

ETSI TS 128 317 V18.0.0 (2024-05)



**5G;
Management and orchestration;
Self-configuration of
Radio Access Network Entities (RAN NEs)
(3GPP TS 28.317 version 18.0.0 Release 18)**



Reference

DTS/TSGS-0528317vi00

Keywords

5G

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

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

Siret N° 348 623 562 00017 - APE 7112B
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° w061004871

Important notice

The present document can be downloaded from:

<https://www.etsi.org/standards-search>

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be modified without the prior written authorization of ETSI. In case of any existing or perceived difference in contents between such versions and/or in print, the prevailing version of an ETSI deliverable is the one made publicly available in PDF format at www.etsi.org/deliver.

Users of the present document should be aware that the document may be subject to revision or change of status.

Information on the current status of this and other ETSI documents is available at

<https://portal.etsi.org/TB/ETSIDeliverableStatus.aspx>

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

<https://portal.etsi.org/People/CommitteeSupportStaff.aspx>

If you find a security vulnerability in the present document, please report it through our
Coordinated Vulnerability Disclosure Program:

<https://www.etsi.org/standards/coordinated-vulnerability-disclosure>

Notice of disclaimer & limitation of liability

The information provided in the present deliverable is directed solely to professionals who have the appropriate degree of experience to understand and interpret its content in accordance with generally accepted engineering or other professional standard and applicable regulations.

No recommendation as to products and services or vendors is made or should be implied.

No representation or warranty is made that this deliverable is technically accurate or sufficient or conforms to any law and/or governmental rule and/or regulation and further, no representation or warranty is made of merchantability or fitness for any particular purpose or against infringement of intellectual property rights.

In no event shall ETSI be held liable for loss of profits or any other incidental or consequential damages.

Any software contained in this deliverable is provided "AS IS" with no warranties, express or implied, including but not limited to, the warranties of merchantability, fitness for a particular purpose and non-infringement of intellectual property rights and ETSI shall not be held liable in any event for any damages whatsoever (including, without limitation, damages for loss of profits, business interruption, loss of information, or any other pecuniary loss) arising out of or related to the use of or inability to use the software.

Copyright Notification

No part may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm except as authorized by written permission of ETSI.

The content of the PDF version shall not be modified without the written authorization of ETSI.

The copyright and the foregoing restriction extend to reproduction in all media.

© ETSI 2024.
All rights reserved.

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The declarations pertaining to these essential IPRs, if any, are publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: "*Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards*", which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (<https://ipr.etsi.org/>).

Pursuant to the ETSI Directives including the ETSI IPR Policy, no investigation regarding the essentiality of IPRs, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

DECT™, **PLUGTESTS™**, **UMTS™** and the ETSI logo are trademarks of ETSI registered for the benefit of its Members. **3GPP™** and **LTE™** are trademarks of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners. **oneM2M™** logo is a trademark of ETSI registered for the benefit of its Members and of the oneM2M Partners. **GSM®** and the GSM logo are trademarks registered and owned by the GSM Association.

Legal Notice

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

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

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

Modal verbs terminology

In the present document "**shall**", "**shall not**", "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the [ETSI Drafting Rules](#) (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

Contents

Intellectual Property Rights	2
Legal Notice	2
Modal verbs terminology.....	2
Foreword.....	5
1 Scope	7
2 References	7
3 Definitions of terms, symbols and abbreviations	7
3.1 Terms.....	7
3.2 Symbols.....	8
3.3 Abbreviations	8
4 Concept and Background	8
4.1 Background	8
4.2 Network configuration data handling	8
4.3 Self-configuration.....	8
4.4 Self-configuration management	9
5 Management capabilities	9
5.1 Network configuration data handling	9
5.1.1 Use cases.....	9
5.1.2 Requirements	9
5.2 Self-configuration management	10
5.2.1 Use cases.....	10
5.2.2 Requirements	10
6 Stage 2 definition	11
6.1 Management operation for Self-configuration management (MnS component typeA)	11
6.2 Information model definition for Self-configuration management (MnS component typeB).....	11
6.2.1 Imported information entities and local labels.....	11
6.2.2 Class diagram.....	11
6.2.2.1 Relationship	11
6.2.2.2 Inheritance.....	12
6.2.3 Class definition	12
6.2.3.1 ScMgmtProfile.....	12
6.2.3.1.1 Definition.....	12
6.2.3.1.2 Attributes	12
6.2.3.1.3 Attributes constraints	13
6.2.3.1.4 Notifications	13
6.2.3.2 Sc_Process.....	13
6.2.3.2.1 Definition.....	13
6.2.3.2.2 Attributes	14
6.2.3.2.3 Attributes constraints	14
6.2.3.2.4 Notifications	14
6.2.4 Attribute definition	14
6.2.5 Common notifications.....	16
6.2.5.1 Configuration notifications	16
7 Stage 3 definition	16
7.1 RESTful HTTP-based solution set	16
7.2 OpenAPI specification	16
7.2.1 OpenAPI document "TS28532_ProvMnS.yaml"	16
7.2.2 OpenAPI document for RANSC NRM.....	16
8 Procedure for Self-establishment	17
8.1 Self-configuration management	17

Annex A (informative): PlantUML source code19

A.1 Procedure for Self-establishment 19

A.1.1 Procedure for start self-configuration management 19

A.2 Information model definition for RANSC management20

A.2.1 Relationship UML diagram 20

A.2.2 Inheritance UML diagram 20

Annex B (informative): Change history21

History22

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.

In the present document, modal verbs have the following meanings:

- shall** indicates a mandatory requirement to do something
- shall not** indicates an interdiction (prohibition) to do something

The constructions "shall" and "shall not" are confined to the context of normative provisions, and do not appear in Technical Reports.

The constructions "must" and "must not" are not used as substitutes for "shall" and "shall not". Their use is avoided insofar as possible, and they are not used in a normative context except in a direct citation from an external, referenced, non-3GPP document, or so as to maintain continuity of style when extending or modifying the provisions of such a referenced document.

- should** indicates a recommendation to do something
- should not** indicates a recommendation not to do something
- may** indicates permission to do something
- need not** indicates permission not to do something

The construction "may not" is ambiguous and is not used in normative elements. The unambiguous constructions "might not" or "shall not" are used instead, depending upon the meaning intended.

- can** indicates that something is possible
- cannot** indicates that something is impossible

The constructions "can" and "cannot" are not substitutes for "may" and "need not".

- will** indicates that something is certain or expected to happen as a result of action taken by an agency the behaviour of which is outside the scope of the present document
- will not** indicates that something is certain or expected not to happen as a result of action taken by an agency the behaviour of which is outside the scope of the present document
- might** indicates a likelihood that something will happen as a result of action taken by some agency the behaviour of which is outside the scope of the present document

might not indicates a likelihood that something will not happen as a result of action taken by some agency the behaviour of which is outside the scope of the present document

In addition:

is (or any other verb in the indicative mood) indicates a statement of fact

is not (or any other negative verb in the indicative mood) indicates a statement of fact

The constructions "is" and "is not" do not indicate requirements.

1 Scope

The present document describes the functionality for RAN NE self-configuration and ARCF data handling to enable RAN NE self-establishment, as well as requirements need to be met to support this functionality. The concepts, use cases, requirements, procedure and management service definition for RAN NE self-configuration management and ARCF data handling of RAN NEs are specified in the present document.

The NE within virtualization is not in the scope of the present document.

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 28.314: "Management and orchestration; Plug and Connect; Concepts and requirements".
- [3] 3GPP TS 28.532: "Management and orchestration; Generic management services".
- [4] 3GPP TS 28.622: "Telecommunication management; Generic Network Resource Model (NRM); Integration Reference Point (IRP); Information Service (IS)".
- [5] Management and Orchestration APIs Stage3 repository, "https://forge.3gpp.org/rep/sa5/MnS/-/tree/Tag_Rel18_SA103/".

3 Definitions of terms, symbols and abbreviations

3.1 Terms

For the purposes of the present document, the terms given in 3GPP 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 3GPP TR 21.905 [1].

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

3.2 Symbols

Void.

3.3 Abbreviations

For the purposes of the present document, the abbreviations given in 3GPP TR 21.905 [1] and the following apply. An abbreviation defined in the present document takes precedence over the definition of the same abbreviation, if any, in 3GPP TR 21.905 [1].

RANSC	RAN NE Self-Configuration
-------	---------------------------

4 Concept and Background

4.1 Background

Establishment of a new RAN NE in network by an autonomous way can greatly improve the efficiency of RAN deployment. It refers to the procedure of a new RAN NE automatically establishing when it is powered up and connects to the IP network, which includes:

- Network configuration data handling
- Plug and connect to management system
- Self-Configuration

Plug and connect to management system has been specified in TS 28.314 [2]. This document mainly focuses on Network configuration data handling and Self-Configuration.

4.2 Network configuration data handling

Network configuration data handling refers to the procedures to make the network configuration data (i.e. additional data used to generate the complete RAN NE configuration data) available to the RANSC MnS producer, which may include network configuration data preparation, network configuration data transfer and network configuration data validation. This happens except all of the network configuration data for RAN NE can be generated by the RANSC MnS producer.

Network configuration data preparation: This makes the network configuration data ready in operator's network management system who provides the network configuration data. How to prepare the network configuration data in operator's network management system is out of scope of the present document.

Network configuration data transfer: This transfers the network configuration data from operator's network management system to the RANSC MnS producer.

Network configuration data validation: This validates the syntax and semantics of network configuration data. It takes place in the RANSC MnS producer.

Note: The process of network configuration data preparation and network configuration data validation is not in the scope of the present document.

4.3 Self-configuration

RAN NE can be taken to a state ready to carry traffic using self-configuration in an automated manner. Self-configuration may include following processes: generate the RAN NE configuration data, download and activate software, download and active configuration data, self-test and update network resource model, etc.

4.4 Self-configuration management

Self-configuration management capability is needed to monitor the self-configuration process and provide the operator with this information (e.g. progress information). In addition, it allows the operator to control the execution of the self-configuration process.

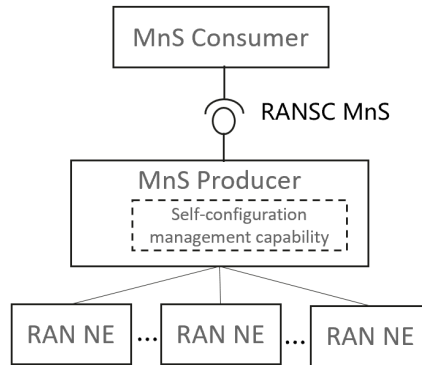


Figure 4.4-1: Example of deployment scenario for Self-configuration management

5 Management capabilities

5.1 Network configuration data handling

5.1.1 Use cases

The goal of this use case is to ensure the RAN NE configuration data available for self-configuration process.

The RANSC MnS producer may need to obtain the additional network configuration data (e.g. nRTAC, gNBId, PCIList) from RANSC MnS consumer and generate the completed RAN NE configuration data to RAN NE. The RANSC MnS consumer will provide this additional network configuration data to RANSC MnS producer directly or indicate it where the configuration data is available and it can retrieve the network configuration data from there.

When RANSC MnS producer receive the additional network configuration data, it may perform network configuration data validation to validate the syntax and semantics of network configuration data.

5.1.2 Requirements

Table 5.1.2-1

Requirement label	Description	Related use case(s)/Motivation
REQ-RANSC_DataHandling-1	The RANSC MnS producer should have the capability to obtain the information which indicate the location where the network configuration data is available for downloading.	Network configuration data handling

5.2 Self-configuration management

5.2.1 Use cases

Self-configuration management capability is performed by RANSC MnS producer. RANSC MnS producer monitors the self-configuration process and provides the RANSC MnS consumer with this information (e.g. progress information). In addition, RANSC MnS producer also allows the RANSC MnS consumer to control the execution of the self-configuration process.

The RANSC MnS consumer can request RANSC MnS producer to create and activate a self-configuration process for certain RAN NE in the case of the RANSC MnS consumer trigger the self-configuration process. Besides, RANSC MnS producer also can create and activate a self-configuration process triggered by itself based on the self-configuration management profile (representing the decision of RANSC MnS consumer) configured by RANSC MnS consumer.

As the self-configuration process is complex and time-consuming, the RANSC MnS consumer needs to obtain the progress of the self-configuration process. Self-configuration process includes several steps (each step can represent one or several of activities) according to the self-configuration capabilities. The authorized RANSC MnS consumer may want to be informed the important events for step transition (e.g. start to execute a new step) and abnormal events during the self-configuration process. The RANSC MnS consumer also wants to be informed the reasons when abnormal event (e.g. failure) occurred.

During the self-configuration, the RANSC MnS consumers can send request to RANSC MnS producer to query the list of ongoing self-configuration process or the progress for certain self-configuration process.

When the last step of the self-configuration process is completed, RANSC MnS producer needs to send the result of this process to the RANSC MnS consumers. The authorized RANSC MnS consumers can terminate an ongoing self-configuration process for failure analysis and solving. When the last step of the self-configuration process is completed successfully, the RANSC MnS producer can delete the self-configuration process automatically.

5.2.2 Requirements

Table 5.2.2-1

Requirement label	Description	Related use case(s)/Motivation
REQ-RANSC_Mgmt-1	The RANSC MnS producer shall have the capability to allow the authorized MnS consumers to request to create and activate a self-configuration process.	Self-configuration management
REQ-RANSC_Mgmt-2	The RANSC MnS producer shall have the capability to allow the authorized MnS consumers to request to query the list of ongoing Self-configuration processes.	Self-configuration management
REQ-RANSC_Mgmt-3	The RANSC MnS producer shall have the capability to report the step information of a self-configuration process to the authorized MnS consumers.	Self-configuration management
REQ-RANSC_Mgmt-4	The RANSC MnS producer should have the capability to report abnormal information to the authorized MnS consumers when detected.	Self-configuration management
REQ-RANSC_Mgmt-5	The RANSC MnS producer shall have the capability to inform the authorized MnS consumers the result (success or failure) of the self-configuration process when the process is finished.	Self-configuration management
REQ-RANSC_Mgmt-6	The RANSC MnS producer shall have the capability to allow the authorized MnS consumers to request to query the progress (e.g. step information) of the self-configuration process when needed.	Self-configuration management
REQ-RANSC_Mgmt-7	The RANSC MnS producer shall have the capability to allow the authorized MnS consumers to request to terminate an ongoing self-configuration process.	Self-configuration management
REQ-RANSC_Mgmt-8	The RANSC MnS producer should have the capability to inform the authorized MnS consumers the information that it has deleted the self-configuration process automatically.	Self-configuration management

6 Stage 2 definition

6.1 Management operation for Self-configuration management (MnS component typeA)

The operations (e.g. createMOI operations) and notifications (e.g. notifyMOIcreation) of generic provisioning MnS defined in 3GPP TS 28.532 [3] can be used for Self-configuration management. The ScMgmtProfile and Sc_Process can be treated as object instance. Following is the IS to support intent lifecycle management.

Table 6.1-1

Self-configuration management	IS operation
Create an ScMgmtProfile	createMOI operation
Delete an ScMgmtProfile	deleteMOI operation
Modify an ScMgmtProfile	modifyMOIAttributes operation
Query an ScMgmtProfile	getMOIAttributes operation
Query an Sc_Process	getMOIAttributes operation
Cancel an Sc_Process	modifyMOIAttributes operation

6.2 Information model definition for Self-configuration management (MnS component typeB)

6.2.1 Imported information entities and local labels

Table 6.2.1-1

Label reference	Local label
TS 28.622 [4], IOC, Top	Top

6.2.2 Class diagram

6.2.2.1 Relationship

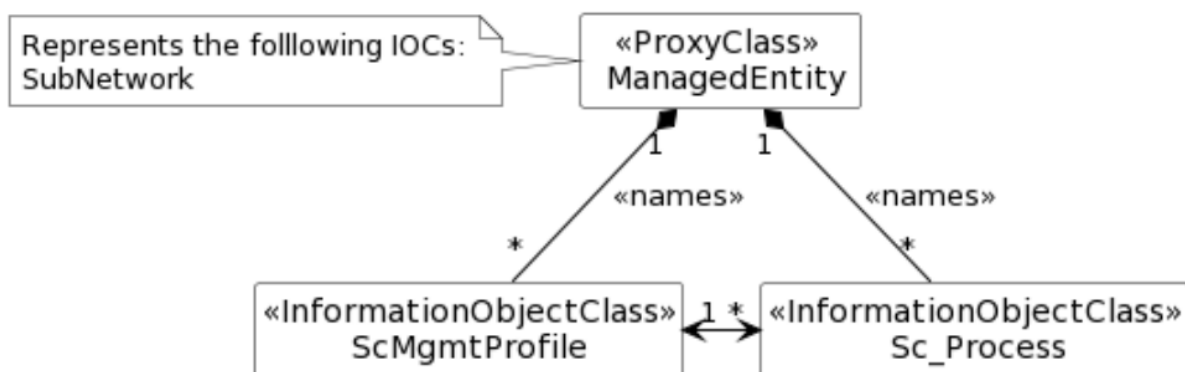


Figure 6.2.2.1-1: Relationship UML diagram

6.2.2.2 Inheritance

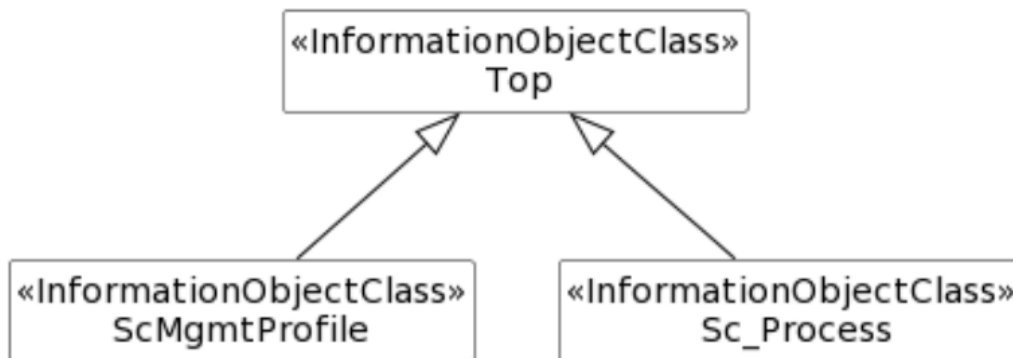


Figure 6.2.2.2-1: Inheritance UML diagram

6.2.3 Class definition

6.2.3.1 ScMgmtProfile

6.2.3.1.1 Definition

The ScMgmtProfile represents MnS Consumer's requirements for self-configuration management for a set of RAN NEs or RAN NE types. It can be name-contained by IOC SubNetwork.

A ScMgmtProfile is created by a MnS consumer to request MnS producer to manage the self-configuration processes for a set of RAN NEs or RAN NE types. The creation request contains the information required by the MnS consumer to start self-configuration processes for a set of RAN NEs or RAN NE types (specified by nEInformation). For ultimate deactivation of requirements for self-configuration management for a set of RAN NEs or NE types, the MnS consumer should request MnS producer to delete the ScMgmtProfile to free up resources.

Attribute nEInformation specifies the RAN NEs or RAN NE types for which this ScMgmtProfile instance is valid. For a RAN NE starting its Self-configuration process, there shall be no ambiguity which ScMgmtProfile is valid for the RAN NE. Therefore, the attribute nEInformation of different ScMgmtProfile instances shall not intersect.

Attribute configDataFileLocation specifies the address where the files of network configuration data can be retrieved for the specified RAN NEs.

The ScMgmtProfile IOC includes the attribute objectClass and objectInstance from the TOP IOC. The value of attribute objectClass is ScMgmtProfile and the value of attribute objectInstance is the DN of the instance of ScMgmtProfile IOC.

6.2.3.1.2 Attributes

The ScMgmtProfile IOC includes attributes inherited from Top IOC (defined in TS 28.622 [4]) and the following attributes:

Table 6.2.3.1.2-1

Attribute Name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifiable
nEInformation	M	T	T	F	T
configDataFileLocation	O	T	T	F	T

6.2.3.1.3 Attributes constraints

None.

6.2.3.1.4 Notifications

The common notifications defined in clause 6.2.5.1 are valid for this IOC, without exceptions or additions.

6.2.3.2 Sc_Process

6.2.3.2.1 Definition

This IOC represents the self-configuration process for a RAN NE, which allows the MnS Consumer to be informed about the current situation of the Self Configuration process.

When the automated management process for an RAN NE starts, an instance of the Sc_Process is created automatically by the MnS Producer and informed to MnS consumer.

The ScProcessMonitor attribute represents the status of self configuration process and includes information the MnS consumer can use to monitor the progress and result of the self configuration process. The data type of this attribute is ProcessMonitor in TS 28.622 [4]. The following specializations are provided for this data type for the self configuration process:

- The value of attribute status are "NOT_STARTED", "RUNNING", "CANCELLING", "FINISHED", "FAILED" and "CANCELLED". The values "SUSPENDED" and "PARTIALLY_FAILED" are not used.
- The timer attribute is not used.
- The attribute progressPercentage indicates progress of the process as percentage. The percent can be measured by number of finished steps from total steps in the self configuration process.
- When the attribute status is equal to "RUNNING" the attribute progressStateInfo attribute shall indicate one of the following states: "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". Vendor specific information may be provided though.
- For the case that the attribute status is equal to "FAILED" the attribute resultStateInfo shall indicate one of the following failure reasons: "UNKNOWN", "INCORRECT_CONFIGURATION", "NE_HARDWARE_ERROR_DETECTED", "DISCONNECTION_BETWEEN_NE_AND_OAM", "OTHER". Vendor specific information may be provided though.
- For the case that the attribute status is equal to "FINISHED" or "CANCELLED", no specializations are provided for the attribute resultStateInfo. Vendor specific information may be provided though.
- No specializations are provided for the attribute startTime and endTime.

MnS consumer can terminate a self-configuration process which is currently ongoing (the value of the attribute status is "RUNING") by configuring attribute cancelScProcess to "True". Then the attribute status will change from "RUNING" to "CANCELLED" when MnS producer have terminated the self-configuration process as MnS consumer requested.

The MnS producer can delete the Sc_Process instance whose attribute status equals to "FINISHED" or "FAILED" or "CANCELLED" automatically.

MnS consumer can query different attributes of the Sc_Process instance or subscribe attribute value change notifications for Sc_Process instance to obtain corresponding progress and result information for a self-configuration process when the Sc_Process instance is created by the MnS Producer and informed to MnS consumer.

The Sc_Process IOC includes the attribute objectClass and objectInstance from the TOP IOC. The value of attribute objectClass is "Sc_Process" and the value of attribute objectInstance is the DN of the instance of Sc_Process IOC.

6.2.3.2.2 Attributes

The `Sc_Process` IOC includes attributes inherited from Top IOC (defined in TS 28.622 [4]) and the following attributes:

Table 6.2.3.2.2-1

Attribute name	Support Qualifier	isReadable	isWritable	isInvariant	isNotifiable
<code>nEIdentification</code>	M	T	F	F	T
<code>scProcessMonitor</code>	M	T	F	F	T
<code>cancelScProcess</code>	M	T	T	F	T
Attributes related to roles					
<code>ScMgmtProfileRef</code>	M	T	F	F	T

6.2.3.2.3 Attributes constraints

6.2.3.2.4 Notifications

The common notifications defined in clause 6.2.5.1 are valid for this IOC, without exceptions or additions.

6.2.4 Attribute definition

The following table defines the properties of attributes specified in the present document.

Table 6.2.4-1 Attribute definition

Attribute Name	Documentation and Allowed Values	Properties
<code>nEInformation</code>	This attribute defines the NE Type(s) or the NE instance(s) for which this <code>ScMgmtProfile</code> instance is valid.	type: String multiplicity: * isOrdered: False isUnique: True defaultValue: None allowedValues: N/A isNullable: False
<code>configDataFileLocation</code>	This attribute specifies the address where the files of network configuration data can be retrieved. allowedValues: File URI	type: String multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None allowedValues: N/A isNullable: False
<code>cancelScProcess</code>	Setting this attribute to "TRUE" cancels the self configuration process. Cancellation is possible in the "NOT_STARTED" and "RUNNING" state. Setting the attribute to "FALSE" has no observable result. allowedValues: TRUE, FALSE	Type: ENUM multiplicity: 0..1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: False
<code>nEIdentification</code>	This attribute identifies the NE for which the <code>Sc_Process</code> instance is done. Note: <code>nEIdentification</code> should be identity of RAN NEs.	type: String multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None allowedValues: N/A isNullable: False

Attribute Name	Documentation and Allowed Values	Properties
ScProcessMonitor	This attribute provides monitoring for the self configuration process. The data type of this attribute is the "ProcessMonitor" as defined in clause 4.3.43 in TS 28.622 [4] with the specialisations defined in clause 6.2.3.2.1. allowedValues: N/A	Type: ProcessMonitor multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: False
ScProcessMonitor.status	This attribute represents the status of the associated self-configuration process, whether it fails (represented by FAILED), succeeds (represented by FINISHED) etc. allowedValues: - NOT_STARTED - RUNNING - CANCELLING - FINISHED - FAILED - CANCELLED	Type: ENUM multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: False
ScProcessMonitor.progressStateInfo	This attributes following specialization for the "progressStateInfo" attribute of the "ProcessMonitor" data type for the Sc_Process". When the " Sc_Process.ScProcessMonitor.status " is equal to "RUNNING", it provides the more detailed progress information. allowedValues - 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	Type: ENUM multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: False
ScProcessMonitor.resultStateInfo	This attribute provides the following specialisation for the "resultStateInfo" attribute of the "ProcessMonitor" data type for the Sc_Process. In the attribute status is equal to "FAILED", it provides the reason for the failure. allowedValues for attribute status = "FAILED": - UNKNOWN - INCORRECT_CONFIGURATION, - NE_HARDWARE_ERROR_DETECTED" - DISCONNECTION_BETWEEN_NE_AND_OAM - OTHER	Type: ENUM multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: False
ScProcessMonitor.startTime	This attribute provides start time of the associated SCprocess, i.e. the time when the status changed from "NOT_STARTED" to "RUNNING". allowedValues: N/A	Type: DateTime multiplicity: 0..1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: False
ScProcessMonitor.endTime	This attribute provides the end time when status changed to FINISHED, CANCELLED or FAILED. allowedValues: N/A	Type: DateTime multiplicity: 0..1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: False
ScProcessMonitor.progressPercentage	This attribute indicates progress of the process as percentage. The percent can be measured by number of finished steps from total steps in the self configuration process. allowedValues: [0, 100]	Type: Integer multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: False

Attribute Name	Documentation and Allowed Values	Properties
ScMgmtProfileRef	This attributes specifies the DN of the associated ScMgmtProfile	Type: DN multiplicity: 1 isOrdered: N/A isUnique: N/A defaultValue: None isNullable: False

6.2.5 Common notifications

6.2.5.1 Configuration notifications

This clause presents a list of notifications, defined in TS 28.532 [3], that an MnS consumer may receive. The notification header attribute `objectClass/objectInstance` shall capture the DN of an instance of a class defined in the present document.

Table 6.2.5.1-1

Name	Qualifier	Notes
<code>notifyMOICreation</code>	O	--
<code>notifyMOIDeletion</code>	O	--
<code>notifyMOIAttributeValueChanges</code>	O	--

7 Stage 3 definition

7.1 RESTful HTTP-based solution set

The RESTful HTTP-based solution set for generic is defined in clause 12.1.1 in 3GPP TS 28.532 [3]. Corresponding `className` is `ScMgmtProfile` and `Sc_Process`.

7.2 OpenAPI specification

7.2.1 OpenAPI document "TS28532_ProvMnS.yaml"

OpenAPI definition of the provisioning MnS which includes the provisioning MnS operations see clause A.1.1 in 3GPP TS 28.532 [3].

7.2.2 OpenAPI document for RANSC NRM

The OpenAPI/YAML definitions for RANSC NRM are specified in 3GPP Forge [5].

Directory: OpenAPI

File: TS28317_RanScNrm.yaml

8 Procedure for Self-establishment

8.1 Self-configuration management

The Figure 8.1-1 illustrates the procedure for start self-configuration management.

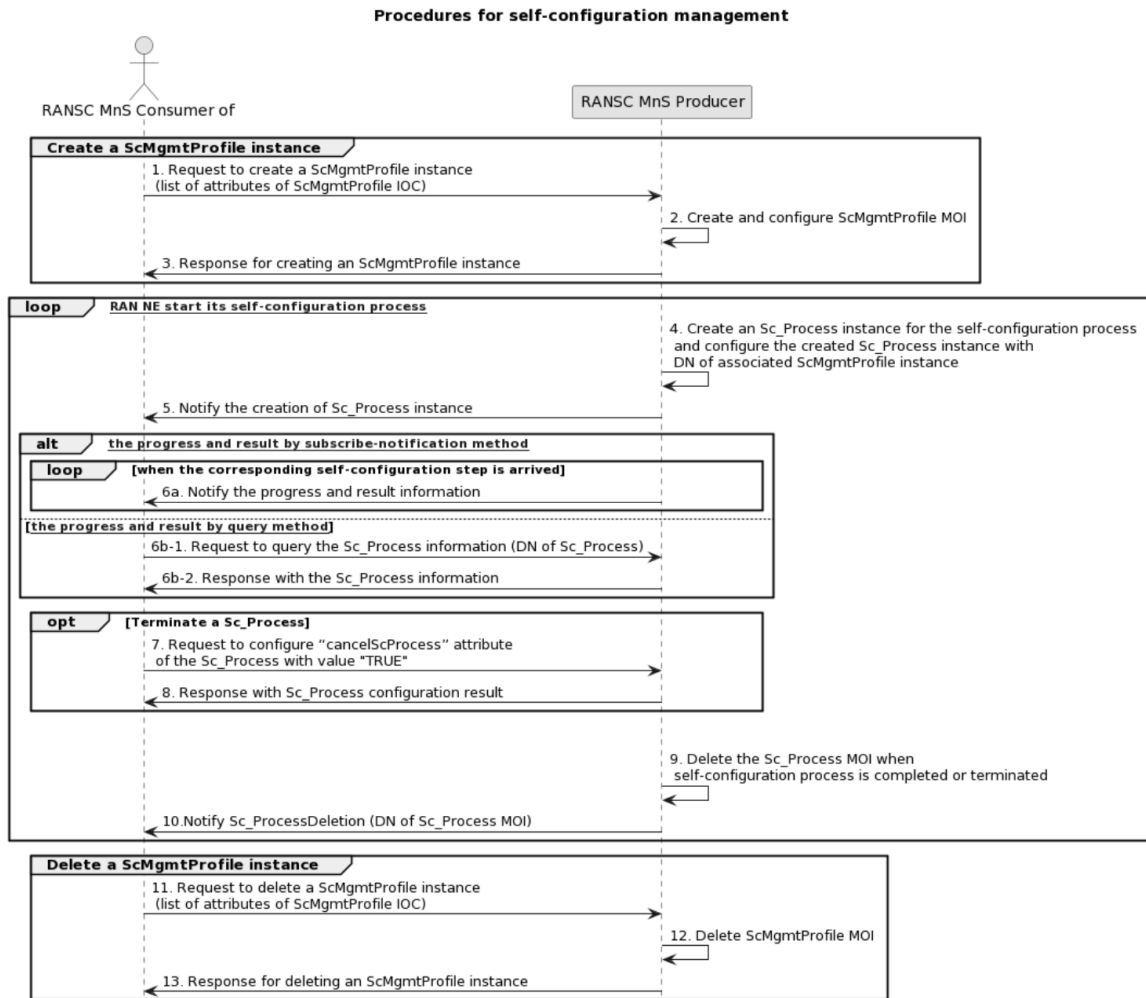


Figure 8.1-1: Procedures for self-configuration management

1. MnS consumer sends a request to create a `ScMgmtProfile` instance (see `createMOI` operation defined in TS 28.532 [3]) to MnS producer with self-configuration profile information. The detailed self-configuration profile information see attribute in `ScMgmtProfile` IOC in clause 6.2.3.1.2.
2. Based on the received request, the MnS producer creates the concrete `ScMgmtProfile` instance (i.e. instance of `ScMgmtProfile`) for NE(s) or NE type(s) specified by attribute "neInformation".
3. MnS Producer sends a response (see `createMOI` operation defined in TS 28.532 [3]) to the MnS Consumer with attribute "objectInstance" of the created `ScMgmtProfile` instance.
4. For each NE (specified in the created `ScMgmtProfile`) starting its self-configuration process, MnS producer creates an `Sc_Process` instance for the started self-configuration process and configure the created `Sc_Process` instance with DN of the associated `ScMgmtProfile`.
5. MnS producer notifies (see `notifyMOICreation` notification defined in TS 28.532 [3]) the MnS consumer about the creation of a new `Sc_Process` instance, including DN of `Sc_Process` instance.

The following step 6 describes the procedures for MnS consumer to monitor self-configuration progress and result. These steps can happen anytime after the Sc_Process instance is created until the Sc_Process instance is deleted.

- 6a) The MnS producer sends notification (see `notifyMOIAttributeValueChanges` defined in TS 28.532 [3]) to MnS consumer to notify the progress and result for self-configuration process (see attributes in Sc_Process IOC).
- 6b) The MnS consumer sends query request to MnS producer to query the attribute values of Sc_Process instance (see `getMOIAttributes` operation defined in TS 28.532 [3]) to obtain the progress and result (including DN of the Sc_Process instance and other attributes of Sc_Process instance) for self-configuration process.

Following Steps 7 and 8 are the steps for MnS consumer request to terminate an ongoing self-configuration process.

7. The MnS consumer sends a request (see `modifyMOIAttributes` operation defined in TS 28.532 [3]) to MnS producer to configure `cancelScProcess` attribute of the Sc_Process instance with value "TRUE" to terminate an ongoing self-configuration process.
8. The MnS producer sends a response (see `modifyMOIAttributes` operation defined in TS 28.532 [3]) for terminating an ongoing self-configuration process to MnS consumer.
9. MnS producer deletes the Sc_Process instance when self-configuration process is completed or terminated.
10. MnS producer notifies the MnS consumer about the deletion of a Sc_Process instance.

The following steps describes the procedures for MnS consumer request to delete a ScMgmtProfile instance to ultimate deactivation of requirements for self-configuration management for a set of RAN NEs.

11. MnS consumer sends a request to delete a ScMgmtProfile instance (see `deleteMOI` operation defined in TS 28.532 [3]) to MnS producer with the DN of the ScMgmtProfile instance.
12. Based on the received request, the MnS producer delete the concrete ScMgmtProfile instance
13. MnS Producer sends a response (see `deleteMOI` operation defined in TS 28.532 [3]) to the MnS Consumer.

Annex A (informative): PlantUML source code

A.1 Procedure for Self-establishment

A.1.1 Procedure for start self-configuration management

```
@startuml
title " Procedures for self-configuration management "
actor "RANSC MnS Consumer of " as SC
participant "RANSC MnS Producer" as SP
group Create a ScMgmtProfile instance
SC -> SP: 1. Request to create a ScMgmtProfile instance \n (list of attributes of ScMgmtProfile IOC)
SP -> SP: 2. Create and configure ScMgmtProfile MOI
SP -> SC: 3. Response for creating an ScMgmtProfile instance
end
loop [Corresponding RAN NE start its self-configuration process]

SP -> SP: 4. Create an Sc_Process instance for the self-configuration process \n and configure the
created Sc_Process instance with \n DN of associated ScMgmtProfile instance
SP -> SC: 5. Notify the creation of Sc_Process instance

alt [obtain the progress and result by subscribe-notification method]
loop when the corresponding self-configuration step is arrived
SP -> SC: 6a. Notify the progress and result information
end
else [[obtain the progress and result by query method]]
SC -> SP: 6b-1. Request to query the Sc_Process information (DN of Sc_Process)
SP -> SC: 6b-2. Response with the Sc_Process information
end
opt Terminate a Sc_Process
SC -> SP: 7. Request to configure "cancelScProcess" attribute \n of the Sc_Process with value "TRUE"
SP -> SC: 8. Response with Sc_Process configuration result
end

|||

SP -> SP: 9. Delete the Sc_Process MOI when \n self-configuration process is completed or terminated
SP->SC: 10.Notify Sc_ProcessDeletion (DN of Sc_Process MOI)
end
group Delete a ScMgmtProfile instance
SC -> SP: 11. Request to delete a ScMgmtProfile instance \n (list of attributes of ScMgmtProfile
IOC)
SP -> SP: 12. Delete ScMgmtProfile MOI
SP -> SC: 13. Response for deleting an ScMgmtProfile instance
end

skinparam sequenceActorBackgroundColor #FFFFFF
skinparam sequenceParticipantBackgroundColor #FFFFFF
skinparam noteBackgroundColor #FFFFFF
autonumber "#.'"
skinparam monochrome true
skinparam shadowing false
hide footbox
```

A.2 Information model definition for RANSC management

A.2.1 Relationship UML diagram

```

@startuml
hide circle
hide methods
hide members

skinparam class {
  AttributeIconSize 0
  BackgroundColor white
  BorderColor black
  ArrowColor black
}
skinparam Shadowing false
skinparam Monochrome true
skinparam ClassBackgroundColor White
skinparam NoteBackgroundColor White

class "<<ProxyClass>> \n ManagedEntity " as ManagedEntity{}
class "<<InformationObjectClass>>\n ScMgmtProfile " as ScMgmtProfile {}
class "<<InformationObjectClass>>\n Sc_Process" as Sc_Process{}

ManagedEntity "1" *-- "*" ScMgmtProfile : <<names>>
ManagedEntity "1" *-- "*" Sc_Process : <<names>>

ScMgmtProfile "1" <-right-> "*"Sc_Process
note left of ManagedEntity
Represents the folllowing IOCs:
SubNetwork
end note
@enduml

```

A.2.2 Inheritance UML diagram

```

@startuml
hide circle
hide methods
hide members

skinparam class {
  AttributeIconSize 0
  BackgroundColor white
  BorderColor black
  ArrowColor black
}
skinparam Shadowing false
skinparam Monochrome true
skinparam ClassBackgroundColor White
skinparam NoteBackgroundColor White

class "<<InformationObjectClass>>\n Top" as Top{}
class "<<InformationObjectClass>>\n ScMgmtProfile " as ScMgmtProfile {}
class "<<InformationObjectClass>>\n Sc_Process" as Sc_Process{}

Top <|--ScMgmtProfile
Top <|--Sc_Process
@enduml

```

Annex B (informative): Change history

Change history							
Date	Meeting	TDoc	CR	Rev	Cat	Subject/Comment	New version
2022-04	SA5#142e	S5-222393	-	-	-	Initial skeleton	0.0.0
2022-04	SA5#142e	S5-222634 S5-222605				1. Add structure 2. Add scope	0.1.0
2022-11	SA5#146	S5-226807 S5-226811				1.pCR 28.317 Add Concept for RANSC 2. PCR 28.317 Usecase and requirement for Self-configuration control and monitor	0.2.0
2023-03	SA5#147	S5-232145				pCR TS 28.317 Update the requirement format to follow the stage1 templated in TS 32.160	0.3.0
2023-05	SA5#149	S5-234581				pCR TR 28.317 Correction of the requirements from mandatory to optional	0.4.0
2023-06	SA5#149					Re-upload due to the previous version being corrupted	0.4.1
2023-08	SA5#150	S5-235818				pCR TS 28.317 Add procedure for Self-configuration management	0.5.0
2023-10	SA5#151	S5-236277 S5-236278				pCR TS 28.317 Add MnS component type A and type B for self-configuration management	0.6.0
2023-11	SA5#152	S5-237445 S5-237465 S5-237466				pCR TS 28.317 Add stage3 definition for MnS component type B pCR TS 28.317 Update use case for self-configuration management pCR TS 28.317 Update attribute definition of self-configuration management (type B)	0.7.0
2023-12	SA#102	SP-231524				EditHelp review and presented for information	1.0.0
2024-02	SA5#153	S5-240940 S5-240941 S5-240942 S5-240943 S5-240030				pCR TS 28.317 Add concept, use case and requirements for network configuration data handling pCR TS 28.317 Rapporteur clean up pCR TS 28.317 Add reference of stage3 definition for MnS component type A pCR TS 28.317 normative yaml code in 3gpp forge TS28.317 Rel18 correction to Schema definition Issues for SubNetwork of OpenAPI SS	1.1.0
2024-03	SA#103	SP-240253				EditHelp review and presented for approval	2.0.0
2024-03	SA#103					Upgrade to change control version	18.0.0

History

Document history		
V18.0.0	May 2024	Publication