystem::OperationNotSupported | M |

Table A.2.2.8: Mapping from IS terminateSwMProcess parameters to SS equivalents

|  |  |  |
| --- | --- | --- |
| IS Operation parameter | SS Method parameter | Qualifier |
| id | SwMIRPConstDefs::Id | M |
| result | Exceptions:  SwMIRPConstDefs::TerminateSwMProcess,  GenericIRPManagementSystem::ParameterNotSupported,  GenericIRPManagementSystem::InvalidParameter,  GenericIRPManagementSystem::ValueNotSupported,  GenericIRPManagementSystem::OperationNotSupported | M |

Table A.2.2.9: Mapping from IS changeSwMProfile parameters to SS equivalents

|  |  |  |
| --- | --- | --- |
| IS Operation parameter | SS Method parameter | Qualifier |
| id | SwMIRPConstDefs::Id | M |
| nEInformation | SwMIRPConstDefs::NEInformation | M |
| swVersionToBeInstalled | SwMIRPConstDefs::SwVersionToBeInstalledConditional | CM |
| stepsAndSelectedStopPointList | SwMIRPConstDefs::StepsAndSelectedStopPointList | M |
| selectedFinalAdministrativeState | SwMIRPConstDefs::SelectedFinalAdministrativeState | M |
| versionNumber | SwMIRPConstDefs::VersionNumber | M |
| conflictingProfileId | SwMIRPConstDefs::ConflictingProfileIdConditional | C |
| result | Exceptions:  SwMIRPConstDefs::CreateSwMProfile,  GenericIRPManagementSystem::ParameterNotSupported,  GenericIRPManagementSystem::InvalidParameter,  GenericIRPManagementSystem::ValueNotSupported,  GenericIRPManagementSystem::OperationNotSupported | M |

Table A.2.2.10: Mapping from IS downloadNESw parameters to SS equivalents

|  |  |  |
| --- | --- | --- |
| IS Operation parameter | SS Method parameter | Qualifier |
| swToBeDownloaded | SwMIRPConstDefs::SWToBeDownloaded | M |
| neIdentifier | SwMIRPConstDefs::NEIdentifier | M |
| downloadProcessId | SwMIRPConstDefs::RequestID | M |
| Reason | SwMIRPConstDefs::Reason | O |
| listOfStepNumbersAndDurations | SwMIRPConstDefs::ListOfStepNumbersAndDurations | CM \*) |
| result | Exceptions:  SwMIRPConstDefs:: OperationFailed,  SwMIRPConstDefs:: ResourceLimitation | M |

\*) Note: For the condition see TS 32.532 [5].

Table A.2.2.11: Mapping from IS installNESw parameters to SS equivalents

|  |  |  |
| --- | --- | --- |
| IS Operation parameter | SS Method parameter | Qualifier |
| swToBeInstalled | SwMIRPConstDefs:: SWToBeInstalled | M |
| neIdentifier | SwMIRPConstDefs::NEIdentifier | M |
| installProcessId | SwMIRPConstDefs::RequestID | M |
| reason | SwMIRPConstDefs::Reason | O |
| listOfStepNumbersAndDurations | SwMIRPConstDefs::ListOfStepNumbersAndDurations | CM \*) |
| result | Exceptions:  SwMIRPConstDefs:: OperationFailed,  SwMIRPConstDefs:: ResourceLimitation  SwMIRPConstDefs:: SWNotAvailable | M |

\*) Note: For the condition see TS 32.532 [5].

Table A.2.2.12: Mapping from IS activateNESw parameters to SS equivalents

|  |  |  |
| --- | --- | --- |
| IS Operation parameter | SS Method parameter | Qualifier |
| swVersionToBeActivated | SwMIRPConstDefs::SWVersion | M |
| neIdentifier | SwMIRPConstDefs::NEIdentifier | M |
| activateProcessId | SwMIRPConstDefs::RequestID | M |
| reason | SwMIRPConstDefs::Reason | O |
| listOfStepNumbersAndDurations | SwMIRPConstDefs::ListOfStepNumbersAndDurations | CM \*) |
| result | Exceptions:  SwMIRPConstDefs:: OperationFailed,  SwMIRPConstDefs:: ResourceLimitation | M |

\*) Note: For the condition see TS 32.532 [5].

Table A.2.2. 13: Mapping from IS listNaswmProcesses parameters to SS equivalents

|  |  |  |
| --- | --- | --- |
| IS Operation parameter | SS Method parameter | Qualifier |
| naswmId | SwMIRPConstDefs::Id | O |
| naswmOperationType | SwMIRPConstDefs::NaswmOperationType | M |
| naswmProcessList | SwMIRPConstDefs::NaswmProcessList | M |
| result | Exceptions:  SwMIRPConstDefs::OperationFailed,  SwMIRPConstDefs::ResourceLimitation | M |
| reason | SwMIRPConstDefs::Reason | O |

\*) Note: For the condition see TS 32.532 [5].

Table A.2.2.14: Mapping from IS cancelNaswmProcesses parameters to SS equivalents

|  |  |  |
| --- | --- | --- |
| IS Operation parameter | SS Method parameter | Qualifier |
| processId | SwMIRPConstDefs::Id | M |
| naswmOperationType | SwMIRPConstDefs::NaswmOperationType | M |
| result | Exceptions:  SwMIRPConstDefs::OperationFailed,  SwMIRPConstDefs::ResourceLimitation | M |
| reason | SwMIRPConstDefs::Reason | O |

## A.2.3 Notification parameter mapping

Reference 3GPP TS 32.532 [5] 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 [5] defined notification parameters.

Table A.2.3.1: Mapping for notifySwMProfileCreation

|  |  |  |  |
| --- | --- | --- | --- |
| IS Parameters | <SS> Parameters | Qualifier | Comment |
| id | SwMIRPConstDefs::Id | M |  |
| versionNumber | SwMIRPConstDefs::VersionNumber | M |  |
| nEInformation | SwMIRPConstDefs::NEInformation | M |  |
| swVersionToBeInstalled | SwMIRPConstDefs::SwVersionToBeInstalledConditional | CM |  |
| stepsAndSelectedStopPointList | SwMIRPConstDefs::StepsAndSelectedStopPointList | M |  |
| selectedFinalAdministrativeState | SwMIRPConstDefs::SelectedFinalAdministrativeState | M |  |

Table A.2.3.2: Mapping for notifySwMProfileDeletion

|  |  |  |  |
| --- | --- | --- | --- |
| IS Parameters | <SS> Parameters | Qualifier | Comment |
| id | SwMIRPConstDefs::Id | M |  |

Table A.2.3.3: Mapping for notifySwMProcessCreation

|  |  |  |  |
| --- | --- | --- | --- |
| IS Parameters | <SS> Parameters | Qualifier | Comment |
| id | SwMIRPConstDefs::Id | M |  |
| nEIdentification | SwMIRPConstDefs::NEIdentification | M |  |
| profileId | SwMIRPConstDefs::ProfileId | M |  |
| matchingNEInformation | SwMIRPConstDefs::MatchingNEInformation | M |  |
| stepInfoList | SwMIRPConstDefs::StepInfoList | M |  |

Table A.2.3.4: Mapping for notifySwMProcessStage

|  |  |  |  |
| --- | --- | --- | --- |
| IS Parameters | <SS> Parameters | Qualifier | Comment |
| id | SwMIRPConstDefs::Id | M |  |
| stepInfoList | SwMIRPConstDefs::StepInfoList | M |  |

Table A.2.3.5: Mapping for notifySwMProcessDeletion

|  |  |  |  |
| --- | --- | --- | --- |
| IS Parameters | <SS> Parameters | Qualifier | Comment |
| id | SwMIRPConstDefs::Id | M |  |
| triggerForDeletion | SwMIRPConstDefs::TriggerForDeletion | M |  |
| additionalInformation | SwMIRPConstDefs::AdditionalInformationOptional | O |  |

Table A.2.3.6: Mapping for notifyNewSwAvailability

|  |  |  |  |
| --- | --- | --- | --- |
| IS Parameters | <SS> Parameters | Qualifier | Comment |
| nEandSWversion | SwMIRPConstDefs::NEandSWversion | M |  |

Table A.2.3.7: Mapping for notifySwMProfileChange

|  |  |  |  |
| --- | --- | --- | --- |
| IS Parameters | <SS> Parameters | Qualifier | Comment |
| id | SwMIRPConstDefs::Id | M |  |
| versionNumber | SwMIRPConstDefs::VersionNumber | M |  |
| nEInformation | SwMIRPConstDefs::NEInformation | M |  |
| swVersionToBeInstalled | SwMIRPConstDefs::SwVersionToBeInstalledConditional | CM |  |
| stepsAndSelectedStopPointList | SwMIRPConstDefs::StepsAndSelectedStopPointList | M |  |
| selectedFinalAdministrativeState | SwMIRPConstDefs::SelectedFinalAdministrativeState | M |  |

Table A.2.3.8: Mapping for notifyDownloadNESwStatusChanged

|  |  |  |  |
| --- | --- | --- | --- |
| IS Parameters | <SS> Parameters | Qualifier | Comment |
| downloadProcessId | SwMIRPConstDefs::DownloadProcessId; | M |  |
| downloadOperationStatus | SwMIRPConstDefs::DownloadNESwOperationStatus | M |  |
| downloadedNESwInfo | SwMIRPConstDefs::DownloadedNESwInfo | O |  |
| failedSwInfo | SwMIRPConstDefs::FailedSwInfo | O |  |

Table A.2.3.9: Mapping for notifyInstallNESwStatusChanged

|  |  |  |  |
| --- | --- | --- | --- |
| IS Parameters | <SS> Parameters | Qualifier | Comment |
| installProcessId | SwMIRPConstDefs::RequestID | M |  |
| installOperationStatus | SwMIRPConstDefs::InstallOperationStatus | M |  |
| installNESwInfo | SwMIRPConstDefs::InstalledNESwInfo | O |  |
| failedSwInfo | SwMIRPConstDefs::FailedSwInfo | O |  |

Table A.2.3.10: Mapping for notifyActivateNESwStatusChanged

|  |  |  |  |
| --- | --- | --- | --- |
| IS Parameters | <SS> Parameters | Qualifier | Comment |
| activateProcessId | SwMIRPConstDefs::RequestID | M |  |
| activateOperationStatus | SwMIRPConstDefs::ActivateOperationStatus | M |  |
| swVersion | SwMIRPConstDefs::SWVersion | M |  |
| failureReason | SwMIRPConstDefs::FailureReason | CM |  |

# A.3 Solution Set definitions

## A.3.1 IDL definition structure

Clause A.3.2 defines the constants and types used by the Software management IRP.

Clause A.3.3 defines the operations which are performed by the Software management IRP agent.

Clause A.3.4 defines the notifications which are performed by the Software management IRP agent.

## A.3.2 IDL specification “SwMIRPConstDefs.idl”

// File: SwMIRPConstDefs.idl

#ifndef \_SWM\_IRP\_CONST\_DEFS\_IDL\_

#define \_SWM\_IRP\_CONST\_DEFS\_IDL\_

#include <NotificationIRPConstDefs.idl>

#include <GenericIRPManagementConstDefs.idl>

// This statement must appear after all include statements

#pragma prefix "3gppsa5.org"

/\* ## Module: SwMIRPConstDefs \*/

module SwMIRPConstDefs

{

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* definition of types used in operations for Software Management : \*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* types used in several operations: \*/

enum Result { SUCCESS, PARTLY\_SUCCESSFUL, FAILURE, NE\_INFORMATION\_INTERSECTION, STEPNAME\_DOES\_NOT\_MATCH };

enum ResultChangeOrCreateProfileOperation { SUCCESS, FAILURE, NE\_INFORMATION\_INTERSECTION, NOT\_ALLOWED\_BECAUSE\_OF\_ONGOING\_ACTIVITY };

enum ResultNaSwmNotification { REQUEST\_ACCEPTED, REQUEST\_FAILED, NOT\_ALLOWED\_BECAUSE\_OF\_ONGOING\_SWM\_ACTIVITY };

enum ResultListNaswmProcesses { SUCCESS, FAILURE};

enum ResultCancelNaswmProcesses { SUCCESS, OPERATION\_IS\_ALREADY\_COMPLETED, NO\_SUCH\_PROCESS, FAILURE };

typedef GenericIRPManagementConstDefs::DN Id;

/\*

IdOpt is a type carrying an optional parameter.

If the boolean is TRUE, then the value is present.

Otherwise the value is absent.

\*/

union IdOpt switch (boolean)

{

case TRUE: GenericIRPManagementConstDefs::DN value;

};

/\*

ConflictingProfileIdConditional is a type carrying a conditional parameter.

The boolean shall be TRUE, if the condition described in TS 32.532 or 32.502 is fulfilled.

In this case the value is present. Otherwise the value is absent.

\*/

union ConflictingProfileIdConditional switch (boolean)

{

case TRUE: Id value;

};

typedef string NEInformation;

typedef NEInformation MatchingNEInformation;

typedef string AdditionalInformation;

/\*

AdditionalInformationOpt is a type carrying a optional parameter.

The boolean shall be TRUE, if the condition described in TS 32.532 is fulfilled.

In this case the value is present. Otherwise the value is absent.

\*/

union AdditionalInformationOpt switch (boolean)

{

case TRUE: AdditionalInformation value;

};

typedef GenericIRPManagementConstDefs::DN NEIdentification;

/\*

NEIdentificationOpt is a type carrying an optional parameter.

If the boolean is TRUE, then the value is present.

Otherwise the value is absent.

\*/

union NEIdentificationOpt switch (boolean)

{

case TRUE: NEIdentification value;

};

enum SwFallbackStatus { FALLBACK\_SUCCESSFUL, FALLBACK\_UNSUCCESSFUL };

struct NeListEntry

{

SwMIRPConstDefs::NEIdentification nEIdentification;

SwMIRPConstDefs::SwFallbackStatus swFallbackStatus;

};

typedef sequence<NeListEntry> NeList;

typedef string SwVersionToBeInstalled;

/\*

SwVersionToBeInstalledConditional is a type carrying a conditional parameter.

The boolean shall be TRUE, if the condition described in TS 32.532 is fulfilled.

In this case the value is present. Otherwise the value may be absent.

\*/

union SwVersionToBeInstalledConditional switch (boolean)

{

case TRUE: SwVersionToBeInstalled value;

};

enum NameOfStep

{

NE\_HEALTH\_CHECK,

SW\_DOWNLOAD,

SW\_INSTALLATION,

SW\_ACTIVATION,

PREPARE\_BASIC\_CONFIGURATION\_AND\_OAMLINK,

RETRIEVE\_CONFIGURATION\_DATA,

SETUP\_PRECONFIGURED\_SIGNALLING\_LINKS,

SET\_FINAL\_STATE\_OF\_NE

};

/\*

The following values are not used in SWM IRP, but only in inheriting Self-Conf IRP TS 32.502:

PREPARE\_BASIC\_CONFIGURATION\_AND\_OAMLINK,

RETRIEVE\_CONFIGURATION\_DATA,

SETUP\_PRECONFIGURED\_SIGNALLING\_LINKS,

SET\_FINAL\_STATE\_OF\_NE

\*/

typedef unsigned short SequenceNumberInProcess;

enum StopPointCanBeSetBeforeThisStep { YES, NO };

struct StepsAndOfferedStopPointListEntry

{

SwMIRPConstDefs::NameOfStep nameOfStep;

SwMIRPConstDefs::SequenceNumberInProcess sequenceNumberInProcess;

SwMIRPConstDefs::StopPointCanBeSetBeforeThisStep stopPointCanBeSetBeforeThisStep

;

};

typedef sequence<StepsAndOfferedStopPointListEntry> StepsAndOfferedStopPointList;

enum StopPointSetIndication { STOP\_POINT\_IS\_SET\_BEFORE\_THIS\_STEP, STOP\_POINT\_IS\_NOT\_SET };

struct StepAndSelectedStopPointListEntry

{

SwMIRPConstDefs::NameOfStep nameOfStep;

SwMIRPConstDefs::SequenceNumberInProcess sequenceNumberInProcess;

SwMIRPConstDefs::StopPointSetIndication stopPointSetIndication

;

};

typedef sequence<StepAndSelectedStopPointListEntry> StepsAndSelectedStopPointList;

enum StepProgress { NOT\_YET\_STARTED, RUNNING, COMPLETED, AWAITING\_RESUME, FAILURE, TERMINATED };

struct StepInfoListEntry

{

SwMIRPConstDefs::NameOfStep nameOfStep;

SwMIRPConstDefs::SequenceNumberInProcess sequenceNumberInProcess;

SwMIRPConstDefs::StopPointSetIndication stopPointSetIndication;

SwMIRPConstDefs::StepProgress stepProgress;

};

typedef sequence<StepInfoListEntry> StepInfoList;

struct SwMProcessListEntry

{

SwMIRPConstDefs::Id id;

SwMIRPConstDefs::NEIdentification nEIdentification;

SwMIRPConstDefs::StepInfoList stepInfoList;

};

typedef sequence<SwMProcessListEntry> SwMProcessList;

enum FinalAdministrativeStateValue { LOCKED, UNLOCKED, DETERMINED\_BY\_CONFIGURATION\_DATA };

typedef FinalAdministrativeStateValue OfferedFinalAdministrativeStateValue;

typedef sequence<OfferedFinalAdministrativeStateValue> OfferedFinalAdministrativeStateInformation;

typedef FinalAdministrativeStateValue SelectedFinalAdministrativeStateValue;

typedef sequence<SwVersionToBeInstalled> SwVersionToBeInstalledOfferList;

/\*

SwVersionToBeInstalledOfferListConditional is a type carrying a conditional parameter.

The boolean shall be TRUE, if the condition described in TS 32.532 is fulfilled.

In this case the value is present. Otherwise the value may be absent.

\*/

union SwVersionToBeInstalledOfferListConditional switch (boolean)

{

case TRUE: SwVersionToBeInstalledOfferList value;

};

typedef unsigned short VersionNumber;

struct ProfileId

{

SwMIRPConstDefs::Id id;

SwMIRPConstDefs::VersionNumber versionNumber;

};

struct SwMCapability

{

SwMIRPConstDefs::Id id;

SwMIRPConstDefs::NEInformation nEInformation;

SwMIRPConstDefs::StepsAndOfferedStopPointList stepsAndOfferedStopPointList;

SwMIRPConstDefs::OfferedFinalAdministrativeStateInformation offeredFinalAdministrativeStateInformation;

SwMIRPConstDefs::SwVersionToBeInstalledOfferListConditional swVersionToBeInstalledOfferList;

};

typedef sequence<SwMCapability> SwMCapabilitiesList;

struct SwMProfile

{

SwMIRPConstDefs::Id id;

SwMIRPConstDefs::VersionNumber versionNumber;

SwMIRPConstDefs::NEInformation nEInformation;

SwMIRPConstDefs::StepsAndSelectedStopPointList stepsAndSelectedStopPointList;

SwMIRPConstDefs::SelectedFinalAdministrativeStateValue selectedFinalAdministrativeState;

SwMIRPConstDefs::SwVersionToBeInstalledConditional swVersionToBeInstalled;

};

typedef sequence<SwMProfile> SwMProfilesList;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

/\* definition of types in notifications for software management : \*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

typedef string Filter;

typedef string NeAndSWVersion;

enum TriggerForDeletion { IRP\_AGENT\_TERMINATION, IRP\_MANAGER\_TERMINATION, AUTOMATED\_SWM\_SUCCESFULLY\_CONCLUDED, SELF\_CONFIGURATION\_SUCCESFULLY\_CONCLUDED };

/\*

The following values are not used in SWM IRP, but only in inheriting Self-Conf IRP TS 32.502:

SELF\_CONFIGURATION\_SUCCESFULLY\_CONCLUDED

\*/

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Definitions for Non-Automated Software Management

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

typedef GenericIRPManagementConstDefs::DN NEIdentifier;

typedef unsigned long RequestID;

typedef string Reason;

/\*

ReasonOpt is a type carrying an optional parameter.

If the boolean is TRUE, then the value is present.

Otherwise the value is absent.

\*/

union ReasonOpt switch (boolean)

{

case TRUE: Reason value;

};

typedef string FileLocation; //The FileLocation may be a directory path or a URL

typedef unsigned long FileSize; //the unit is byte

typedef string FileCompression;

typedef string FileFormat;

struct SWInfo

{

SwMIRPConstDefs::FileLocation swLocation;

SwMIRPConstDefs::FileSize swFileSize;

SwMIRPConstDefs::FileCompression swFileCompression;

SwMIRPConstDefs::FileFormat swFileFormat;

};

typedef SwMIRPConstDefs::FileLocation SWToBeInstalled;

typedef sequence<SWInfo> SWToBeDownloaded;

typedef string SWVersion;

typedef unsigned short Hours;

typedef unsigned short Minutes;

typedef unsigned short Seconds;

struct HoursMinutesSeconds

{

SwMIRPConstDefs::Hours hours;

SwMIRPConstDefs::Minutes minutes;

SwMIRPConstDefs::Seconds seconds;

};

/\*

HoursMinutesSecondsOpt is a type carrying an optional parameter.

If the boolean is TRUE, then the value is present.

Otherwise the value is absent.

\*/

union HoursMinutesSecondsOpt switch (boolean)

{

case TRUE: GenericIRPManagementConstDefs::HoursMinutesSeconds value;

};

typedef unsigned short ProcessStepNumber;

/\*

ProcessStepNumberOpt is a type carrying an optional parameter.

If the boolean is TRUE, then the value is present.

Otherwise the value is absent.

\*/

union ProcessStepNumberOpt switch (boolean)

{

case TRUE: GenericIRPManagementConstDefs::ProcessStepNumber value;

};

struct StepNumberAndDuration

{

SwMIRPConstDefs::ProcessStepNumber processStepNumber;

SwMIRPConstDefs::HoursMinutesSeconds estimatedDuration;

};

typedef sequence<StepNumberAndDuration> ListOfStepNumbersAndDurations;

/\*

ListOfStepNumbersAndDurationsOpt is a type carrying an optional parameter.

If the boolean is TRUE, then the value is present.

Otherwise the value is absent.

\*/

union ListOfStepNumbersAndDurationsOpt switch (boolean)

{

case TRUE: GenericIRPManagementConstDefs::ProcessStepNumber value;

};

enumNaswm OperationType { operationDownloadNESw, operationActivateNESw, operationInstallNESw };

struct NaswmProcessInfo

{

SwMIRPConstDefs::Id naswmId;

SwMIRPConstDefs::NEIdentification nEIdentification;

SwMIRPConstDefs::NaswmOperationType naswmOperationType;

SwMIRPConstDefs::HoursMinutesSeconds estimatedRemainingCompletionTimeForTheOperation;

SwMIRPConstDefs::ListOfStepNumbersAndDurationsOpt listOfStepNumbersAndDurations;

SwMIRPConstDefs::ProcessStepNumberOpt numberOfCurrentProcessStep;

SwMIRPConstDefs::HoursMinutesSecondsOpt estimatedRemainingCompletionTimeForTheCurrentStep;

};

typedef sequence<NaswmProcessInfo> naswmProcessList;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\* Definitions for Non-Automatic Software Management Notifications

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

typedef RequestID DownloadProcessId;

typedef RequestID InstallProcessId;

typedef RequestID ActivateProcessId;

typedef string FailureReason;

typedef sequence<FileLocation> DownloadedNESwInfo;

typedef sequence<FileLocation> InstalledNESwInfo;

struct FailedSwEntry

{

SwMIRPConstDefs::FileLocation failedSw;

SwMIRPConstDefs::FailureReason failureReason;

};

typedef sequence<FailedSwEntry> FailedSwInfo;

typedef SWVersion swVersionActivated;

enum DownloadNESwOperationStatus {NE\_SW\_DOWNLOAD\_SUCCESSFUL, NE\_SW\_DOWNLOAD\_FAILED, NE\_SW\_DOWNLOAD\_PARTIALLY\_SUCCESSFUL};

enum InstallOperationStatus {NE\_SW\_INSTALLATION\_SUCCESSFUL, NE\_SW\_INSTALLATION\_FAILED, NE\_SW\_INSTALLATION\_ PARTIALLY\_SUCCESSFUL};

enum ActivateOperationStatus {NE\_SW\_ACTIVATION\_SUCCESSFUL, NE\_SW\_ACTIVATION\_FAILED, NE\_SWACTIVATION\_PARTIALLY\_SUCCESSFUL};

interface AttributeNameValue

{

const string ID = "ID";

const string VERSION\_NUMBER = "VERSION\_NUMBER";

const string NE\_INFORMATION = "NE\_INFORMATION";

const string SW\_VERSION\_TO\_BE\_INSTALLED = "SW\_VERSION\_TO\_BE\_INSTALLED";

const string STEPS\_AND\_SELECTED\_STOP\_POINT\_LIST = "STEPS\_AND\_SELECTED\_STOP\_POINT\_LIST";

const string SELECTED\_FINAL\_ADMINISTRATIVE\_STATE = "SELECTED\_FINAL\_ADMINISTRATIVE\_STATE";

const string NE\_IDENTIFICATION = "NE\_IDENTIFICATION";

const string PROFILE\_ID = "PROFILE\_ID";

const string MATCHING\_NE\_INFORMATION = "MATCHING\_NE\_INFORMATION";

const string STEP\_INFO\_LIST = "STEP\_INFO\_LIST";

const string NE\_AND\_SW\_VERSION = "NE\_AND\_SW\_VERSION";

const string TRIGGER\_FOR\_DELETION = "TRIGGER\_FOR\_DELETION";

const string ADDITIONAL\_INFORMATION = "ADDITIONAL\_INFORMATION";

const string DOWNLOAD\_PROCESS\_ID = "DOWNLOAD\_PROCESS\_ID";

const string DOWNLOAD\_OPERATION\_STATUS = "DOWNLOAD\_OPERATION\_STATUS";

const string DOWNLOADED\_NESW\_INFO = "DOWNLOADED\_NESW\_INFO";

const string INSTALL\_PROCESS\_ID = "INSTALL\_PROCESS\_ID";

const string INSTALL\_OPERATION\_STATUS = "INSTALL\_OPERATION\_STATUS";

const string INSTALLED\_NESW\_INFO = "INSTALLED\_NESW\_INFO";

const string ACTIVATE\_PROCESS\_ID = "ACTIVATE\_PROCESS\_ID";

const string ACTIVATE\_OPERATION\_STATUS = "ACTIVATE\_OPERATION\_STATUS";

const string FAILURE\_REASON = " FAILURE\_REASON";

const string SW\_VERSION = "SW\_VERSION";

const string FAILED\_SW\_INFO = "FAILED\_SW\_INFO"; };

const string NASWM\_ID = "NASWM\_ID";

const string NUMBER\_OF\_CURRENT\_PROCESS\_STEP = "NUMBER\_OF\_CURRENT\_PROCESS\_STEP";

const string LIST\_OF\_STEP\_NUMBERS\_AND\_DURATIONS = "LIST\_OF\_STEP\_NUMBERS\_AND\_DURATIONS";

const string ESTIMATED\_REMAINING\_COMPLETION\_TIME\_FOR\_THE\_OPERATION= " ESTIMATED\_REMAINING\_COMPLETION\_TIME\_FOR\_THE\_OPERATION";

const string ESTIMATED\_REMAINING\_COMPLETION\_TIME\_FOR\_THE\_CURRENT\_STEP = "ESTIMATED\_REMAINING\_COMPLETION\_TIME\_FOR\_THE\_CURRENT\_STEP";

const string NASWM\_ OPERATION\_TYPE = " NASWM\_ OPERATION\_TYPE";

};

#endif // \_SWM\_IRP\_CONST\_DEFS\_IDL\_

## A.3.3 IDL specification “SwMIRPSystem.idl”

//File: SwMIRPSystem.idl

#ifndef \_SWM\_IRP\_SYSTEM\_IDL\_

#define \_SWM\_IRP\_SYSTEM\_IDL\_

#include <SwMIRPConstDefs.idl>

#include <GenericIRPManagementSystem.idl>

// This statement must appear after all include statements

#pragma prefix "3gppsa5.org"

/\* ## Module: SwMIRPSystem \*/

module SwMIRPSystem

{

/\*

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 ListSwMCapabilities { string reason; };

exception ListSwMProfiles { string reason; };

exception CreateSwMProfile { string reason; };

exception DeleteSwMProfile { string reason; };

exception ListSwMProcesses { string reason; };

exception ResumeSwMProcess { string reason; };

exception SwFallback { string reason; };

exception TerminateSwMProcess { string reason; };

exception ChangeSwMProfile { string reason; };

exception OperationFailed { string reason; };

exception ResourceLimitation { string reason; };

exception SWNotAvailable { string reason; };

exception ListNaswmProcesses{ string reason; };

exception CancelNaswmProcesses { string reason; };

interface SwMIRPOperations\_1

{

/\* for the purpose of this operation see 3GPP TS 32.532 \*/

SwMIRPConstDefs::Result listSwMCapabilities

(

in SwMIRPConstDefs::NEInformation nEInformation,

out SwMIRPConstDefs::SwMCapabilitiesList capabilitiesList

)

raises

(

ListSwMCapabilities,

GenericIRPManagementSystem::ParameterNotSupported,

GenericIRPManagementSystem::InvalidParameter,

GenericIRPManagementSystem::ValueNotSupported,

GenericIRPManagementSystem::OperationNotSupported

);

/\* for the purpose of this operation see 3GPP TS 32.532 \*/

SwMIRPConstDefs::Result listSwMProfiles

(

in SwMIRPConstDefs::NEInformation nEInformation,

out SwMIRPConstDefs::SwMProfileList profileList

)

raises

(

ListSwMProfiles,

GenericIRPManagementSystem::ParameterNotSupported,

GenericIRPManagementSystem::InvalidParameter,

GenericIRPManagementSystem::ValueNotSupported,

GenericIRPManagementSystem::OperationNotSupported

);

/\* for the purpose of this operation see 3GPP TS 32.532 \*/

SwMIRPConstDefs::ResultChangeOrCreateProfileOperation createSWMProfile

(

in SwMIRPConstDefs::IdOpt id,

in SwMIRPConstDefs::NEInformation nEInformation,

in SwMIRPConstDefs::SwVersionToBeInstalledConditional swVersionToBeInstalled,

in SwMIRPConstDefs::StepsAndSelectedStopPointList stepsAndSelectedStopPointList,

in SwMIRPConstDefs::SelectedFinalAdministrativeState selectedFinalAdministrativeState

)

raises

(

CreateSWMProfile,

GenericIRPManagementSystem::ParameterNotSupported,

GenericIRPManagementSystem::InvalidParameter,

GenericIRPManagementSystem::ValueNotSupported,

GenericIRPManagementSystem::OperationNotSupported

);

/\* for the purpose of this operation see 3GPP TS 32.532 \*/

SwMIRPConstDefs::Result deleteSWMProfile

(

in SwMIRPConstDefs::Id id

)

raises

(

DeleteSWMProfile,

GenericIRPManagementSystem::ParameterNotSupported,

GenericIRPManagementSystem::InvalidParameter,

GenericIRPManagementSystem::ValueNotSupported,

GenericIRPManagementSystem::OperationNotSupported

);

/\* for the purpose of this operation see 3GPP TS 32.532 \*/

SwMIRPConstDefs::Result listSwMProcesses

(

in SwMIRPConstDefs::NEIdentificationOpt nEIdentification,

out SwMIRPConstDefs::ProcessList processList

)

raises

(

ListSwMProcesses,

GenericIRPManagementSystem::ParameterNotSupported,

GenericIRPManagementSystem::InvalidParameter,

GenericIRPManagementSystem::ValueNotSupported,

GenericIRPManagementSystem::OperationNotSupported

);

/\* for the purpose of this operation see 3GPP TS 32.532 \*/

SwMIRPConstDefs::Result resumeSwMProcess

(

in SwMIRPConstDefs::Id id,

in SwMIRPConstDefs::NameOfStep startStepName

)

raises

(

ResumeSwMProcess,

GenericIRPManagementSystem::ParameterNotSupported,

GenericIRPManagementSystem::InvalidParameter,

GenericIRPManagementSystem::ValueNotSupported,

GenericIRPManagementSystem::OperationNotSupported

);

/\* for the purpose of this operation see 3GPP TS 32.532 \*/

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 32.532 \*/

SwMIRPConstDefs::Result terminateSwMProcess

(

in SwMIRPConstDefs::Id id

)

raises

(

TerminateSwMProcess,

GenericIRPManagementSystem::ParameterNotSupported,

GenericIRPManagementSystem::InvalidParameter,

GenericIRPManagementSystem::ValueNotSupported,

GenericIRPManagementSystem::OperationNotSupported

);

};

interface SwMIRPOperations\_2

{

/\* for the purpose of this operation see 3GPP TS 32.532 \*/

SwMIRPConstDefs::ResultChangeOrCreateProfileOperation changeSWMProfile

(

in SwMIRPConstDefs::Id id,

in SwMIRPConstDefs::NEInformation nEInformation,

in SwMIRPConstDefs::SwVersionToBeInstalledConditional swVersionToBeInstalled,

in SwMIRPConstDefs::StepsAndSelectedStopPointList stepsAndSelectedStopPointList,

in SwMIRPConstDefs::SelectedFinalAdministrativeState selectedFinalAdministrativeState

out SwMIRPConstDefs::VersionNumber versionNumber

out SwMIRPConstDefs::ConflictingProfileIdConditional conflictingProfileId

)

raises

(

ChangeSWMProfile,

GenericIRPManagementSystem::ParameterNotSupported,

GenericIRPManagementSystem::InvalidParameter,

GenericIRPManagementSystem::ValueNotSupported,

GenericIRPManagementSystem::OperationNotSupported

);

};

interface SwMIRPOperations\_3

{

/\* for the purpose of this operation see 3GPP TS 32.532 \*/

SwMIRPConstDefs::ResultNaSwmNotification downloadNESw

(

in SwMIRPConstDefs::SWToBeDownloaded swToBeDownloaded,

in SwMIRPConstDefs::NEIdentifier neIdentifier,

out SwMIRPConstDefs::RequestID downloadProcessId,

out SwMIRPConstDefs::Reason reason

)

raises

(

OperationFailed,

ResourceLimitation

);

/\* for the purpose of this operation see 3GPP TS 32.532 \*/

SwMIRPConstDefs::ResultNaSwmNotification activateNESw

(

in SwMIRPConstDefs::SWVersion swVersionToBeActivated,

in SwMIRPConstDefs::NEIdentifier neIdentifier,

out SwMIRPConstDefs::RequestID requestID,

out SwMIRPConstDefs::Reason reason

)

raises

(

OperationFailed,

ResourceLimitation

);

};

interface SwMIRPOperations\_4

{

/\* for the purpose of this operation see 3GPP TS 32.532 \*/

SwMIRPConstDefs::ResultNaSwmNotification installNESw

(

in SwMIRPConstDefs::SWToBeInstalled swToBeInstalled,

in SwMIRPConstDefs::NEIdentifiert neIdentifier,

out SwMIRPConstDefs::RequestID installProcessId,

out SwMIRPConstDefs::Reason reason

)

raises

(

OperationFailed,

ResourceLimitation,

SWNotAvailable

);

};

interface SwMIRPOperations\_5

{

/\* for the purpose of this operation see 3GPP TS 32.532 \*/

SwMIRPConstDefs::ResultListNaswmProcesses listNaswmProcesses

(

in SwMIRPConstDefs::IdOpt naswmId,

out SwMIRPConstDefs::NaswmProcessList naswmProcessList,

out SwMIRPConstDefs::ReasonOpt reason

)

raises

(

OperationFailed,

ResourceLimitation,

);

};

interface SwMIRPOperations\_6

{

/\* for the purpose of this operation see 3GPP TS 32.532 \*/

SwMIRPConstDefs::ResultCancelNaswmProcesses cancel NaswmProcesses

(

in SwMIRPConstDefs::Id p rocessId,

in SwMIRPConstDefs::NaswmOperationType naswmOperationType,

out SwMIRPConstDefs::ReasonOpt reason

)

raises

(

OperationFailed,

ResourceLimitation,

);

};

};

#endif // \_SWM\_IRP\_SYSTEM\_IDL\_

## A.3.4 IDL specification “SwMIRPNotifications.idl”

//File: SwMIRPNotifications.idl

#ifndef \_SWM\_IRP\_NOTIFICATIONS\_IDL\_

#define \_SWM\_IRP\_NOTIFICATIONS\_IDL\_

#include <SwMIRPConstDefs.idl>

#include <NotificationIRPNotifications.idl>

// This statement must appear after all include statements

#pragma prefix "3gppsa5.org"

/\* ## Module: SwMIRPNotifications

This contains the specification of notifications of Software Management.

==========================================================================

\*/

module SwMIRPNotifications

{

/\* Constant definitions for the notifySwMProfileCreation notification \*/

interface NotifySwMProfileCreation: NotificationIRPNotifications::Notify

{

const string EVENT\_TYPE = "notifySwMProfileCreation";

/\*\*

\* 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 =

SwMIRPConstDefs::AttributeNameValue::ID;

/\*\*

\* This constant defines the name of the VersionNumber property,

\* which is transported in the filterable\_body\_fields.

\* The data type for the value of this property is

\* SwMIRPConstDefs::VersionNumber.

\*/

const string VERSION\_NUMBER =

SwMIRPConstDefs::AttributeNameValue::VERSION\_NUMBER;

/\*\*

\* 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 NE\_INFORMATION =

SwMIRPConstDefs::AttributeNameValue::NE\_INFORMATION;

/\*\*

\* This constant defines the name of the SwVersionToBeInstalled property,

\* which is transported in the filterable\_body\_fields.

\* The data type for the value of this property is

\* SwMIRPConstDefs::SwVersionToBeInstalledConditional.

\*/

const string SW\_VERSION\_TO\_BE\_INSTALLED =

SwMIRPConstDefs::AttributeNameValue::SW\_VERSION\_TO\_BE\_INSTALLED;

/\*\*

\* This constant defines the name of the StepsAndSelectedStopPointList property,

\* which is transported in the remaining\_body.

\* The data type for the value of this property is

\* SwMIRPConstDefs::StepsAndSelectedStopPointList.

\*/

const string STEPS\_AND\_SELECTED\_STOP\_POINT\_LIST =

SwMIRPConstDefs::AttributeNameValue::STEPS\_AND\_SELECTED\_STOP\_POINT\_LIST;

/\*\*

\* This constant defines the name of the SelectedFinalAdministrativeState property,

\* which is transported in the remaining\_body.

\* The data type for the value of this property is

\* SwMIRPConstDefs::SelectedFinalAdministrativeState.

\*/

const string SELECTED\_FINAL\_ADMINISTRATIVE\_STATE =

SwMIRPConstDefs::AttributeNameValue::SELECTED\_FINAL\_ADMINISTRATIVE\_STATE;

};

/\* Constant definitions for the notifySwMProfileDeletion notification \*/

interface NotifySwMProfileDeletion: NotificationIRPNotifications::Notify

{

const string EVENT\_TYPE = "notifySwMProfileDeletion";

/\*\*

\* 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 =

SwMIRPConstDefs::AttributeNameValue::ID;

};

/\* Constant definitions for the notifySwMProcessCreation notification \*/

interface NotifySwMProcessCreation: NotificationIRPNotifications::Notify

{

const string EVENT\_TYPE = "notifySwMProcessCreation";

/\*\*

\* 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 =

SwMIRPConstDefs::AttributeNameValue::ID;

/\*\*

\* This constant defines the name of the NEIdentification property,

\* which is transported in the filterable\_body\_fields.

\* The data type for the value of this property is

\* SwMIRPConstDefs::NEIdentification.

\*/

const string NE\_IDENTIFICATION =

SwMIRPConstDefs::AttributeNameValue::NE\_IDENTIFICATION;

/\*\*

\* This constant defines the name of the ProfileId property,

\* which is transported in the remaining\_body.

\* The data type for the value of this property is

\* SwMIRPConstDefs::ProfileId.

\*/

const string PROFILE\_ID =

SwMIRPConstDefs::AttributeNameValue::PROFILE\_ID;

/\*\*

\* This constant defines the name of the MatchingNEInformation property,

\* which is transported in the remaining\_body.

\* The data type for the value of this property is

\* SwMIRPConstDefs::MatchingNEInformation.

\*/

const string MATCHING\_NE\_INFORMATION =

SwMIRPConstDefs::AttributeNameValue::MATCHING\_NE\_INFORMATION;

/\*\*

\* This constant defines the name of the StepInfoList property,

\* which is transported in the remaining\_body.

\* The data type for the value of this property is

\* SwMIRPConstDefs::StepInfoList.

\*/

const string STEP\_INFO\_LIST =

SwMIRPConstDefs::AttributeNameValue::STEP\_INFO\_LIST;

};

/\* Constant definitions for the notifySwMProcessStage notification \*/

interface NotifySwMProcessStage: NotificationIRPNotifications::Notify

{

const string EVENT\_TYPE = "notifySwMProcessStage";

/\*\*

\* 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 =

SwMIRPConstDefs::AttributeNameValue::ID;

/\*\*

\* This constant defines the name of the StepInfoList property,

\* which is transported in the remaining\_body.

\* The data type for the value of this property is

\* SwMIRPConstDefs::StepInfoList.

\*/

const string STEP\_INFO\_LIST =

SwMIRPConstDefs::AttributeNameValue::STEP\_INFO\_LIST;

};

/\* Constant definitions for the notifySwMProcessDeletion notification \*/

interface NotifySwMProcessDeletion: NotificationIRPNotifications::Notify

{

const string EVENT\_TYPE = "notifySwMProcessDeletion";

/\*\*

\* 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 =

SwMIRPConstDefs::AttributeNameValue::ID;

/\*\*

\* This constant defines the name of the TriggerForDeletion property,

\* which is transported in the remaining\_body.

\* The data type for the value of this property is

\* SwMIRPConstDefs::TriggerForDeletion.

\*/

const string ID =

SwMIRPConstDefs::AttributeNameValue::TRIGGER\_FOR\_DELETION;

/\*\*

\* This constant defines the name of the AdditionalInformation property,

\* which is transported in the remaining\_body.

\* The data type for the value of this property is

\* SwMIRPConstDefs::AdditionalInformationOptional.

\*/

const string ID =

SwMIRPConstDefs::AttributeNameValue::ADDITIONAL\_INFORMATION;

};

/\* Constant definitions for the notifyNewSwAvailability notification \*/

interface NotifyNewSwAvailability: NotificationIRPNotifications::Notify

{

const string EVENT\_TYPE = "notifyNewSwAvailability";

/\*\*

\* This constant defines the name of the NEandSWversion property,

\* which is transported in the filterable\_body\_fields.

\* The data type for the value of this property is

\* SwMIRPConstDefs::NEandSWversion.

\*/

const string NE\_AND\_SW\_VERSION =

SwMIRPConstDefs::AttributeNameValue::NE\_AND\_SW\_VERSION;

};

/\* Constant definitions for the notifySwMProfileChange notification \*/

interface NotifySwMProfileChange: NotificationIRPNotifications::Notify

{

const string EVENT\_TYPE = "notifySwMProfileChange";

/\*\*

\* 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 =

SwMIRPConstDefs::AttributeNameValue::ID;

/\*\*

\* This constant defines the name of the VersionNumber property,

\* which is transported in the filterable\_body\_fields.

\* The data type for the value of this property is

\* SwMIRPConstDefs::VersionNumber.

\*/

const string VERSION\_NUMBER =

SwMIRPConstDefs::AttributeNameValue::VERSION\_NUMBER;

/\*\*

\* 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 NE\_INFORMATION =

SwMIRPConstDefs::AttributeNameValue::NE\_INFORMATION;

/\*\*

\* This constant defines the name of the SwVersionToBeInstalled property,

\* which is transported in the filterable\_body\_fields.

\* The data type for the value of this property is

\* SwMIRPConstDefs::SwVersionToBeInstalledConditional.

\*/

const string SW\_VERSION\_TO\_BE\_INSTALLED =

SwMIRPConstDefs::AttributeNameValue::SW\_VERSION\_TO\_BE\_INSTALLED;

/\*\*

\* This constant defines the name of the StepsAndSelectedStopPointList property,

\* which is transported in the filterable\_body\_fields.

\* The data type for the value of this property is

\* SwMIRPConstDefs::StepsAndSelectedStopPointList.

\*/

const string STEPS\_AND\_SELECTED\_STOP\_POINT\_LIST =

SwMIRPConstDefs::AttributeNameValue::STEPS\_AND\_SELECTED\_STOP\_POINT\_LIST;

/\*\*

\* This constant defines the name of the SelectedFinalAdministrativeState property,

\* which is transported in the filterable\_body\_fields.

\* The data type for the value of this property is

\* SwMIRPConstDefs::SelectedFinalAdministrativeState.

\*/

const string SELECTED\_FINAL\_ADMINISTRATIVE\_STATE =

SwMIRPConstDefs::AttributeNameValue::SELECTED\_FINAL\_ADMINISTRATIVE\_STATE;

};

/\* Constant definitions for the notifyDownloadNESwStatusChanged notification \*/

interface notifyDownloadNESwStatusChanged: NotificationIRPNotifications::Notify

{

const string EVENT\_TYPE = "notifyDownloadNESwStatusChanged";

/\*\*

\* This constant defines the name of the downloadProcessId property used

\* during downloadNESw operation and transported in the

\* filterable\_body\_fields. The data type for the value of this property

\* is SwMIRPConstDefs::RequestID.

\*/

const string DOWNLOAD\_PROCESS\_ID =

SwMIRPConstDefs::AttributeNameValue::DOWNLOAD\_PROCESS\_ID;

/\*\*

\* This constant defines the name of the downloadOperationStatus

\* property which is transported in the filterable\_body\_fields.

\* The data type for the value of this property is

\* SwMIRPConstDefs::DownloadNESwOperationStatus.

\*/

const string DOWNLOAD\_OPERATION\_STATUS =

SwMIRPConstDefs::AttributeNameValue::DOWNLOAD\_OPERATION\_STATUS;

/\*\*

\* This constant defines the name of the downloadedNESwInfo

\* property, which is transported in the remaining\_body.

\* The data type for the value of this property is

\* SwMIRPConstDefs::DownloadedNESwInfo.

\*/

const string DOWNLOADED\_NESW\_INFO =

SwMIRPConstDefs::AttributeNameValue::DOWNLOADED\_NESW\_INFO;

/\*\*

\* This constant defines the name of the FailedSwInfo property,

\* which is transported in the remaining\_body.

\* The data type for the value of this property is

\* SwMIRPConstDefs::FailedSwInfo.

\*/

const string FAILED\_SW\_INFO =

SwMIRPConstDefs::AttributeNameValue::FAILED\_SW\_INFO;

};

/\* Constant definitions for the notifyInstallNESwStatusChanged notification \*/

interface notifyInstallNESwStatusChanged: NotificationIRPNotifications::Notify

{

const string EVENT\_TYPE = "notifyInstallNESwStatusChanged";

/\*\*

\* This constant defines the name of the InstallProcessId property used

\* during installNESw operation and transported in the

\* filterable\_body\_fields. The data type for the value of this property

\* is SwMIRPConstDefs::RequestID.

\*/

const string INSTALL\_PROCESS\_ID =

SwMIRPConstDefs::AttributeNameValue::INSTALL\_PROCESS\_ID;

/\*\*

\* This constant defines the name of the InstallOperationStatus

\* property which is transported in the filterable\_body\_fields.

\* The data type for the value of this property is

\* SwMIRPConstDefs::InstallOperationStatus.

\*/

const string INSTALL\_OPERATION\_STATUS =

SwMIRPConstDefs::AttributeNameValue::INSTALL\_OPERATION\_STATUS;

/\*\*

\* This constant defines the name of the InstalledNESwInfo property,

\* which is transported in the remaining\_body.

\* The data type for the value of this property is

\* SwMIRPConstDefs::InstalledNESwInfo.

\*/

const string INSTALLED\_NESW\_INFO =

SwMIRPConstDefs::AttributeNameValue::INSTALLED\_NESW\_INFO;

/\*\*

\* This constant defines the name of the FailedSwInfo property,

\* which is transported in the remaining\_body.

\* The data type for the value of this property is

\* SwMIRPConstDefs::FailedSwInfo.

\*/

const string FAILED\_SW\_INFO =

SwMIRPConstDefs::AttributeNameValue::FAILED\_SW\_INFO;

};

/\* Constant definitions for the notifyActivateNESwStatusChanged notification \*/

interface notifyActivateNESwStatusChanged: NotificationIRPNotifications::Notify

{

const string EVENT\_TYPE = "notifyActivateNESwStatusChanged";

/\*\*

\* This constant defines the name of the activateProcessId property used

\* during activateNESw operation and transported in the

\* filterable\_body\_fields. The data type for the value of this property

\* is SwMIRPConstDefs::RequestID.

\*/

const string ACTIVATE\_PROCESS\_ID =

SwMIRPConstDefs::AttributeNameValue::ACTIVATE\_PROCESS\_ID;

/\*\*

\* This constant defines the name of the ActivateOperationStatus

\* property which is transported in the filterable\_body\_fields.

\* The data type for the value of this property is

\* SwMIRPConstDefs::ActivateOperationStatus.

\*/

const string ACTIVATE\_OPERATION\_STATUS =

SwMIRPConstDefs::AttributeNameValue::ACTIVATE\_OPERATION\_STATUS;

/\*\*

\* This constant defines the name of the software version property

\* activated, which is transported in the filterable\_body\_fields.

\* The data type for the value of this property is

\* SwMIRPConstDefs::SWVersion.

\*/

const string SW\_VERSION =

SwMIRPConstDefs::AttributeNameValue::SW\_VERSION;

/\*\*

\* This constant defines the name of the failureReason property,

\* which is transported in the remaining\_body.

\* The data type for the value of this property is

\* SwMIRPConstDefs::FailureReason.

\*/

const string FAILURE\_REASON =

SwMIRPConstDefs::AttributeNameValue::FAILURE\_REASON;

};

};

#endif // \_SWM\_IRP\_NOTIFICATIONS\_IDL\_

Annex B (normative):   
XML definitions

The annex specifies the XML Definitions for the Software management Integration Reference Point (IRP) as it applies to Itf-N, in accordance with Software management IRP IS definitions [5], for usage with the Notification Log IRP XML Definitions [7].

# B.1 Architectural features

The overall architectural feature of Software management IRP is specified in 3GPP TS 32.532 [5].

This clause specifies features that are specific to the XML Schema definitions.

## B.1.1 Syntax for Distinguished Names

The syntax of a Distinguished Name is defined in 3GPP TS 32.300 [15].

# B.2 Mapping

Not present in the current version of this specification.

# B.3 Solution Set definitions

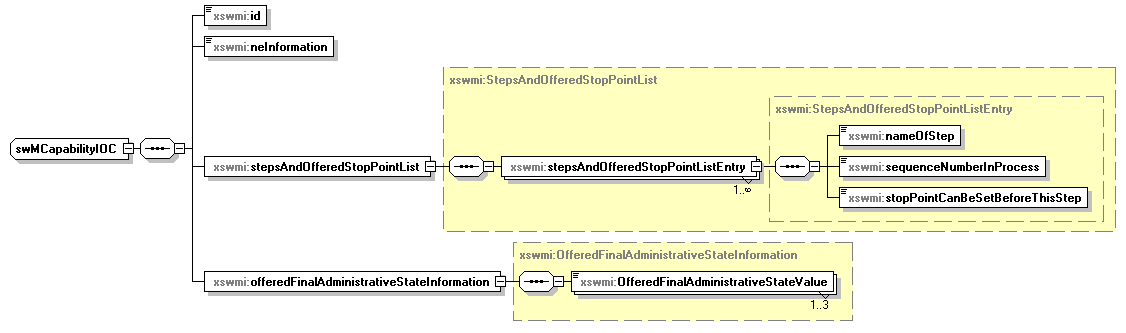
## B.3.1 XML definition structure

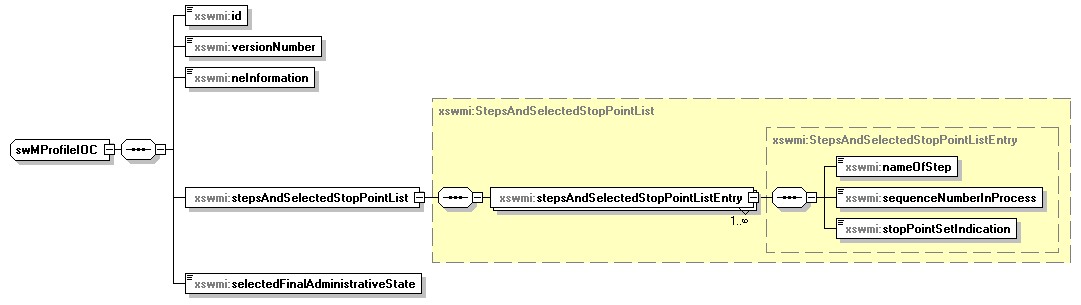
Clause B.3.2 provides a graphical representation of the XML elements.

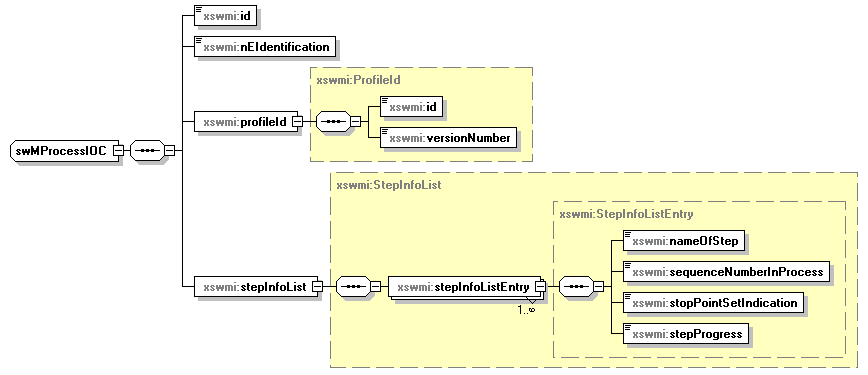
Clause B.3.3 provides XML definitions of Software management IRP notifications as defined in 3GPP TS 32.532 [5].

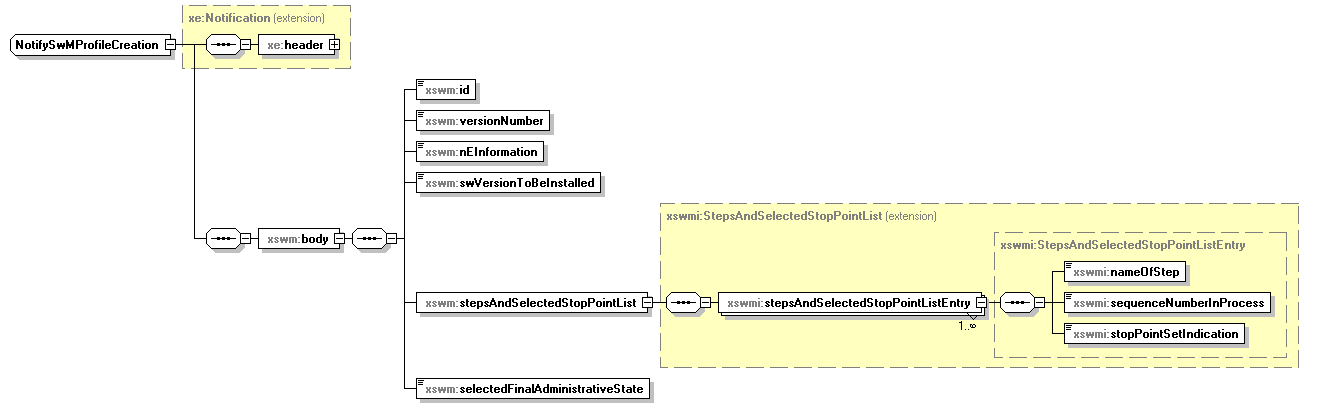
Clause B.3.4 provides XML definitions of Software management IRP IOCs as defined in 3GPP TS 32.532 [5].

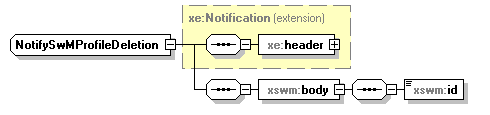
## B.3.2 Graphical Representation

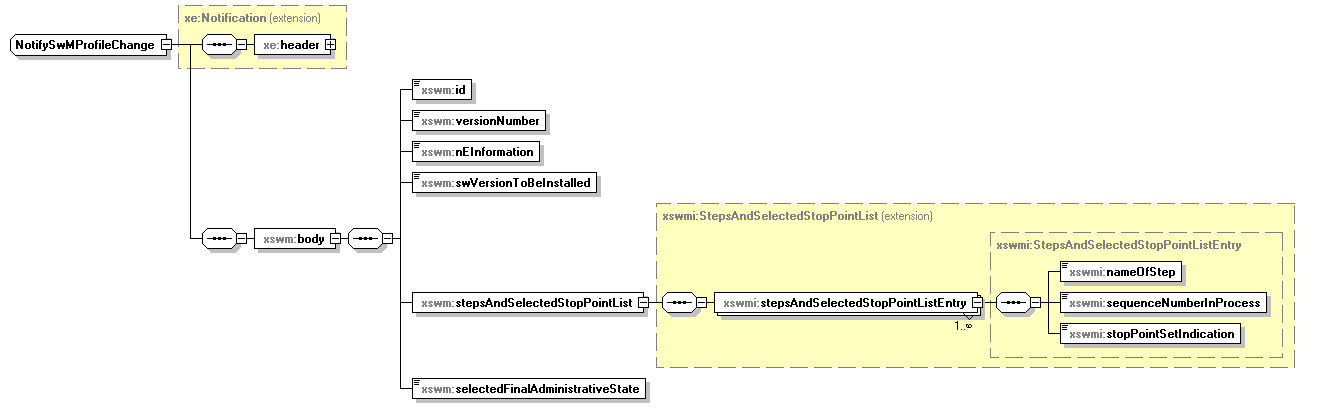


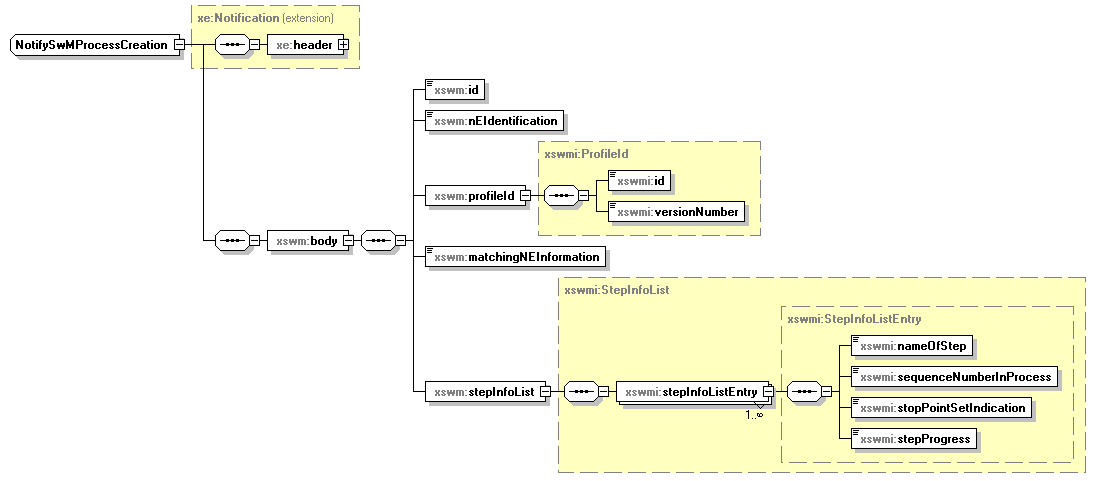


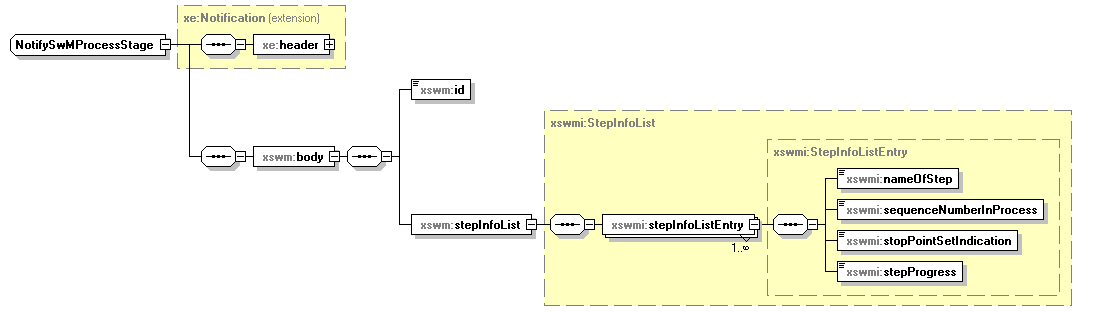


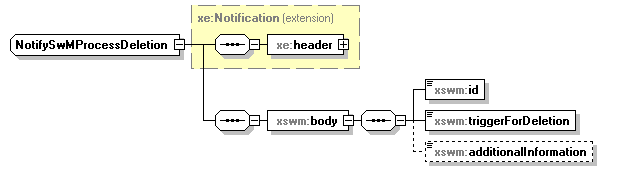


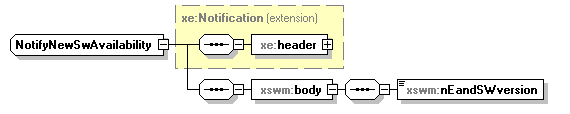


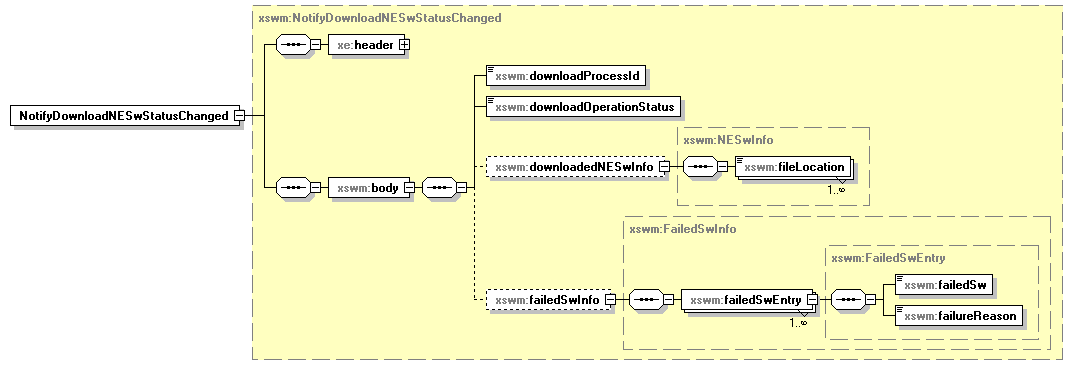


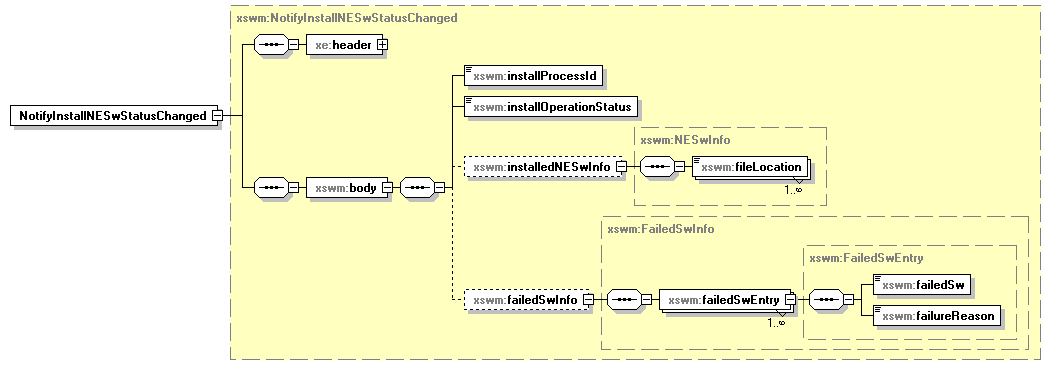


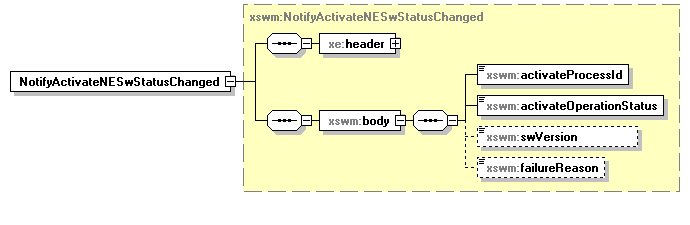












## B.3.3 XML Schema “swManagementIRPNotif.xsd”

<?xml version="1.0" encoding="UTF-8"?>

<schema xmlns="http://www.w3.org/2001/XMLSchema" xmlns:xe="http://www.3gpp.org/ftp/specs/archive/32\_series/32.306#notification" xmlns:xswm="http://www.3gpp.org/ftp/specs/archive/32\_series/32.536#swManagementIRPNotif" xmlns:xswmi="http://www.3gpp.org/ftp/specs/archive/32\_series/32.536#swManagementIRPIOCs" targetNamespace="http://www.3gpp.org/ftp/specs/archive/32\_series/32.536#swManagementIRPNotif" elementFormDefault="qualified" attributeFormDefault="unqualified">

<import namespace="http://www.3gpp.org/ftp/specs/archive/32\_series/32.306#notification"/>

<import namespace="http://www.3gpp.org/ftp/specs/archive/32\_series/32.536#swManagementIRPIOCs"/>

<simpleType name="SwVersionToBeInstalledConditional">

<restriction base="string"/>

</simpleType>

<complexType name="OfferedFinalAdministrativeStateInformation">

<sequence>

<element name="OfferedFinalAdministrativeStateValue" type="xswmi:FinalAdministrativeStateValue" maxOccurs="3"/>

</sequence>

</complexType>

<simpleType name="TriggerForDeletion">

<restriction base="string">

<enumeration value="IRP\_AGENT\_TERMINATION"/>

<enumeration value="IRP\_MANAGER\_TERMINATION"/>

<enumeration value="AUTOMATED\_SWM\_SUCCESFULLY\_CONCLUDED"/>

<enumeration value="SELF\_CONFIGURATION\_SUCCESFULLY\_CONCLUDED"/>

</restriction>

</simpleType>

<simpleType name="RequestID">

<restriction base="unsignedLong"/>

</simpleType>

<simpleType name="DownloadNESwOperationStatus">

<restriction base="string">

<enumeration value="NE\_SWDOWNLOAD\_SUCCESSFUL"/>

<enumeration value="NE\_SWDOWNLOAD\_FAILED"/>

<enumeration value="NE\_SWDOWNLOAD\_PARTIALLY\_SUCCESSFUL"/>

</restriction>

</simpleType>

<simpleType name="FileLocation">

<restriction base="string"/>

</simpleType>

<complexType name="NESwInfo">

<sequence>

<element name="fileLocation" type="xswm:FileLocation" maxOccurs="unbounded"/>

</sequence>

</complexType>

<complexType name="FailedSwEntry">

<sequence>

<element name="failedSw" type="xswm:FileLocation"/>

<element name="failureReason" type="xswm:FailureReason"/>

</sequence>

</complexType>

<complexType name="FailedSwInfo">

<sequence>

<element name="failedSwEntry" type="xswm:FailedSwEntry" maxOccurs="unbounded"/>

</sequence>

</complexType>

<simpleType name="FailureReason">

<restriction base="string"/>

</simpleType>

<simpleType name="InstallNESwOperationStatus">

<restriction base="string">

<enumeration value="NE\_SW\_INSTALLATION\_SUCCESSFUL"/>

<enumeration value="NE\_SW\_INSTALLATION\_FAILED"/>

<enumeration value="NE\_SWINSTALLATION\_PARTIALLY\_SUCCESSFUL"/>

</restriction>

</simpleType>

<simpleType name="ActivateNESwOperationStatus">

<restriction base="string">

<enumeration value="NE\_SW\_ACTIVATION\_SUCCESSFUL"/>

<enumeration value="NE\_SW\_ACTIVATION\_FAILED"/>

<enumeration value="NE\_SWACTIVATION\_PARTIALLY\_SUCCESSFUL"/>

</restriction>

</simpleType>

<complexType name="NotifySwMProfileChange">

<complexContent>

<extension base="xe:Notification">

<sequence>

<element name="body">

<complexType>

<sequence>

<element name="id" type="xswmi:Id"/>

<element name="versionNumber" type="xswmi:VersionNumber"/>

<element name="nEInformation" type="xswmi:NEInformation"/>

<element name="swVersionToBeInstalled" type="xswm:SwVersionToBeInstalledConditional"/>

<element name="stepsAndSelectedStopPointList">

<complexType>

<complexContent>

<extension base="xswmi:StepsAndSelectedStopPointList"/>

</complexContent>

</complexType>

</element>

<element name="selectedFinalAdministrativeState" type="xswmi:FinalAdministrativeStateValue"/>

</sequence>

</complexType>

</element>

</sequence>

</extension>

</complexContent>

</complexType>

<complexType name="NotifyNewSwAvailability">

<complexContent>

<extension base="xe:Notification">

<sequence>

<element name="body">

<complexType>

<sequence>

<element name="nEandSWversion" type="string"/>

</sequence>

</complexType>

</element>

</sequence>

</extension>

</complexContent>

</complexType>

<complexType name="NotifySwMProcessDeletion">

<complexContent>

<extension base="xe:Notification">

<sequence>

<element name="body">

<complexType>

<sequence>

<element name="id" type="xswmi:Id"/>

<element name="triggerForDeletion" type="xswm:TriggerForDeletion"/>

<element name="additionalInformation" type="string" minOccurs="0"/>

</sequence>

</complexType>

</element>

</sequence>

</extension>

</complexContent>

</complexType>

<complexType name="NotifySwMProcessStage">

<complexContent>

<extension base="xe:Notification">

<sequence>

<element name="body">

<complexType>

<sequence>

<element name="id" type="xswmi:Id"/>

<element name="stepInfoList" type="xswmi:StepInfoList"/>

</sequence>

</complexType>

</element>

</sequence>

</extension>

</complexContent>

</complexType>

<complexType name="NotifySwMProcessCreation">

<complexContent>

<extension base="xe:Notification">

<sequence>

<element name="body">

<complexType>

<sequence>

<element name="id" type="xswmi:Id"/>

<element name="nEIdentification" type="xswmi:Id"/>

<element name="profileId" type="xswmi:ProfileId"/>

<element name="matchingNEInformation" type="xswmi:NEInformation"/>

<element name="stepInfoList" type="xswmi:StepInfoList"/>

</sequence>

</complexType>

</element>

</sequence>

</extension>

</complexContent>

</complexType>

<complexType name="NotifySwMProfileDeletion">

<complexContent>

<extension base="xe:Notification">

<sequence>

<element name="body">

<complexType>

<sequence>

<element name="id" type="xswmi:Id"/>

</sequence>

</complexType>

</element>

</sequence>

</extension>

</complexContent>

</complexType>

<complexType name="NotifySwMProfileCreation">

<complexContent>

<extension base="xe:Notification">

<sequence>

<element name="body">

<complexType>

<sequence>

<element name="id" type="xswmi:Id"/>

<element name="versionNumber" type="xswmi:VersionNumber"/>

<element name="nEInformation" type="xswmi:NEInformation"/>

<element name="swVersionToBeInstalled" type="xswm:SwVersionToBeInstalledConditional"/>

<element name="stepsAndSelectedStopPointList">

<complexType>

<complexContent>

<extension base="xswmi:StepsAndSelectedStopPointList"/>

</complexContent>

</complexType>

</element>

<element name="selectedFinalAdministrativeState" type="xswmi:FinalAdministrativeStateValue"/>

</sequence>

</complexType>

</element>

</sequence>

</extension>

</complexContent>

</complexType>

<complexType name="NotifyDownloadNESwStatusChanged">

<complexContent>

<extension base="xe:Notification">

<sequence>

<element name="body">

<complexType>

<sequence>

<element name="downloadProcessId" type="xswm:RequestID"/>

<element name="downloadOperationStatus" type="xswm:DownloadNESwOperationStatus"/>

<element name="downloadedNESwInfo" type="xswm:NESwInfo" minOccurs="0"/>

<element name="failedSwInfo" type="xswm:FailedSwInfo" minOccurs="0"/>

</sequence>

</complexType>

</element>

</sequence>

</extension>

</complexContent>

</complexType>

<complexType name="NotifyInstallNESwStatusChanged">

<complexContent>

<extension base="xe:Notification">

<sequence>

<element name="body">

<complexType>

<sequence>

<element name="installProcessId" type="xswm:RequestID"/>

<element name="installOperationStatus" type="xswm:InstallNESwOperationStatus"/>

<element name="installedNESwInfo" type="xswm:NESwInfo" minOccurs="0"/>

<element name="failedSwInfo" type="xswm:FailedSwInfo" minOccurs="0"/>

</sequence>

</complexType>

</element>

</sequence>

</extension>

</complexContent>

</complexType>

<complexType name="NotifyActivateNESwStatusChanged">

<complexContent>

<extension base="xe:Notification">

<sequence>

<element name="body">

<complexType>

<sequence>

<element name="activateProcessId" type="xswm:RequestID"/>

<element name="activateOperationStatus" type="xswm:ActivateNESwOperationStatus"/>

<element name="swVersion" type="string"/>

<element name="failureReason" type="xswm:FailureReason" minOccurs="0"/>

</sequence>

</complexType>

</element>

</sequence>

</extension>

</complexContent>

</complexType>

<element name="NotifySwMProfileCreation" type="xswm:NotifySwMProfileCreation"/>

<element name="NotifySwMProfileDeletion" type="xswm:NotifySwMProfileDeletion"/>

<element name="NotifySwMProcessCreation" type="xswm:NotifySwMProcessCreation"/>

<element name="NotifySwMProcessStage" type="xswm:NotifySwMProcessStage"/>

<element name="NotifySwMProcessDeletion" type="xswm:NotifySwMProcessDeletion"/>

<element name="NotifyNewSwAvailability" type="xswm:NotifyNewSwAvailability"/>

<element name="NotifySwMProfileChange" type="xswm:NotifySwMProfileChange"/>

<element name="NotifyDownloadNESwStatusChanged" type="xswm:NotifyDownloadNESwStatusChanged"/>

<element name="NotifyActivateNESwStatusChanged" type="xswm:NotifyActivateNESwStatusChanged"/>

<element name="NotifyInstallNESwStatusChanged" type="xswm:NotifyInstallNESwStatusChanged"/>

</schema>

## B.3.4 XML Schema “swManagementIRPIOCs.xsd”

<?xml version="1.0" encoding="UTF-8"?>

<schema xmlns="http://www.w3.org/2001/XMLSchema" xmlns:xswmi="http://www.3gpp.org/ftp/specs/archive/32\_series/32.536#swManagementIRPIOCs" xmlns:xn="http://www.3gpp.org/ftp/specs/archive/32\_series/32.626#genericNrm" targetNamespace="http://www.3gpp.org/ftp/specs/archive/32\_series/32.536#swManagementIRPIOCs" elementFormDefault="qualified" attributeFormDefault="unqualified">

<import namespace="http://www.3gpp.org/ftp/specs/archive/32\_series/32.626#genericNrm" schemaLocation="genericNrm.xsd"/>

<simpleType name="NEInformation">

<restriction base="string"/>

</simpleType>

<simpleType name="Id">

<restriction base="xn:dn"/>

</simpleType>

<simpleType name="VersionNumber">

<restriction base="unsignedShort"/>

</simpleType>

<complexType name="ProfileId">

<sequence>

<element name="id" type="xswmi:Id"/>

<element name="versionNumber" type="xswmi: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="xswmi:FinalAdministrativeStateValue" maxOccurs="3"/>

</sequence>

</complexType>

<complexType name="StepsAndOfferedStopPointListEntry">

<sequence>

<element name="nameOfStep" type="xswmi:NameOfStep"/>

<element name="sequenceNumberInProcess" type="xswmi:SequenceNumberInProcess"/>

<element name="stopPointCanBeSetBeforeThisStep" type="xswmi:StopPointCanBeSetBeforeThisStep"/>

</sequence>

</complexType>

<complexType name="StepsAndOfferedStopPointList">

<sequence>

<element name="stepsAndOfferedStopPointListEntry" type="xswmi:StepsAndOfferedStopPointListEntry" maxOccurs="unbounded"/>

</sequence>

</complexType>

<simpleType name="StopPointCanBeSetBeforeThisStep">

<restriction base="boolean"/>

</simpleType>

<complexType name="StepsAndSelectedStopPointListEntry">

<sequence>

<element name="nameOfStep" type="xswmi:NameOfStep"/>

<element name="sequenceNumberInProcess" type="xswmi:SequenceNumberInProcess"/>

<element name="stopPointSetIndication" type="xswmi:StopPointSetIndication"/>

</sequence>

</complexType>

<complexType name="StepsAndSelectedStopPointList">

<sequence>

<element name="stepsAndSelectedStopPointListEntry" type="xswmi:StepsAndSelectedStopPointListEntry" maxOccurs="unbounded"/>

</sequence>

</complexType>

<simpleType name="StopPointSetIndication">

<restriction base="string">

<enumeration value="STOP\_POINT\_IS\_SET\_BEFORE\_THIS\_STEP"/>

<enumeration value="STOP\_POINT\_IS\_NOT\_SET"/>

</restriction>

</simpleType>

<simpleType name="SequenceNumberInProcess">

<restriction base="unsignedShort"/>

</simpleType>

<simpleType name="NameOfStep">

<restriction base="string">

<enumeration value="NE\_HEALTH\_CHECK"/>

<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="xswmi:NameOfStep"/>

<element name="sequenceNumberInProcess" type="xswmi:SequenceNumberInProcess"/>

<element name="stopPointSetIndication" type="xswmi:StopPointSetIndication"/>

<element name="stepProgress" type="xswmi:StepProgress"/>

</sequence>

</complexType>

<complexType name="StepInfoList">

<sequence>

<element name="stepInfoListEntry" type="xswmi:StepInfoListEntry" maxOccurs="unbounded"/>

</sequence>

</complexType>

<complexType name="swMCapabilityIOC">

<sequence>

<element name="id" type="xswmi:Id"/>

<element name="neInformation" type="xswmi:NEInformation"/>

<element name="stepsAndOfferedStopPointList" type="xswmi:StepsAndOfferedStopPointList"/>

<element name="offeredFinalAdministrativeStateInformation" type="xswmi:OfferedFinalAdministrativeStateInformation"/>

</sequence>

</complexType>

<simpleType name="ProcessStepNumber">

<restriction base="unsignedShort"/>

</simpleType>

<simpleType name="Hours">

<restriction base="unsignedShort"/>

</simpleType>

<simpleType name="Minutes">

<restriction base="unsignedShort">

<maxInclusive value="59"/>

</restriction>

</simpleType>

<simpleType name="Seconds">

<restriction base="unsignedShort">

<maxInclusive value="59"/>

</restriction>

</simpleType>

...<complexType name="HoursMinutesSeconds">

......<sequence>

.........<element name="hours" type="xswmi:Hours"/>

.........<element name="minutes" type="xswmi:Minutes"/>

.........<element name="seconds" type="xswmi:Seconds"/>

......</sequence>

</complexType>

<simpleType name="Naswm OperationType">

<restriction base="string">

<enumeration value="OPERATION\_DOWNLOAD\_NE\_SW"/>

<enumeration value="OPERATION\_ACTIVATE\_NE\_SW"/>

<enumeration value="OPERATION\_INSTALL\_NE\_SW"/>

</restriction>

</simpleType>

...<complexType name="StepNumberAndDuration">

......<sequence>

.........<element name="processStepNumber" type="xswmi:ProcessStepNumber"/>

.........<element name="estimatedDuration" type="xswmi:HoursMinutesSeconds"/>

......</sequence>

</complexType>

...<complexType name="ListOfStepNumbersAndDurations">

......<sequence>

.........<element name="stepNumberAndDuration" type="xswmi:StepNumberAndDuration" maxOccurs=”unbounded”/>

......</sequence>

</complexType>

...<complexType name="NaswmProcessInfo">

......<sequence>

.........<element name="naswmId" type="xswmi:Id"/>

.........<element name="nEIdentification" type="xswmi:NEInformation"/>

.........<element name="naswmOperationType" type="xswmi:NaswmOperationType"/>

.........<element name="estimatedRemainingCompletionTimeForTheOperation" type="xswmi:HoursMinutesSeconds"/>

.........<element name="listOfStepNumbersAndDurations" type="xswmi:ProcessStepNumberOpt" minOccurs=”0”/>

.........<element name="numberOfCurrentProcessStep" type="xswmi:NumberOfCurrentProcessStep" minOccurs=”0”/>

.........<element name="estimatedRemainingCompletionTimeForTheCurrentStep" type="xswmi:HoursMinutesSeconds" minOccurs=”0”/>

......</sequence>

</complexType>

...<complexType name="NaswmProcessList">

......<sequence>

.........<element name="naswmProcessInfo" type="xswmi:NaswmProcessInfo" maxOccurs=”unbounded”/>

......</sequence>

</complexType>

<complexType name="swMProfileIOC">

<sequence>

<element name="id" type="xswmi:Id"/>

<element name="versionNumber" type="xswmi:VersionNumber"/>

<element name="neInformation" type="xswmi:NEInformation"/>

<element name="stepsAndSelectedStopPointList" type="xswmi:StepsAndSelectedStopPointList"/>

<element name="selectedFinalAdministrativeState" type="xswmi:FinalAdministrativeStateValue"/>

</sequence>

</complexType>

<complexType name="swMProcessIOC">

<sequence>

<element name="id" type="xswmi:Id"/>

<element name="nEIdentification" type="xswmi:NEInformation"/>

<element name="profileId" type="xswmi:ProfileId"/>

<element name="stepInfoList" type="xswmi:StepInfoList"/>

</sequence>

</complexType>

</schema>

Annex C (normative):  
SOAP Solution Set

The overall architectural feature of the Software Management IRP is specified in 3GPP TS 32.532 [5]. This clause specifies features that are specific to the SOAP solution set.

# C.1 Architectural Features

## C.1.1 Syntax for Distinguished Names and versions

The syntax of a Distinguished Name is defined in 3GPP TS 32.300 [15].

## C.1.2 General

The SOAP 1.1 specification [11] and WSDL 1.1 specification [13] are supported.

The SOAP 1.2 specification [14] 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 [12]). IRPAgents may throw a FilterComplexityLimit fault when a given filter is too complex.

Relevant definitions are imported from the Software Management XML definitions of Annex B.

This specification uses a number of namespace prefixes throughout that are listed in Table C.1.2.

Table C.1.2: Prefixes and Namespaces used in this specification

|  |  |
| --- | --- |
| **PREFIX** | **NAMESPACE** |
| (no prefix) | http://schemas.xmlsoap.org/wsdl/ |
| soap | http://schemas.xmlsoap.org/wsdl/soap/ |
| swMIRPSystem | http://www.3gpp.org/ftp/specs/archive/32\_series/32.536#SWMIRPSystem |
| swMIRPData | http://www.3gpp.org/ftp/specs/archive/32\_series/32.536#SWMIRPData |
| xswmi | http://www.3gpp.org/ftp/specs/archive/32\_series/32.536#swManagementIRPIOCs |
| genericIRPSystem | http://www.3gpp.org/ftp/specs/archive/32\_series/32.316#GenericIRPSystem |
| ntfIRPNtfSystem | http://www.3gpp.org/ftp/specs/archive/32\_series/32.536#NotificationSwNtfSystem |

# C.2 Mapping

## C.2.1 Operation and Notification mapping

The Software Management IRP IS (3GPP TS 32.532 [5]) defines the operations and their semantics.

Table C.2.1 maps the operations defined in the Software Management 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 Software Management 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 32.532 [5] | SS: Operation for WSDL port type and WSDL binding | SS: Port of EntryPointIRPService | Qualifier |
| listSwMCapabilities | listSwMCapabilities(NOTE1) | SwMIRPOperation1 | M |
| listSwMProfiles | listSwMProfiles(NOTE1) | SwMIRPOperation1 | M |
| createSwMProfile | createSwMProfile(NOTE1) | SwMIRPOperation1 | M |
| deleteSwMProfile | deleteSwMProfile(NOTE1) | SwMIRPOperation1 | M |
| listSwMProcesses | listSwMProcesses(NOTE1) | SwMIRPOperation1 | M |
| resumeSwMProcess | resumeSwMProcess(NOTE1) | SwMIRPOperation1 | M |
| swFallback | swFallback(NOTE1) | SwMIRPOperation1 | M |
| terminateSwMProcess | terminateSwMProcess(NOTE1) | SwMIRPOperation1 | M |
| changeSwMProfile | changeSwMProfile(NOTE1) | SwMIRPOperation2 | O |
| downloadNESw | downloadNESw(NOTE1) | SwMIRPOperation3 | M |
| activateNESw | activateNESw (NOTE1) | SwMIRPOperation3 | O |
| installNESw | installNESw (NOTE1) | SwMIRPOperation4 | M |
| notifySwMProfileCreation | notifySwMProfileCreation(NOTE2) | NotificationIRPNtfPort | M |
| notifySwMProfileDeletion | notifySwMProfileDeletion(NOTE2) | NotificationIRPNtfPort | M |
| notifySwMProcessCreation | notifySwMProcessCreation(NOTE2) | NotificationIRPNtfPort | M |
| notifySwMProcessStage | notifySwMProcessStage(NOTE2) | NotificationIRPNtfPort | M |
| notifySwMProcessDeletion | notifySwMProcessDeletion(NOTE2) | NotificationIRPNtfPort | M |
| notifyNewSwAvailability | notifyNewSwAvailability(NOTE2) | NotificationIRPNtfPort | M |
| notifySwMProfileChange | notifySwMProfileChange(NOTE2) | NotificationIRPNtfPort | O |
| notifyDownloadNESwStatusChanged | notifyDownloadNESwStatusChanged  (NOTE2) | NotificationIRPNtfPort | M |
| notifyInstallNESwStatusChanged | notifyInstallNESwStatusChanged  (NOTE2) | NotificationIRPNtfPort | O |
| notifyActivateNESwStatusChanged | notifyActivateNESwStatusChanged  (NOTE2) | NotificationIRPNtfPort | M |
| NOTE 1: The operation is under the port type SwMIRPSystem:SwMIRPPortType and under the binding SwMIRPSystem:SwMIRPBinding. | | | |
| 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 [10]. This binding is linked to a port of the SwMIRPService as indicated in the table above. | | | |

## C.2.2 Operation parameter mapping

The SwMIRP IS (3GPP TS 32.532 [5]) 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 listSwMCapabilities

#### C.2.2.1.1 Input parameters

Table C.2.2.1.1: Mapping from IS listSwMCapabilities 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 listSwMCapabilities output parameters to SS equivalents

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

#### C.2.2.1.3 Fault definition

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

|  |  |  |
| --- | --- | --- |
| Assertion name | SS WSDL type enumeration value  used in corresponding fault message  under corresponding port type operation  as indicated in Table C.2.1 | Qualifier |
| operation\_failed | operation\_failed | M |

### C.2.2.2 Operation listSwMProfiles

#### C.2.2.2.1 Input parameters

TableC.2.2.2.1: Mapping from IS listSwMProfiles 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

TableC.2.2.2.2: Mapping from IS listSwMProfiles output parameters to SS equivalents

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

### C.2.2.3 Operation createSwMProfile

#### C.2.2.3.1 Input parameters

TableC.2.2.3.1: Mapping from IS createSwMProfile 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 createSwMProfile output parameters to SS equivalents

|  |  |  |
| --- | --- | --- |
| IS Operation parameter | SS WSDL type sub-element  used in corresponding output 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 deleteSwMProfile

#### C.2.2.4.1 Input parameters

TableC.2.2.4.1: Mapping from IS deleteSwMProfile 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 deleteSwMProfile output parameters to SS equivalents

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

### C.2.2.5 Operation listSwMProcesses

#### C.2.2.5.1 Input parameters

TableC.2.2.5.1: Mapping from IS listSwMProcesses 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 listSwMProcesses output parameters to SS equivalents

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

### C.2.2.6 Operation resumeSwMProcess

#### C.2.2.6.1 Input parameters

TableC.2.2.6.1: Mapping from IS resumeSwMProcess 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 resumeSwMProcess output parameters to SS equivalents

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

### C.2.2.7 Operation swFallback

#### C.2.2.7.1 Input parameters

TableC.2.2.7.1: Mapping from IS swFallback 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 |
| filter | filter | M |

#### C.2.2.7.2 Output parameters

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

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

### C.2.2.8 Operation terminateSwMProcess

#### C.2.2.8.1 Input parameters

TableC.2.2.8.1: Mapping from IS terminateSwMProcess 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.2.2.8.2 Output parameters

Table C.2.2.8.2: Mapping from IS terminateSwMProcess output parameters to SS equivalents

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

### C.2.2.9 Operation changeSwMProfile

#### C.2.2.9.1 Input parameters

TableC.2.2.9.1: Mapping from IS changeSwMProfile 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.2.2.9.2 Output parameters

Table C.2.2.9.2: Mapping from IS changeSwMProfile input parameters to SS equivalents

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

### C.2.2.10 Operation downloadNESw

#### C.2.2.10.1 Input parameters

TableC.2.2.10.1: Mapping from IS downloadNESw 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 |
| swToBeDownloaded | swToBeDownloaded | M |
| neIdentifier | neIdentifier | M |

#### C.2.2.10.2 Output parameters

Table C.2.2.10.2: Mapping from IS downloadNESw output parameters to SS equivalents

|  |  |  |
| --- | --- | --- |
| IS Operation parameter | SS WSDL type sub-element  used in corresponding output message  under corresponding port type operation  as indicated in Table C.2.1 | Qualifier |
| downloadProcessId | downloadProcessId | M |
| result | result | M |
| reason | reason | O |
| listOfStepNumbersAndDurations | listOfStepNumbersAndDurations | CM \*) |

\*) Note: For the condition see TS 32.532 [5].

#### C.2.2.10.3 Fault definition

Table C.2.2.10.3: Mapping from IS downloadNESw exceptions to SS equivalents

|  |  |  |
| --- | --- | --- |
| Assertion name | SS WSDL type enumeration value  used in corresponding fault message  under corresponding port type operation  as indicated in Table C.2.1 | Qualifier |
| operationFailed | operationFailed | M |
| resourceLimitation | resourceLimitation | M |

### C.2.2.11 Operation activateNESw

#### C.2.2.11.1 Input parameters

TableC.2.2.11.1: Mapping from IS activateNESw 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 |
| swVersionToBeActivated | swVersionToBeActivated | M |
| neIdentifier | neIdentifier | M |

#### C.2.2.11.2 Output parameters

Table C.2.2.11.2: Mapping from IS activateNESw output parameters to SS equivalents

|  |  |  |
| --- | --- | --- |
| IS Operation parameter | SS WSDL type sub-element  used in corresponding output message  under corresponding port type operation  as indicated in Table C.2.1 | Qualifier |
| activateProcessId | activateProcessId | M |
| result | result | M |
| reason | reason | O |
| listOfStepNumbersAndDurations | listOfStepNumbersAndDurations | CM \*) |

\*) Note: For the condition see TS 32.532 [5].

#### C.2.2.11.3 Fault definition

Table C.2.2.11.3: Mapping from IS activateNESw exceptions to SS equivalents

|  |  |  |
| --- | --- | --- |
| Assertion name | SS WSDL type enumeration value  used in corresponding fault message  under corresponding port type operation  as indicated in Table C.2.1 | Qualifier |
| operationFailed | operationFailed | M |
| resourceLimitation | resourceLimitation | M |

### C.2.2.12 Operation installNESw

#### C.2.2.12.1 Input parameters

TableC.2.2.12.1: Mapping from IS installNESw 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 |
| swTobeInstalled | swTobeInstalled | M |
| neIdentifier | neIdentifier | M |

#### C.2.2.12.2 Output parameters

Table C.2.2.12.2: Mapping from IS installNESw output parameters to SS equivalents

|  |  |  |
| --- | --- | --- |
| IS Operation parameter | SS WSDL type sub-element  used in corresponding output message  under corresponding port type operation  as indicated in Table C.2.1 | Qualifier |
| installProcessId | installProcessId | M |
| result | result | M |
| reason | reason | O |
| listOfStepNumbersAndDurations | listOfStepNumbersAndDurations | CM \*) |

\*) Note: For the condition see TS 32.532 [5].

#### C.2.2.12.3 Fault definition

Table C.2.2.12.3: Mapping from IS installNESw exceptions to SS equivalents

|  |  |  |
| --- | --- | --- |
| Assertion name | SS WSDL type enumeration value  used in corresponding fault message  under corresponding port type operation  as indicated in Table C.2.1 | Qualifier |
| operationFailed | operationFailed | M |
| resourceLimitation | resourceLimitation | M |
| swNotAvailable | swNotAvailable | M |

### C.2.2.13 Operation listNaswmProcesses

#### C.2.2.13.1 Input parameters

TableC.2.2.13.1: Mapping from IS listNaswmProcesses 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 |
| naswmId | id | O |
| naswmOperationType | naswmOperationType | M |

#### C.2.2.13.2 Output parameters

Table C.2.2.13.2: Mapping from IS listNaswmProcesses output parameters to SS equivalents

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

#### C.2.2.13.3 Fault definition

Table C.2.2.13.3: Mapping from IS listNaswmProcesses exceptions to SS equivalents

|  |  |  |
| --- | --- | --- |
| Assertion name | SS WSDL type enumeration value  used in corresponding fault message  under corresponding port type operation  as indicated in Table C.2.1 | Qualifier |
| operationFailed | operationFailed | M |
| resourceLimitation | resourceLimitation | M |
| operationIsAlreadyCompleted | operationIsAlreadyCompleted | M |
| noSuchProcess | noSuchProcess | M |

### C.2.2.14 Operation cancelNaswmProcesses

#### C.2.2.14.1 Input parameters

Table C.2.2.14.1: Mapping from IS cancelNaswmProcesses 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 |
| processId | Id | M |
| naswmOperationType | NaswmOperationType | M |

#### C.2.2.14.2 Output parameters

Table C.2.2.14.2: Mapping from IS cancelNaswmProcesses output parameters to SS equivalents

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

#### C.2.2.14.3 Fault definition

Table C.2.2.14.3: Mapping from IS cancelNaswmProcesses exceptions to SS equivalents

|  |  |  |
| --- | --- | --- |
| Assertion name | SS WSDL type enumeration value  used in corresponding fault message  under corresponding port type operation  as indicated in Table C.2.1 | Qualifier |
| operationFailed | operationFailed | M |
| resourceLimitation | resourceLimitation | M |

# C.3 Solution Set definitions

## C.3.1 WSDL definition structure

Clause C.3.2 provides a graphical representation of the Software management IRP service.

Clause C.3.3 defines the services which are supported the Software management 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.

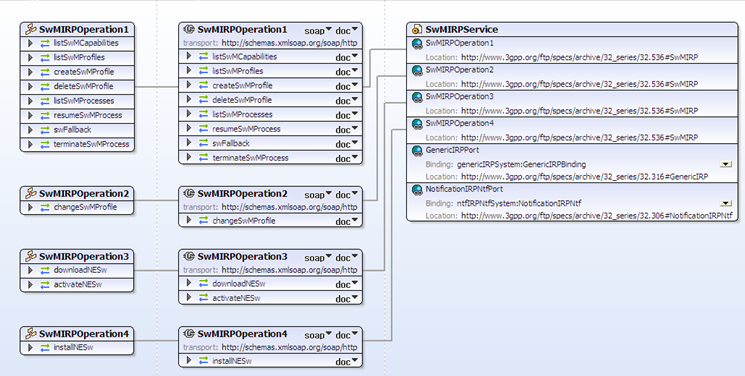


Figure C.3.2: Software Management IRP SOAP Solution Set WSDL structure

## C.3.3 WSDL specification “SWMIRPSystem.wsdl”

<?xml version="1.0" encoding="UTF-8"?>

<!--

3GPP TS 32.536 Software Management IRP SOAP Solution Set

-->

<definitions xmlns="http://schemas.xmlsoap.org/wsdl/" xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/" xmlns:swMIRPSystem="http://www.3gpp.org/ftp/specs/archive/32\_series/32.536#SWMIRPSystem" xmlns:swMIRPData="http://www.3gpp.org/ftp/specs/archive/32\_series/32.536#SWMIRPData" xmlns:xswmi="http://www.3gpp.org/ftp/specs/archive/32\_series/32.536#SwMIRPIOCs" 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#GenericIRPSystem" xmlns:ntfIRPNtfSystem="http://www.3gpp.org/ftp/specs/archive/32\_series/32.306#NotificationIRPNtfSystem" xmlns:ns="http://www.w3.org/2001/XMLSchema" targetNamespace="http://www.3gpp.org/ftp/specs/archive/32\_series/32.536#SWMIRPSystem">

<import namespace="http://www.3gpp.org/ftp/specs/archive/32\_series/32.306#NotificationIRPNtfSystem"/>

<import namespace="http://www.3gpp.org/ftp/specs/archive/32\_series/32.316#GenericIRPSystem"/>

<types>

<schema xmlns="http://www.w3.org/2001/XMLSchema" xmlns:xswmi="http://www.3gpp.org/ftp/specs/archive/32\_series/32.536#swManagementIRPIOCs" targetNamespace="http://www.3gpp.org/ftp/specs/archive/32\_series/32.536#SWMIRPData">

<import namespace="http://www.3gpp.org/ftp/specs/archive/32\_series/32.536#swManagementIRPIOCs"/>

<import namespace="http://www.3gpp.org/ftp/specs/archive/32\_series/32.626#genericNrm"/>

<!-- The following types are defined for the SW IRP operations -->

<complexType name="capability">

<sequence>

<element name="Id" type="xswmi:Id"/>

<element name="NEInformation" type="xswmi:NEInformation"/>

<element name="StepsAndOfferedStopPointList" type="xswmi:StepsAndOfferedStopPointList"/>

<element name="OfferedFinalAdministrativeStateInformation" type="xswmi:OfferedFinalAdministrativeStateInformation"/>

</sequence>

</complexType>

<complexType name="capabilityList">

<sequence>

<element name="capability" type="swMIRPData:capability" maxOccurs="unbounded"/>

</sequence>

</complexType>

<complexType name="swMProfile">

<sequence>

<element name="Id" type="xswmi:Id"/>

<element name="VersionNumber" type="xswmi:VersionNumber"/>

<element name="NEInformation" type="xswmi:NEInformation"/>

<element name="StepsAndSelectedStopPointList" type="xswmi:StepsAndSelectedStopPointList"/>

<element name="FinalAdministrativeStateValue" type="xswmi:FinalAdministrativeStateValue"/>

<element name="swVersionToBeInstalled" type="xswmi:VersionNumber" minOccurs="0"/>

</sequence>

</complexType>

<complexType name="swMProfileList">

<sequence>

<element name="swMProfile" type="swMIRPData:swMProfile" maxOccurs="unbounded"/>

</sequence>

</complexType>

<complexType name="swMprocess">

<sequence>

<element name="Id" type="xswmi:Id"/>

<element name="NEInformation" type="xswmi:NEInformation"/>

<element name="ProfileId" type="xswmi:ProfileId"/>

<element name="StepInfoList" type="xswmi:StepInfoList"/>

</sequence>

</complexType>

<complexType name="swMprocessList">

<sequence>

<element name="swMprocess" type="swMIRPData:swMprocess" maxOccurs="unbounded"/>

</sequence>

</complexType>

<complexType name="startStepName">

<sequence>

<element name="startStepName" type="xswmi:NameOfStep"/>

</sequence>

</complexType>

<complexType name="nEList">

<sequence>

<element name="nEIdentification" type="xswmi:Id"/>

<element name="swFallbackStatus">

<simpleType>

<restriction base="string">

<enumeration value="fallbackSuccessful"/>

<enumeration value="fallbackUnsuccessful"/>

</restriction>

</simpleType>

</element>

<element name="VersionNumber" type="xswmi:VersionNumber"/>

</sequence>

</complexType>

<complexType name="StepsAndSelectedStopPointListEntry">

<sequence>

<element name="nameOfStep" type="xswmi:NameOfStep"/>

<element name="sequenceNumberInProcess" type="xswmi:SequenceNumberInProcess"/>

<element name="stopPointSetIndication" type="xswmi:StopPointSetIndication"/>

</sequence>

</complexType>

<complexType name="StepsAndSelectedStopPointList">

<sequence>

<element name="stepsAndSelectedStopPointListEntry" type="swMIRPData:StepsAndSelectedStopPointListEntry" maxOccurs="unbounded"/>

</sequence>

</complexType>

<complexType name="SelectedFinalAdministrativeState">

<sequence>

<element name="SelectedFinalAdministrativeState" type="xswmi:FinalAdministrativeStateValue" maxOccurs="3"/>

</sequence>

</complexType>

<simpleType name="swLocation">

<restriction base="string"/>

</simpleType>

<simpleType name="swFileSize">

<restriction base="unsignedLong"/>

</simpleType>

<simpleType name="swFileCompression">

<restriction base="string"/>

</simpleType>

<simpleType name="swFileFormat">

<restriction base="string"/>

</simpleType>

<simpleType name="swVersion">

<restriction base="string"/>

</simpleType>

<complexType name="swToBeDownloaded">

<sequence>

<element name="swLocation" type="swMIRPData:swLocation"/>

<element name="swFileSize" type="swMIRPData:swFileSize"/>

<element name="swFileCompression" type="swMIRPData:swFileCompression"/>

<element name="swFileFormat" type="swMIRPData:swFileFormat"/>

</sequence>

</complexType>

<complexType name="swVersionToBeActivated">

<sequence>

<element name="swVersion" type="swMIRPData:swVersion"/>

</sequence>

</complexType>

<complexType name="swTobeInstalled">

<sequence>

<element name="swLocation" type="swMIRPData:swLocation"/>

</sequence>

</complexType>

<!--listSwMCapabilities Request-->

<element name="listSwMCapabilitiesRequest">

<complexType>

<sequence>

<element name="nEInformation" type="xswmi:NEInformation"/>

</sequence>

</complexType>

</element>

<!--listSwMCapabilities Response -->

<element name="listSwMCapabilitiesResponse">

<complexType>

<sequence>

<element name="result">

<simpleType>

<restriction base="string">

<enumeration value="success"/>

<enumeration value="failure"/>

<enumeration value="stepNameNotMatch"/>

<enumeration value="nEInformationIntersection"/>

</restriction>

</simpleType>

</element>

<element name="capabilityList" type="swMIRPData:capabilityList"/>

</sequence>

</complexType>

</element>

<!--listSwMCapabilities Fault -->

<element name="listSwMCapabilitiesFault">

<simpleType>

<restriction base="string">

<enumeration value="operationfailed"/>

</restriction>

</simpleType>

</element>

<!--listSwMProfiles Request -->

<element name="listSwMProfilesRequest">

<complexType>

<sequence>

<element name="nEInformation" type="xswmi:NEInformation"/>

</sequence>

</complexType>

</element>

<!--listSwMProfiles Response -->

<element name="listSwMProfilesResponse">

<complexType>

<sequence>

<element name="swMProfileList" type="swMIRPData:swMProfileList"/>

<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>

<!--createSwMProfile Request -->

<element name="createSwMProfileRequest">

<complexType>

<sequence>

<element name="id" type="xswmi:Id" minOccurs="0"/>

<element name="nEInformation" type="xswmi:NEInformation"/>

<element name="swVersionToBeInstalled" type="swMIRPData:swVersion"/>

<element name="stepsAndSelectedStopPointList" type="swMIRPData:StepsAndSelectedStopPointList"/>

<element name="selectedFinalAdministrativeState" type="swMIRPData:SelectedFinalAdministrativeState"/>

</sequence>

</complexType>

</element>

<!--createSwMProfile Response -->

<element name="createSwMProfileResponse">

<complexType>

<sequence>

<element name="id" type="xswmi: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>

<!--deleteSwMProfile Request -->

<element name="deleteSwMProfileRequest">

<complexType>

<sequence>

<element name="id" type="xswmi:Id"/>

</sequence>

</complexType>

</element>

<!--deleteSwMProfile Response -->

<element name="deleteSwMProfileResponse">

<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>

<!--listSwMProcesses Request -->

<element name="listSwMProcessesRequest">

<complexType>

<sequence>

<element name="nEIdentification" type="xswmi:Id" minOccurs="0"/>

</sequence>

</complexType>

</element>

<!--listSwMProcesses Response -->

<element name="listSwMProcessesResponse">

<complexType>

<sequence>

<element name="swMprocessList" type="swMIRPData:swMprocessList"/>

<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>

<!--resumeSwMProcess Request -->

<element name="resumeSwMProcessRequest">

<complexType>

<sequence>

<element name="id" type="xswmi:Id"/>

<element name="startStepName" type="swMIRPData:startStepName"/>

</sequence>

</complexType>

</element>

<!--resumeSwMProcess Response -->

<element name="resumeSwMProcessResponse">

<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>

<!--swFallback Request -->

<element name="swFallbackRequest">

<complexType>

<sequence>

<element name="filter" type="string"/>

</sequence>

</complexType>

</element>

<!--swFallback Response -->

<element name="swFallbackResponse">

<complexType>

<sequence>

<element name="nEList" type="swMIRPData:nEList"/>

<element name="result">

<simpleType>

<restriction base="string">

<enumeration value="Success"/>

<enumeration value="Partly successful"/>

<enumeration value="Failure"/>

</restriction>

</simpleType>

</element>

</sequence>

</complexType>

</element>

<!--terminateSwMProcess Request -->

<element name="terminateSwMProcessRequest">

<complexType>

<sequence>

<element name="id" type="xswmi:Id"/>

</sequence>

</complexType>

</element>

<!--terminateSwMProcess Response -->

<element name="terminateSwMProcessResponse">

<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>

<!--changeSwMProfile Request -->

<element name="changeSwMProfileRequest">

<complexType>

<sequence>

<element name="id" type="xswmi:Id"/>

<element name="nEInformation" type="xswmi:NEInformation"/>

<element name="swVersionTobeInstalled" type="swMIRPData:swTobeInstalled"/>

<element name="stepsAndSelectedStopPointList" type="swMIRPData:StepsAndSelectedStopPointList"/>

<element name="selectedFinalAdministrativeState" type="swMIRPData:SelectedFinalAdministrativeState"/>

</sequence>

</complexType>

</element>

<!--changeSwMProfile Response -->

<element name="changeSwMProfileResponse">

<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="xswmi: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>

<!--downloadNESw Request -->

<element name="downloadNESwRequest">

<complexType>

<sequence>

<element name="swToBeDownloaded" type="swMIRPData:swToBeDownloaded"/>

<element name="neIdentifier" type="xswmi:Id"/>

</sequence>

</complexType>

</element>

<!--downloadNESw Response -->

<element name="downloadNESwResponse">

<complexType>

<sequence>

<element name="downloadProcessId" type="integer"/>

<element name="listOfStepNumbersAndDurations" type="xswmi:ListOfStepNumbersAndDurations" minOccurs="0"/>

<element name="result">

<simpleType>

<restriction base="string">

<enumeration value="requestAccepted"/>

<enumeration value="requestFailed"/>

<enumeration value="notAllowedBecauseOfOngoingSwmActivity"/>

</restriction>

</simpleType>

</element>

<element name="reason" minOccurs="0">

<simpleType>

<restriction base="string"/>

</simpleType>

</element>

</sequence>

</complexType>

</element>

<!--downloadNESw Fault -->

<element name="downloadNESwFault">

<simpleType>

<restriction base="string">

<enumeration value="operationFailed"/>

<enumeration value="resourceLimitation"/>

</restriction>

</simpleType>

</element>

<!--installNESw Request -->

<element name="installNESwRequest">

<complexType>

<sequence>

<element name="swTobeInstalled" type="swMIRPData:swTobeInstalled"/>

<element name="neIdentifier" type="xswmi:Id"/>

</sequence>

</complexType>

</element>

<!--installNESw Response -->

<element name="installNESwResponse">

<complexType>

<sequence>

<element name="installProcessId" type="integer"/>

<element name="listOfStepNumbersAndDurations" type="xswmi:ListOfStepNumbersAndDurations" minOccurs="0"/>

<element name="result">

<simpleType>

<restriction base="string">

<enumeration value="requestAccepted"/>

<enumeration value="requestFailed"/>

<enumeration value="notAllowedBecauseOfOngoingSwmActivity"/>

</restriction>

</simpleType>

</element>

<element name="reason" minOccurs="0">

<simpleType>

<restriction base="string"/>

</simpleType>

</element>

</sequence>

</complexType>

</element>

<!--installNESw Fault -->

<element name="installNESwFault">

<simpleType>

<restriction base="string">

<enumeration value="operationFailed"/>

<enumeration value="resourceLimitation"/>

<enumeration value="swNotAvailable"/>

</restriction>

</simpleType>

</element>

<!--activateNESw Request -->

<element name="activateNESwRequest">

<complexType>

<sequence>

<element name="swVersionToBeActivated" type="swMIRPData:swVersionToBeActivated"/>

<element name="neIdentifier" type="xswmi:Id"/>

</sequence>

</complexType>

</element>

<!--activateNESw Response -->

<element name="activateNESwResponse">

<complexType>

<sequence>

<element name="activateProcessId" type="integer"/>

<element name="listOfStepNumbersAndDurations" type="xswmi:ListOfStepNumbersAndDurations" minOccurs="0"/>

<element name="result">

<simpleType>

<restriction base="string">

<enumeration value="requestAccepted"/>

<enumeration value="requestFailed"/>

<enumeration value="notAllowedBecauseOfOngoingSwmActivity"/>

</restriction>

</simpleType>

</element>

<element name="reason" minOccurs="0">

<simpleType>

<restriction base="string"/>

</simpleType>

</element>

</sequence>

</complexType>

</element>

<!--activateNESw Fault -->

<element name="activateNESwFault">

<simpleType>

<restriction base="string">

<enumeration value="operationFailed"/>

<enumeration value="resourceLimitation"/>

</restriction>

</simpleType>

</element>

<!--listNaswmProcesses Request -->

<element name="listNaswmProcesses">

<complexType>

<sequence>

<element name="naswmId" type="xswmi:Id" minOccurs="0"/>

<element name="naswmOperationType" type="xswmi:NaswmOperationType"/>

</sequence>

</complexType>

</element>

<!--listNaswmProcesses Response -->

<element name="listNaswmProcessesResponse">

<complexType>

<sequence>

<element name="naswmProcessList" type="xswmi:NaswmProcessList"/>

<element name="result">

<simpleType>

<restriction base="string">

<enumeration value="success"/>

<enumeration value="failure"/>

</restriction>

</simpleType>

</element>

<element name="reason" minOccurs="0">

<simpleType>

<restriction base="string"/>

</simpleType>

</element>

</sequence>

</complexType>

</element>

<!--listNaswmProcesses Fault -->

<element name="listNaswmProcessesFault">

<simpleType>

<restriction base="string">

<enumeration value="operationFailed"/>

<enumeration value="resourceLimitation"/>

</restriction>

</simpleType>

</element>

<!--cancelNaswmProcesses Request -->

<element name="cancelNaswmProcessesRequest">

<complexType>

<sequence>

<element name="naswmId" type="xswmi:Id"/>

<element name="naswmOperationType" type="xswmi:NEaswmOperationType"/>

</sequence>

</complexType>

</element>

<!-- cancel NaswmProcesses Response -->

<element name="cancelNaswmProcessesResponse">

<complexType>

<sequence>

<element name="result">

<simpleType>

<restriction base="string">

<enumeration value="success"/>

<enumeration value="failure"/>

<enumeration value="operationIsAlreadyCompleted"/>

<enumeration value="noSuchProcess"/>

</restriction>

</simpleType>

</element>

<element name="reason" minOccurs="0">

<simpleType>

<restriction base="string"/>

</simpleType>

</element>

</sequence>

</complexType>

</element>

<!-- cancelNaswmProcesses Fault -->

<element name="cancelNaswmProcessesFault">

<simpleType>

<restriction base="string">

<enumeration value="operationFailed"/>

<enumeration value="resourceLimitation"/>

</restriction>

</simpleType>

</element>

</schema>

</types>

<message name="listSwMCapabilitiesRequest">

<part name="parameter" element="swMIRPData:listSwMCapabilitiesRequest"/>

</message>

<message name="listSwMCapabilitiesResponse">

<part name="parameter" element="swMIRPData:listSwMCapabilitiesResponse"/>

</message>

<message name="listSwMCapabilitiesFault">

<part name="parameter" element="swMIRPData:listSwMCapabilitiesFault"/>

</message>

<message name="listSwMProfilesRequest">

<part name="parameter" element="swMIRPData:listSwMProfilesRequest"/>

</message>

<message name="listSwMProfilesResponse">

<part name="parameter" element="swMIRPData:listSwMProfilesResponse"/>

</message>

<message name="createSwMProfileRequest">

<part name="parameter" element="swMIRPData:createSwMProfileRequest"/>

</message>

<message name="createSwMProfileResponse">

<part name="parameter" element="swMIRPData:createSwMProfileResponse"/>

</message>

<message name="deleteSwMProfileRequest">

<part name="parameter" element="swMIRPData:deleteSwMProfileRequest"/>

</message>

<message name="deleteSwMProfileResponse">

<part name="parameter" element="swMIRPData:deleteSwMProfileResponse"/>

</message>

<message name="listSwMProcessesRequest">

<part name="parameter" element="swMIRPData:listSwMProcessesRequest"/>

</message>

<message name="listSwMProcessesResponse">

<part name="parameter" element="swMIRPData:listSwMProcessesResponse"/>

</message>

<message name="resumeSwMProcessRequest">

<part name="parameter" element="swMIRPData:resumeSwMProcessRequest"/>

</message>

<message name="resumeSwMProcessResponse">

<part name="parameter" element="swMIRPData:resumeSwMProcessResponse"/>

</message>

<message name="swFallbackRequest">

<part name="parameter" element="swMIRPData:swFallbackRequest"/>

</message>

<message name="swFallbackResponse">

<part name="parameter" element="swMIRPData:swFallbackResponse"/>

</message>

<message name="terminateSwMProcessRequest">

<part name="parameter" element="swMIRPData:terminateSwMProcessRequest"/>

</message>

<message name="terminateSwMProcessResponse">

<part name="parameter" element="swMIRPData:terminateSwMProcessResponse"/>

</message>

<message name="changeSwMProfileRequest">

<part name="parameter" element="swMIRPData:changeSwMProfileRequest"/>

</message>

<message name="changeSwMProfileResponse">

<part name="parameter" element="swMIRPData:changeSwMProfileResponse"/>

</message>

<message name="downloadNESwRequest">

<part name="parameter" element="swMIRPData:downloadNESwRequest"/>

</message>

<message name="downloadNESwResponse">

<part name="parameter" element="swMIRPData:downloadNESwResponse"/>

</message>

<message name="downloadNESwFault">

<part name="parameter" element="swMIRPData:downloadNESwFault"/>

</message>

<message name="installNESwRequest">

<part name="parameter" element="swMIRPData:installNESwRequest"/>

</message>

<message name="installNESwResponse">

<part name="parameter" element="swMIRPData:installNESwResponse"/>

</message>

<message name="installNESwFault">

<part name="parameter" element="swMIRPData:installNESwFault"/>

</message>

<message name="activateNESwRequest">

<part name="parameter" element="swMIRPData:activateNESwRequest"/>

</message>

<message name="activateNESwResponse">

<part name="parameter" element="swMIRPData:activateNESwResponse"/>

</message>

<message name="activateNESwFault">

<part name="parameter" element="swMIRPData:activateNESwFault"/>

</message>

<message name="listNaswmProcessesResponse ">

<part name="parameter" element="swMIRPData:listNaswmProcessesResponse"/>

</message>

<message name="listNaswmProcessesResponse">

<part name="parameter" element="swMIRPData:listNaswmProcessesResponse"/>

</message>

<message name="listNaswmProcessesFault">

<part name="parameter" element="swMIRPData:listNaswmProcessesFault"/>

</message>

<message name="cancelNaswmProcesses">

<part name="parameter" element="swMIRPData:cancelNaswmProcessesResponse"/>

</message>

<message name="cancelNaswmProcessesResponse">

<part name="parameter" element="swMIRPData:cancelNaswmProcessesResponse"/>

</message>

<message name="cancelNaswmProcessesFault">

<part name="parameter" element="swMIRPData:cancelNaswmProcesses Fault"/>

</message>

<portType name="SwMIRPOperation1">

<operation name="listSwMCapabilities">

<input message="swMIRPSystem:listSwMCapabilitiesRequest"/>

<output message="swMIRPSystem:listSwMCapabilitiesResponse"/>

<fault name="listSwMCapabilitiesFault" message="swMIRPSystem:listSwMCapabilitiesFault"/>

</operation>

<operation name="listSwMProfiles">

<input message="swMIRPSystem:listSwMProfilesRequest"/>

<output message="swMIRPSystem:listSwMProfilesResponse"/>

</operation>

<operation name="createSwMProfile">

<input message="swMIRPSystem:createSwMProfileRequest"/>

<output message="swMIRPSystem:createSwMProfileResponse"/>

</operation>

<operation name="deleteSwMProfile">

<input message="swMIRPSystem:deleteSwMProfileRequest"/>

<output message="swMIRPSystem:deleteSwMProfileResponse"/>

</operation>

<operation name="listSwMProcesses">

<input message="swMIRPSystem:listSwMProcessesRequest"/>

<output message="swMIRPSystem:listSwMProcessesResponse"/>

</operation>

<operation name="resumeSwMProcess">

<input message="swMIRPSystem:resumeSwMProcessRequest"/>

<output message="swMIRPSystem:resumeSwMProcessResponse"/>

</operation>

<operation name="swFallback">

<input message="swMIRPSystem:swFallbackRequest"/>

<output message="swMIRPSystem:swFallbackResponse"/>

</operation>

<operation name="terminateSwMProcess">

<input message="swMIRPSystem:terminateSwMProcessRequest"/>

<output message="swMIRPSystem:terminateSwMProcessResponse"/>

</operation>

</portType>

<portType name="SwMIRPOperation2">

<operation name="changeSwMProfile">

<input message="swMIRPSystem:changeSwMProfileRequest"/>

<output message="swMIRPSystem:changeSwMProfileResponse"/>

</operation>

</portType>

<portType name="SwMIRPOperation3">

<operation name="downloadNESw">

<input message="swMIRPSystem:downloadNESwRequest"/>

<output message="swMIRPSystem:downloadNESwResponse"/>

<fault name="downloadNESwFault" message="swMIRPSystem:downloadNESwFault"/>

</operation>

<operation name="activateNESw">

<input message="swMIRPSystem:activateNESwRequest"/>

<output message="swMIRPSystem:activateNESwResponse"/>

<fault name="activateNESwFault" message="swMIRPSystem:activateNESwFault"/>

</operation>

</portType>

<portType name="SwMIRPOperation4">

<operation name="installNESw">

<input message="swMIRPSystem:installNESwRequest"/>

<output message="swMIRPSystem:installNESwResponse"/>

<fault name="installNESwFault" message="swMIRPSystem:installNESwFault"/>

</operation>

</portType>

<binding name="SwMIRPOperation1" type="swMIRPSystem:SwMIRPOperation1">

<soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>

<operation name="listSwMCapabilities">

<soap:operation soapAction="http://www.3gpp.org/ftp/specs/archive/32\_series/32.536#listSwMCapabilities" style="document"/>

<input>

<soap:body use="literal"/>

</input>

<output>

<soap:body use="literal"/>

</output>

<fault name="listSwMCapabilitiesFault"/>

</operation>

<operation name="listSwMProfiles">

<soap:operation soapAction="http://www.3gpp.org/ftp/specs/archive/32\_series/32.536#listSwMProfiles" style="document"/>

<input>

<soap:body use="literal"/>

</input>

<output>

<soap:body use="literal"/>

</output>

</operation>

<operation name="createSwMProfile">

<soap:operation soapAction="http://www.3gpp.org/ftp/specs/archive/32\_series/32.536#createSwMProfile" style="document"/>

<input>

<soap:body use="literal"/>

</input>

<output>

<soap:body use="literal"/>

</output>

</operation>

<operation name="deleteSwMProfile">

<soap:operation soapAction="http://www.3gpp.org/ftp/specs/archive/32\_series/32.536#deleteSwMProfile" style="document"/>

<input>

<soap:body use="literal"/>

</input>

<output>

<soap:body use="literal"/>

</output>

</operation>

<operation name="listSwMProcesses">

<soap:operation soapAction="http://www.3gpp.org/ftp/specs/archive/32\_series/32.536#listSwMProcesses" style="document"/>

<input>

<soap:body use="literal"/>

</input>

<output>

<soap:body use="literal"/>

</output>

</operation>

<operation name="resumeSwMProcess">

<soap:operation soapAction="http://www.3gpp.org/ftp/specs/archive/32\_series/32.536#resumeSwMProcess" style="document"/>

<input>

<soap:body use="literal"/>

</input>

<output>

<soap:body use="literal"/>

</output>

</operation>

<operation name="swFallback">

<soap:operation soapAction="http://www.3gpp.org/ftp/specs/archive/32\_series/32.536#swFallback" style="document"/>

<input>

<soap:body use="literal"/>

</input>

<output>

<soap:body use="literal"/>

</output>

</operation>

<operation name="terminateSwMProcess">

<soap:operation soapAction="http://www.3gpp.org/ftp/specs/archive/32\_series/32.536#terminateSwMProcess" style="document"/>

<input>

<soap:body use="literal"/>

</input>

<output>

<soap:body use="literal"/>

</output>

</operation>

</binding>

<binding name="SwMIRPOperation2" type="swMIRPSystem:SwMIRPOperation2">

<soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>

<operation name="changeSwMProfile">

<soap:operation soapAction="http://www.3gpp.org/ftp/specs/archive/32\_series/32.536#changeSwMProfile" style="document"/>

<input>

<soap:body use="literal"/>

</input>

<output>

<soap:body use="literal"/>

</output>

</operation>

</binding>

<binding name="SwMIRPOperation3" type="swMIRPSystem:SwMIRPOperation3">

<soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>

<operation name="downloadNESw">

<soap:operation soapAction="http://www.3gpp.org/ftp/specs/archive/32\_series/32.536#downloadNESw" style="document"/>

<input>

<soap:body use="literal"/>

</input>

<output>

<soap:body use="literal"/>

</output>

<fault name="downloadNESwFault"/>

</operation>

<operation name="activateNESw">

<soap:operation soapAction="http://www.3gpp.org/ftp/specs/archive/32\_series/32.536#activateNESw" style="document"/>

<input>

<soap:body use="literal"/>

</input>

<output>

<soap:body use="literal"/>

</output>

<fault name="activateNESwFault"/>

</operation>

</binding>

<binding name="SwMIRPOperation4" type="swMIRPSystem:SwMIRPOperation4">

<soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>

<operation name="installNESw">

<soap:operation soapAction="http://www.3gpp.org/ftp/specs/archive/32\_series/32.536#installNESw" style="document"/>

<input>

<soap:body use="literal"/>

</input>

<output>

<soap:body use="literal"/>

</output>

<fault name="installNESwFault"/>

</operation>

</binding>

<binding name="SwMIRPOperation5" type="swMIRPSystem:SwMIRPOperation5">

<soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>

<operation name="listNaswmProcesses">

<soap:operation soapAction="http://www.3gpp.org/ftp/specs/archive/32\_series/32.536#listNaswmProcesses" style="document"/>

<input>

<soap:body use="literal"/>

</input>

<output>

<soap:body use="literal"/>

</output>

<fault name="listNaswmProcessesFault"/>

</operation>

</binding>

<binding name="SwMIRPOperation6" type="swMIRPSystem:SwMIRPOperation6">

<soap:binding style="document" transport="http://schemas.xmlsoap.org/soap/http"/>

<operation name="cancelNaswmProcesses">

<soap:operation soapAction="http://www.3gpp.org/ftp/specs/archive/32\_series/32.536#cancelNaswmProcesses" style="document"/>

<input>

<soap:body use="literal"/>

</input>

<output>

<soap:body use="literal"/>

</output>

<fault name="cancelNaswmProcesses Fault"/>

</operation>

</binding>

<service name="SwMIRPService">

<port name="SwMIRPOperation1" binding="swMIRPSystem:SwMIRPOperation1">

<soap:address location="http://www.3gpp.org/ftp/specs/archive/32\_series/32.536#SwMIRP"/>

</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="SwMIRPOperation2" binding="swMIRPSystem:SwMIRPOperation2">

<soap:address location="http://www.3gpp.org/ftp/specs/archive/32\_series/32.536#SwMIRP"/>

</port>

<port name="SwMIRPOperation3" binding="swMIRPSystem:SwMIRPOperation3">

<soap:address location="http://www.3gpp.org/ftp/specs/archive/32\_series/32.536#SwMIRP"/>

</port>

<port name="SwMIRPOperation4" binding="swMIRPSystem:SwMIRPOperation4">

<soap:address location="http://www.3gpp.org/ftp/specs/archive/32\_series/32.536#SwMIRP"/>

</port>

<port name="SwMIRPOperation5" binding="swMIRPSystem:SwMIRPOperation5">

<soap:address location="http://www.3gpp.org/ftp/specs/archive/32\_series/32.536#SwMIRP"/>

</port>

<port name="SwMIRPOperation6" binding="swMIRPSystem:SwMIRPOperation6">

<soap:address location="http://www.3gpp.org/ftp/specs/archive/32\_series/32.536#SwMIRP"/>

</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-100511 | -- | -- | Presentation to SA for Information and Approval | --- | 1.0.0 |
| 2010-10 | -- | -- | -- | -- | Publication | 1.0.0 | 10.0.0 |
| 2010-12 | SA#50 | SP-10833 | 002 | 1 | Modify the input parameters of notifyActivateNESwStatusChanged - Align with 32.532 IS | 10.0.0 | 10.1.0 |
| 2010-12 | SA#50 | SP-10833 | 003 | 1 | Add NE health check step in automatic software management – Align with 32.532 IS | 10.0.0 | 10.1.0 |
| 2010-12 | SA#50 | SP-10831 | 004 | -- | Align support qualifier of notifyNewSwAvailability with 32.532 | 10.0.0 | 10.1.0 |
| 2012-09 | SA#57 | SP-120645 | 009 | 1 | Addition of progress reporting and cancellation of Non-Automated Software Management operations | 10.1.0 | 11.0.0 |
| 2014-09 | SA#65 | SP-140559 | 010 | - | Update the link from Solution Set to Information Service due to the end of Release 12 | 11.0.0 | 12.0.0 |
| 2016-01 | - | - | - | - | Update to Rel-13 version (MCC) | 12.0.0 | **13.0.0** |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Change history** | | | | | | | |
| **Date** | **Meeting** | **TDoc** | **CR** | **Rev** | **Cat** | **Subject/Comment** | **New version** |
| 2016-06 | SA#72 | SP-160407 | 0011 | - | F | Update the link from IRP Solution Set to IRP Information Service | 13.1.0 |
| 2017-04 | SA#75 | - | - | - |  | Promotion to Release 14 without technical change | **14.0.0** |
| 2017-06 | SA#76 | SP-170502 | 0012 | - | F | Update the link from IRP Solution Set to IRP Information Service | **14.1.0** |
| 2018-06 | - | - | - | - | - | Update to Rel-15 version (MCC) | **15.0.0** |
| 2020-07 | - | - | - | - | - | Update to Rel-16 version (MCC) | **16.0.0** |