

---

**3GPP TSG-SA5 (Telecom Management)**  
**Meeting #36, Shanghai, CHINA, 17 - 21 Nov 2003**

**S5-038807**

**Title:** LS on Reply LS on RAN Work Item "'Control of Remote Electrical Tilting Antenna" and possible impact on SA5'

**Response to:** LS (S5-038788 / R3-031756) on 'Reply LS on RAN Work Item "Control of Remote Electrical Tilting Antenna" and possible impact on TSG SA 5' from RAN3

**Release:** Release 6

**Work Item:** Remote Control of Electrical Tilting Antennas

**Source:** SA5

**To:** RAN3

**Cc:** RAN, SA, SA2

**Contact Person:**

**Name:** John MUDGE

**Tel. Number:** +44 1635 673587

**E-mail Address:** [john.mudge@vodafone.co.uk](mailto:john.mudge@vodafone.co.uk)

**Attachments:** Latest version of TS 32.642 (v511)  
Latest version of TS 32.612 (v520)

---

**1. Overall Description:**

SA5 would like to thank RAN3 for the LS on 'Reply LS on RAN Work Item "Control of Remote Electrical Tilting Antenna" and possible impact on TSG SA 5' and is grateful that RAN3 has responded to SA5's questions. This LS in reply is for ACTION.

On RAN3's responses to SA5's questions SA5 makes the following observations:

**Question 1**

From RAN3 LS R3-031756/S5-038788:

"...

It is the current TSG RAN WG 3 understanding that RET adjustments will be necessary for the following scenarios:

- a. Implementation of a new network plan with e.g. 4 RET adjustments per year
- b. Optimisation of the network depending on the interference situation with e.g. 4 RET adjustments per month
- c. Potentially optimisation of the network depending on the traffic situation with daily RET adjustments irrespective of the signalling scheme.

"..."

(a) and (b) are fine and would fit in with the network planning activities. On point (c) SA5 would like to point out that the minimum performance data collecting interval is 15 minutes; some analysis would need to be done and probably more than one performance data collecting interval would be required in order to assess the impact of a RET adjustment. This would need to be considered if there were to be rapid RET adjustments several times a day. Furthermore widespread RET adjustments to many RET antennas a number of times a day may have Configuration Management performance impacts.

## Question 2

SA5 is glad to report that the measurements Uplink traffic per cell, Downlink traffic per cell, Uplink interference per cell and Node B power per cell identified by RAN3 are already supported by the ltf-N interface.

Parameters for the configuration management and fault management of RET antennas like tilt values and alarms will require modifications to 3GPP specifications to allow these to be transferred over ltf-N.

Are there additional UTRAN cell parameters (not currently specified in the Network Resource Model (NRM)) that need to be managed for RET purposes when tilt adjustments are made? Please see the attached TS 32.642 "Configuration Management (CM); UTRAN network resources Integration Reference Point (IRP): Network Resource Model (NRM)" for the attributes that are currently standardised over ltf-N.

## Question 3

If there were a new UTRAN network element for RET control then the impact on SA5 would be:

- The O&M architecture specified in "Telecommunication management; Architecture" (TS 32.102) would need to be enhanced.
- Perhaps "Telecommunication management; Principles and high-level requirements" (TS 32.101) would need to be modified in addition to take account of the architectural changes.
- An architectural change would have a larger impact on the UTRAN NRM (TS 32.642) than if a few parameters were required to be added to a cell.

Please note that a new UTRAN element for RET may need the involvement of SA2.

If there is no new UTRAN network element then less work would be required from SA5. Thus providing such a solution would be more likely to be achievable by SA5 in the Release 6 timeframe.

## Question 4

When there is a set of RET adjustments how should error scenarios be handled? Would it be required to provide a reset capability in which all the set of RET adjustments are undone if one RET antenna from a set could not be successfully adjusted? Bulk CM follows a best efforts policy and requires the manager to instigate any resets. For information on Bulk CM please see the attached specification "Configuration Management (CM); Bulk CM Integration Reference Point (IRP); Information service" (TS 32.612).

Perhaps some NetMeeting/telephone conference calls could be arranged in order to efficiently progress the RET work? The purpose of the discussions would be to coordinate the RET work, to select a good O&M architecture for RET and to agree the enhancements to the ltf-N interface. Perhaps a joint meeting between RAN3 and SA5 would also be beneficial. RAN3 and SA5 are both meeting in Malaga in February during adjoining weeks and perhaps a joint meeting could be arranged then?

SA5 are considering creating a work item regardless of which architectural solution is adopted.

## 2. Actions:

### To RAN3 group:

- a) SA5 asks RAN3 whether there are additional UTRAN cell parameters (not currently specified in the Network Resource Model (NRM)) that need to be managed for RET purposes when tilt adjustments are made? Please see the attached TS 32.642 "Configuration Management (CM); UTRAN network resources Integration Reference Point (IRP): Network Resource Model (NRM)" for the attributes that are currently standardised over ltf-N.

- b) SA5 asks RAN3 when there is a set of RET adjustments how should error scenarios be handled? Would it be required to provide a reset capability in which all the set of RET adjustments are undone if one RET antenna from a set could not be successfully adjusted ?
- c) SA5 asks RAN3 to consider joint NetMeeting/telephone conference calls and a physical meeting to progress the RET work.

**3. Date of Next SA5 Meetings:**

<u>3GPPSA5#36-Bis</u>	WG	12 - 16 Jan 2004	TBD	North America
<u>3GPPSA5#37</u>	WG	23 - 27 Feb 2004	Malaga	ES
<u>3GPPSA5#37-Bis</u>	WG	29 Mar - 2 Apr 2004	Sophia Antipolis	FR
<u>3GPPSA5#38</u>	WG	10 - 14 May 2004	Beijing	CN
<u>3GPPSA5#38-bis</u>	WG	28 Jun - 2 Jul 2004	Sophia Antipolis	FR
<u>3GPPSA5#39</u>	WG	16 - 20 Aug 2004	TBD	
<u>3GPPSA5#39-Bis</u>	WG	4 - 8 Oct 2004	Sophia Antipolis	FR
<u>3GPPSA5#40</u>	WG	15 - 19 Nov 2004	TBD	

# 3GPP TS 32.612 V5.1.1 (2003-03)

---

*Technical Specification*

**3rd Generation Partnership Project;  
Technical Specification Group Services and System Aspects;  
Telecommunication management;  
Configuration Management (CM);  
Bulk CM Integration Reference Point (IRP); Information service  
(Release 5)**

---



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

---

Keywords

---

UMTS, management

**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.

© 2003, 3GPP Organizational Partners (ARIB, CWTS, ETSI, T1, TTA, TTC).  
All rights reserved.

# Contents

Foreword.....	6
Introduction.....	6
1 Scope.....	7
2 References.....	7
3 Definitions and abbreviations.....	7
3.1 Definitions.....	7
3.2 Abbreviations.....	9
4 System Overview.....	10
4.1 System Context.....	10
4.2 Compliance rules.....	10
4.3 Scope of Bulk CM Management Specification.....	11
5 Modelling approach.....	11
6 Information Object Classes.....	11
6.1 Information entities imported and local label.....	11
6.2 Class diagram.....	11
6.2.1 Attributes and relations.....	12
6.2.2 Inheritance.....	12
6.3 Information object classes definition.....	13
6.3.1 BulkCMIRP.....	13
6.3.1.1 Definition.....	13
6.4 Network Resource Model (NRM).....	13
7 Interface Definition.....	13
7.1 Class Diagram.....	14
7.1.1 Main Operations and Notifications.....	14
7.1.2 Suboperations of clause 10.....	15
7.2 Generic rules.....	15
7.3 bulkCmSession Interface.....	15
7.3.1 Operation startSession (M).....	15
7.3.1.1 Definition.....	15
7.3.1.2 Input parameters.....	15
7.3.1.3 Output parameters.....	15
7.3.1.4 Pre-condition.....	16
7.3.1.5 Post-condition.....	16
7.3.1.6 Exceptions.....	16
7.3.1.6.1 operationFailed.....	16
7.3.2 Operation endSession (M).....	16
7.3.2.1 Definition.....	16
7.3.2.2 Input parameters.....	16
7.3.2.3 Output parameters.....	16
7.3.2.4 Pre-condition.....	16
7.3.2.5 Post-condition.....	17
7.3.2.6 Exceptions.....	17
7.3.2.6.1 operationFailed.....	17
7.3.3 Operation abortSessionOperation (M).....	17
7.3.3.1 Definition.....	17
7.3.3.2 Input parameters.....	17
7.3.3.3 Output parameters.....	17
7.3.3.4 Pre-condition.....	17
7.3.3.5 Post-condition.....	17
7.3.3.6 Exceptions.....	18
7.3.3.6.1 operationFailed.....	18
7.3.4 Operation getSessionIds (M).....	18
7.3.4.1 Definition.....	18

7.3.4.2	Input parameters.....	18
7.3.4.3	Output parameters.....	18
7.3.4.4	Pre-condition.....	18
7.3.4.5	Post-condition.....	18
7.3.5	Operation getSessionStatus (M).....	18
7.3.5.1	Definition.....	18
7.3.5.2	Input parameters.....	18
7.3.5.3	Output parameters.....	19
7.3.5.4	Pre-condition.....	19
7.3.5.5	Post-condition.....	19
7.3.5.6	Exceptions.....	19
7.3.5.6.1	operationFailed.....	19
7.3.6	Operation getSessionLog (M).....	19
7.3.6.1	Definition.....	19
7.3.6.2	Input parameters.....	20
7.3.6.3	Output parameters.....	20
7.3.6.4	Pre-condition.....	20
7.3.6.5	Post-condition.....	20
7.3.6.6	Exceptions.....	20
7.3.6.6.1	operationFailed.....	20
7.4	bulkCmPassive Interface.....	20
7.4.1	Operation upload (M).....	20
7.4.1.1	Definition.....	20
7.4.1.2	Input parameters.....	21
7.4.1.3	Output parameters.....	21
7.4.1.4	Pre-condition.....	21
7.4.1.5	Post-condition.....	21
7.4.1.6	Exceptions.....	22
7.4.1.6.1	operationFailed.....	22
7.5	bulkCmActive Interface.....	22
7.5.1	Operation download (M).....	22
7.5.1.1	Definition.....	22
7.5.1.2	Input parameters.....	22
7.5.1.3	Output parameters.....	22
7.5.1.4	Pre-condition.....	22
7.5.1.5	Post-condition.....	22
7.5.1.6	Exceptions.....	23
7.5.1.6.1	operationFailed.....	23
7.5.2	Operation validate (O).....	23
7.5.2.1	Definition.....	23
7.5.2.2	Input parameters.....	23
7.5.2.3	Output parameters.....	23
7.5.2.4	Pre-condition.....	23
7.5.2.5	Post-condition.....	24
7.5.2.6	Exceptions.....	24
7.5.2.6.1	operationFailed.....	24
7.5.3	Operation preactivate (O).....	24
7.5.3.1	Definition.....	24
7.5.3.2	Input parameters.....	25
7.5.3.3	Output parameters.....	25
7.5.3.4	Pre-condition.....	25
7.5.2.5	Post-condition.....	25
7.5.3.6	Exceptions.....	25
7.5.3.6.1	operationFailed.....	25
7.5.4	Operation activate (M).....	25
7.5.4.1	Definition.....	25
7.5.4.2	Input parameters.....	26
7.5.4.3	Output parameters.....	26
7.5.4.4	Pre-condition.....	27
7.5.4.5	Post-condition.....	27
7.5.4.6	Exceptions.....	27
7.5.4.6.1	operationFailed.....	27
7.5.5	Operation fallback (M).....	27

7.5.5.1	Definition.....	27
7.5.5.2	Input parameters.....	27
7.5.5.3	Output parameters .....	28
7.5.5.4	Pre-condition.....	28
7.5.5.5	Post-condition .....	28
7.5.5.6	Exceptions .....	28
7.5.5.6.1	operationFailed .....	28
7.5.6	Validation and Checking Functions.....	28
7.5.6.1	Download Checks .....	28
7.5.6.2	Validate Checks .....	28
7.5.6.3	Preactivation Checks .....	28
7.5.6.4	Activate Checks .....	29
8	Bulk Configuration Data File .....	29
8.1	Interface BulkCmIRPNotifications#1 .....	29
8.1.1	Notification notifySessionStateChanged (M).....	29
8.1.1.1	Definition.....	29
8.1.1.3	Triggering events .....	30
8.1.1	Notification NotifyGetSessionLogEnded (M) .....	30
8.1.1.1	Definition.....	30
8.1.1.2	Input parameters.....	30
8.1.1.3	Triggering event.....	30
9	State Machine .....	30
9.1	State Machine Overview .....	30
9.2	State Machine Description .....	31
9.2.1	Upload Phase .....	33
9.2.2	Download Phase.....	34
9.2.4	Validation Phase.....	34
9.2.5	Preactivation Phase.....	35
9.2.6	Activation Phase.....	35
9.2.7	Fallback Phase .....	37
9.3	State Machine Pre and Post Conditions Tables .....	38
10	Bulk Configuration Data File .....	39
10.1	Bulk Configuration Data Management Actions – Sub-operations .....	39
10.1.1	bulkCmCreateMo (Create MO Sub-operation) (M) .....	39
10.1.2	bulkCmDeleteMo (Delete MO Sub-operation) (M) .....	40
10.1.3	bulkCmChangeMo (Change MO Sub-operation) (M).....	40
10.2	Rules for ordering Management Actions (Sub-operations) in Configuration Data Files .....	40
10.2.1	Download files .....	40
10.2.2	Upload files.....	41
<b>Annex A (informative):</b>	<b>Scenarios .....</b>	<b>42</b>
<b>Annex B (informative):</b>	<b>Bulk CM Application and Operation Principles.....</b>	<b>47</b>
B1	Key characteristics.....	47
<b>Annex C (informative):</b>	<b>Change history .....</b>	<b>48</b>



---

## Foreword

This Technical Specification (TS) has been produced by the 3<sup>rd</sup> 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

Configuration Management (CM), in general, provides the operator with the ability to assure correct and effective operation of the 3G network as it evolves. CM actions have the objective to control and monitor the actual configuration on the Network Element (NEs) and Network Resources (NRs), and they may be initiated by the operator or by functions in the Operations Systems (OSs) or NEs.

CM actions may be requested as part of an implementation programme (e.g. additions and deletions), as part of an optimisation programme (e.g. modifications), and to maintain the overall Quality of Service. The CM actions are initiated either as a single actions on single NEs of the 3G network or as part of a complex procedure involving actions on many resources/objects in one or several NEs.

---

# 1 Scope

The present document (Bulk Configuration Management IRP: Information Service) defines an Integration Reference Point (IRP) through which an 'IRPAgent' (typically an Element Manager or Network Element) can communicate bulk Configuration Management related information to one or several 'IRPManagers' (typically Network Managers).

---

## 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.

- [1] 3GPP TS 32.101: "3G Telecom Management principles and high level requirements".
- [2] 3GPP TS 32.102: "3G Telecom Management architecture".
- [3] 3GPP TS 32.302: "Telecommunication Management; Notification Management; Part 2: Notification Integration Reference Point; Information Service".
- [4] 3GPP TS 32.622: "3G Configuration Management: Generic Network Resources IRP: NRM".
- [5] 3GPP TS 32.642: "3G Configuration Management: UTRAN Network Resources IRP: NRM".
- [6] 3GPP TS 32.652: "3G Configuration Management: GERAN Network Resources IRP: NRM".
- [7] 3GPP TS 32.300: "Name Convention for Managed Objects".
- [8] 3GPP TS 32.600: "3G Configuration Management: Concepts and requirements".
- [9] 3GPP TS 32.312: "Generic IRP Management: Information Service".
- [10] 3GPP TS 32.632: "3G Configuration Management: Core Network Resource IRP: NRM"

---

## 3 Definitions and abbreviations

### 3.1 Definitions

For the purposes of the present document, the following terms and definitions apply. For terms and definitions not found here, please refer to 3GPP TS 32.101 [1], 3GPP TS 32.102 [2] and 3GPP TS 32.600 [8].

**Association:** In general it is used to model relationships between Managed Objects. Associations can be implemented in several ways, such as:

- (1) name bindings,
- (2) reference attributes, and
- (3) association objects.

This IRP stipulates that containment associations shall be expressed through name bindings, but it does not stipulate the implementation for other types of associations as a general rule. These are specified as separate entities in the object models (UML diagrams). Currently (in R99) however, all (non-containment) associations are modelled. by means of reference attributes of the participating MOs.

**Data:** is any information or set of information required to give software or equipment or combinations thereof a specific state of functionality.

**Element Manager (EM):** provides a package of end-user functions for management of a set of closely related types of Network Elements (NEs). These functions can be divided into two main categories:

- *Element Management Functions* for management of NEs on an individual basis. These are basically the same functions as supported by the corresponding local terminals.
- *Sub-Network Management Functions* that are related to a network model for a set of NEs constituting a clearly defined sub-network, which may include relations between the NEs. This model enables additional functions on the sub-network level (typically in the areas of network topology presentation, alarm correlation, service impact analysis and circuit provisioning).

**IRP:** See 3GPP TS 32.101 [1].

**IRP Information Service (IS):** See 3GPP TS 32.101 [1].

**IRP Network Resource Model (NRM):** See 3GPP TS 32.101 [1].

**IRP Solution Set (SS):** See 3GPP TS 32.101 [1].

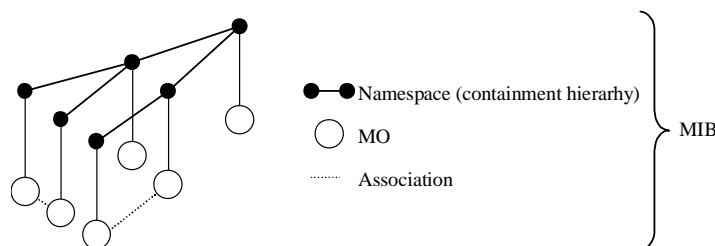
**Managed Element (ME):** An instance of the Managed Object Class G3ManagedElement/ManagedElement.

**Managed Object (MO):** In the context of the present document, a Managed Object (MO) is a software object that encapsulates the manageable characteristics and behaviour of a particular Network Resource. The MO is instance of a MO class defined in a MIM/NRM. An MO class has attributes that provide information used to characterize the objects that belong to the class (the term "attribute " is taken from TMN and corresponds to a "property " according to CIM). Furthermore, a MO class can have operations that represent the behaviour relevant for that class (the term "operation " is taken from TMN and corresponds to a "method " according to CIM). An MO class may support notifications that provide information about an event occurrence within a network resource.

**Managed Object Class (MOC):** a description of all the common characteristics for a number of MOs, such as their attributes, operations, notifications and behaviour.

**Managed Object Instance (MOI):** an instance of a MOC, which is the same as a MO as described above.

**Management Information Base (MIB):** A MIB is an instance of an NRM and has some values on the defined attributes and associations specific for that instance. In the context of the present document , a MIB consist of (1) a Name space (describing the MO containment hierarchy in the MIB through Distinguished Names), (2) a number of Managed Objects with their attributes and (3) a number of Associations between these MOs. Also note that TMN (X.710 [7]) defines a concept of a Management Information Tree (also known as a Naming Tree) that corresponds to the name space (containment hierarchy) portion of this MIB definition. The following figure depicts the relationships between a Name space and a number of participating MOs (the shown association is of a non-containment type)



**Figure 1: Relationships between a Name space and a number of participating MOs**

**Management Information Model (MIM):** Also referred to as NRM – see the definition below. There is a slight difference between the meaning of MIM and NRM – the term MIM is generic and can be used to denote any type of management model, while NRM denotes the model of the actual managed telecommunications Network Resources (NRs).

**Name space:** A name space is a collection of names. The IRP name convention [7] restricts the name space to a hierarchical containment structure, including its simplest form - the one-level, flat name space. All Managed Objects in a MIB shall be included in the corresponding name space and the MIB/name space shall only support a strict

hierarchical containment structure (with one root object). A Managed Object that contains another is said to be the superior (parent); the contained Managed Object is referred to as the subordinate (child). The parent of all MOs in a single name space is called a Local Root. The ultimate parent of all MOs of all managed systems is called the Global Root.

**Network Element (NE):** is a discrete telecommunications entity, which can be, managed over a specific interface e.g. the RNC.

**Network Manager (NM):** provides a package of end-user functions with the responsibility for the management of a network, mainly as supported by the EM(s) but it may also involve direct access to the NEs. All communication with the network is based on open and well-standardised interfaces supporting management of multi-vendor and multi-technology NEs.

**Network Resource (NR):** is a component of a NE, which can be identified as a discrete separate entity and is in an object oriented environment for the purpose of management represented by an abstract entity called Managed Object (MO).

**Network Resource Model (NRM):** a model representing the actual managed telecommunications Network Resources (NRs) that a System is providing through the subject IRP. An NRM describes Managed Object Classes (MOC), their associations, attributes and operations. The NRM is also referred to as "MIM" (see above) which originates from the ITU-T TMN.

**Operator:** is either

- a human being controlling and managing the network; or
- a company running a network (the 3G network operator).

## 3.2 Abbreviations

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

CM	Configuration Management
CMIP	Common Management Information Protocol
CORBA	Common Object Request Broker Architecture
EM	Element Manager
FM	Fault Management
IRP	Integration Reference Point
ITU-T	International Telecommunication Union, Telecommunication Standardisation Sector
MIB	Management Information Base
MIM	Management Information Model
MO	Managed Object
MOC	Managed Object Class
MOI	Managed Object Instance
NE	Network Element
NM	Network Manager
NR	Network Resource
NRM	Network Resource Model
PM	Performance Management
SS	Solution Set
SW	Software
TM	Telecom Management
UML	Unified Modelling Language (OMG)
UMTS	Universal Mobile Telecommunications System
XML	EXtensible Markup Language

## 4 System Overview

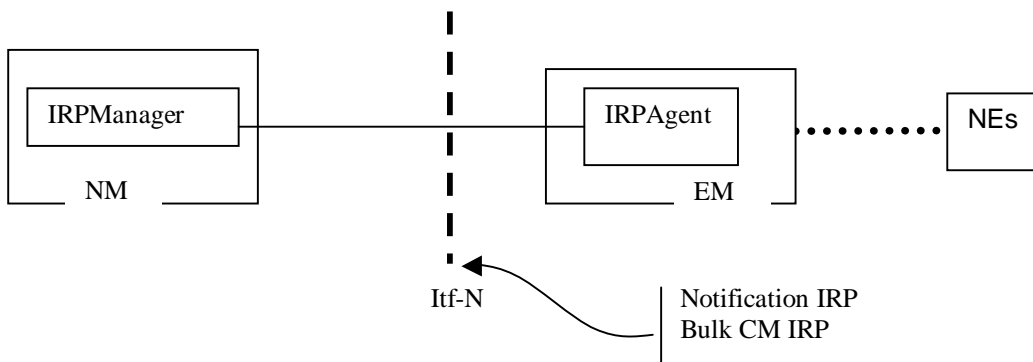
### 4.1 System Context

Figure 4.1 and 4.2 identify system contexts of the IRP defined by the present specification in terms of its implementation called IRPAgent and the user of the IRPAgent, called IRPManager. For a definition of IRPManager and IRPAgent, see 3GPP TS 32.102 [2].

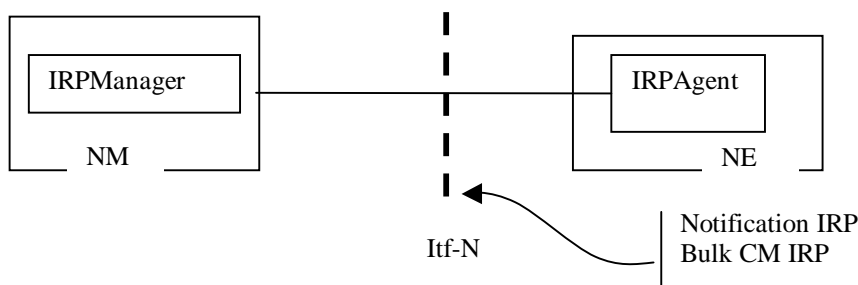
The IRPAgent implements and supports this IRP. The IRPAgent can reside in an Element Manager (EM) or a Network Element (NE) (see also [2] clause 8). In the former case, the interfaces (represented by a thick dotted line) between the EM and the NEs is not the subject of this IRP.

An IRPManager using this IRP shall choose one of the two System Contexts defined here, for each NE. For instance, if an EM is responsible for managing a number of NEs, the NM shall access this IRP through the EM and not directly to those NEs. For another IRP though, the System Context may be different.

As indicated in Figure 2 and Figure 3, the subject IRP needs to be complemented with the Notification IRP 3GPP TS 32.302 [3]. (This is to allow the IRP Manager to subscribe and unsubscribe to notifications issued by the IRP Agent).



**Figure 2: System Context A**



**Figure 3: System Context B**

### 4.2 Compliance rules

For general definitions of compliance rules related to qualifiers (Mandatory/Optional/Conditional) for *operations*, *notifications* and *parameters* (of operations and notifications) please refer to 3GPP TS 32.102 [2].

The following defines the meaning of Mandatory and Optional attributes and associations for Operations, in Solution Sets to the Bulk CMIRP:

- The IRPManager shall support all mandatory attributes/associations. The IRPManager shall be prepared to receive information related to mandatory as well as optional attributes/associations without failure; however the IRPManager does not have to support handling of the optional attributes/associations.

- The IRPAgent shall support all mandatory attributes/associations. It may support optional attributes/associations.

An IRPAgent that incorporates vendor-specific extensions must support normal communication with a 3GPP SA5-compliant IRPManager with respect to all mandatory and optional managed object classes, attributes, associations, operations, parameters and notifications without requiring the IRPManager to have any knowledge of the extensions.

Given that

- rules for vendor-specific extensions remain to be fully specified, and
- many scenarios under which IRPManager and IRPAgent interwork may exist,

it is recognised that in R4/R5 the IRPManager, even though it is not required to have knowledge of vendor-specific extensions, may be required to be implemented with an awareness that extensions can exist and behave accordingly.

## 4.3 Scope of Bulk CM Management Specification

Within the scope of this document it is specified how Bulk CM IRP IS allows an IRPManager to actively configure NEs over interface-N using an IRPAgent supporting Bulk CM IRP IS. It is not within the scope of this document to specify how Bulk CM IRP IS and the IRPAgent shall resolve any potentially conflicting CM management activities that could arise from either multiple concurrent active IRPManager management Bulk CM IRP sessions, any other IRP conflicting CM management activities, or any CM management activities outside of the scope of an IRP and interface-N. From a system perspective such potential conflicts need to be guarded against, but how this done e.g. operational procedures or implementation specific recovery in an IRPManager or IRPAgent, is beyond the scope of this document.

---

## 5 Modelling approach

This clause identifies the modelling approach adopted and used in this IRP.

The modelling approach adopted and used in this IRP is the same as that defined in 3GPP TS 32.622: "Generic Network Resources IRP: NRM" [4].

---

## 6 Information Object Classes

### 6.1 Information entities imported and local label

Label reference	Local label
32.622, information object class, Top	Top
32.302 [3], information object class, NotificationIRP	NotificationIRP
32.302 [3], interface, notificationIRPNotification	notificationIRPNotification
32.622 [4], [information object class, GenericIRP	GenericIRP
32.622 [4], information object class, IRPAgent	IRPAgent
32.312 [9], information object class, ManagedGenericIRP	ManagedGenericIRP

### 6.2 Class diagram

This subclass introduces the set of information object classes (IOCs) that encapsulate information within the IRPAgent. The intent is to identify the information required for the BulkCMIRP Agent implementation of its operations and notification emission. This sub-clause provides the overview of all support object classes in UML. Subsequent sub-clauses provide more detailed specification of various aspects of these support object classes.

### 6.2.1 Attributes and relations

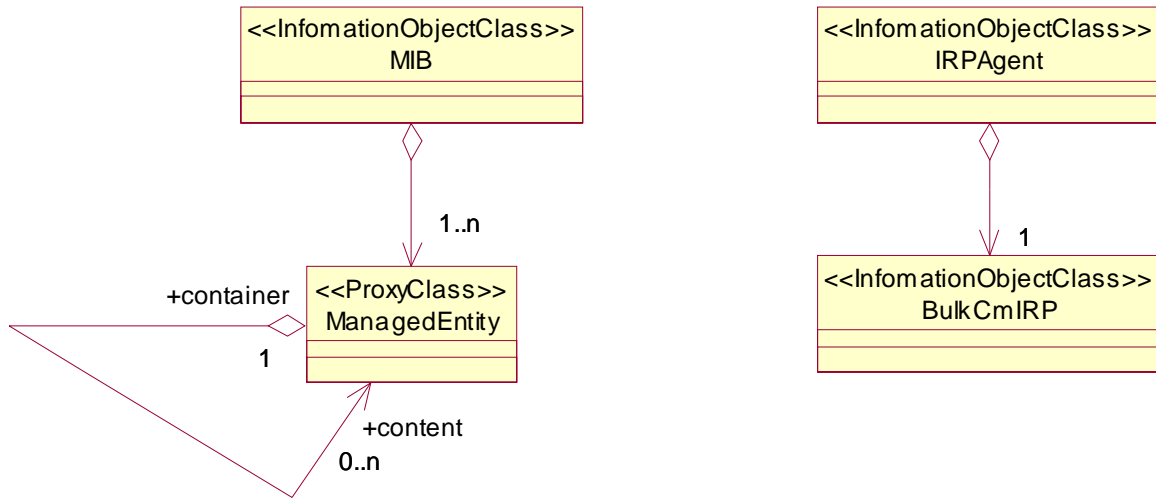


Figure:

### 6.2.2 Inheritance

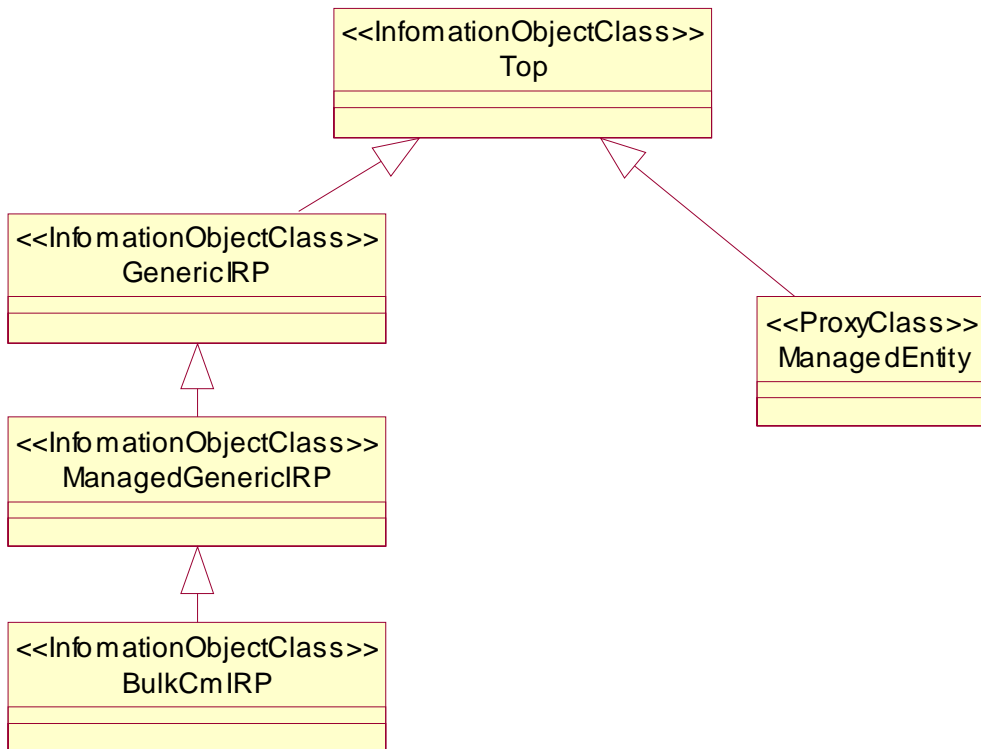


Figure:

## 6.3 Information object classes definition

### 6.3.1 BulkCMIRP

#### 6.3.1.1 Definition

BulkCMIRP is the representation of the configuration management capabilities specified by this specification. This IOC inherits from ManagedGenericIRP IOC specified in TS.32.312 [9].

## 6.4 Network Resource Model (NRM)

NRMs for Bulk CM IRP are defined in other Network Resource IRP documents of CM, For Bulk CM IRP IS these are:

32.622: "3G Configuration Management: Generic Network Resources IRP: NRM" [4],

32.632: "3G Configuration Management: Core Network Resource IRP: NRM" [10],

32.642: "3G Configuration Management: UTRAN Network Resources IRP: NRM" [5],

32.652: "3G Configuration Management: GERAN Network Resources IRP: NRM" [6].

These NRM documents define all the MOCs and attributes that can be configuration managed by Bulk CM IRP IS.

---

## 7 Interface Definition

Subclause 7.1 illustrates the operations and notifications defined as interfaces implemented and used by IRPAgent and IRPManager, described using UML notation (Interface in IRP Information Model is identical to concepts conveyed by stereotype <<interface>> of UML). Parameters and return status are not indicated.

The interfaces support multiple IRPManagers connected to an IRPAgent. Configuration data files defined in clause 10 define bulk configuration management changes. The following configuration data file handling operations exist in the Itf-N.

- startSession
- endSession
- upload
- download
- validate
- preactivate
- activate
- fallback
- abortSessionOperation
- getSessionIds
- getSessionStatus
- getSessionLog
- getBulkCmIRPVersion

Notification IRP [3] related operations are also associated with Bulk CM IRP (e.g. Subscribe an Unsubscribe), but these operations are described in 32.302: "Telecommunication Management; Notification Management: Part 2: Notification IRP; Information Service " [3].).



The operations, upload, download, validate, preactivate, activate, fallback and getSessionLog are performed asynchronously in that when the operations are initiated, the IRPAgent returns an indication that the requested activity has begun, and the IRPManager may release and continue with other tasks. If the IRPManager has subscribed on event notifications, then the IRPManager will receive a notification when the task requested in the operation is complete.

The operations startSession, endSession, abortSessionOperation, getSessionIds, getSessionStatus and getBulkCmIRPVersion are performed synchronously in that the result of the operation is returned as a callback to the operation, and the IRPManager will wait until the response is received before continuing. Refer to subclause 4.3 for system conditions that need to be potentially managed, but are outside the scope of this document.

## 7.1 Class Diagram

### 7.1.1 Main Operations and Notifications

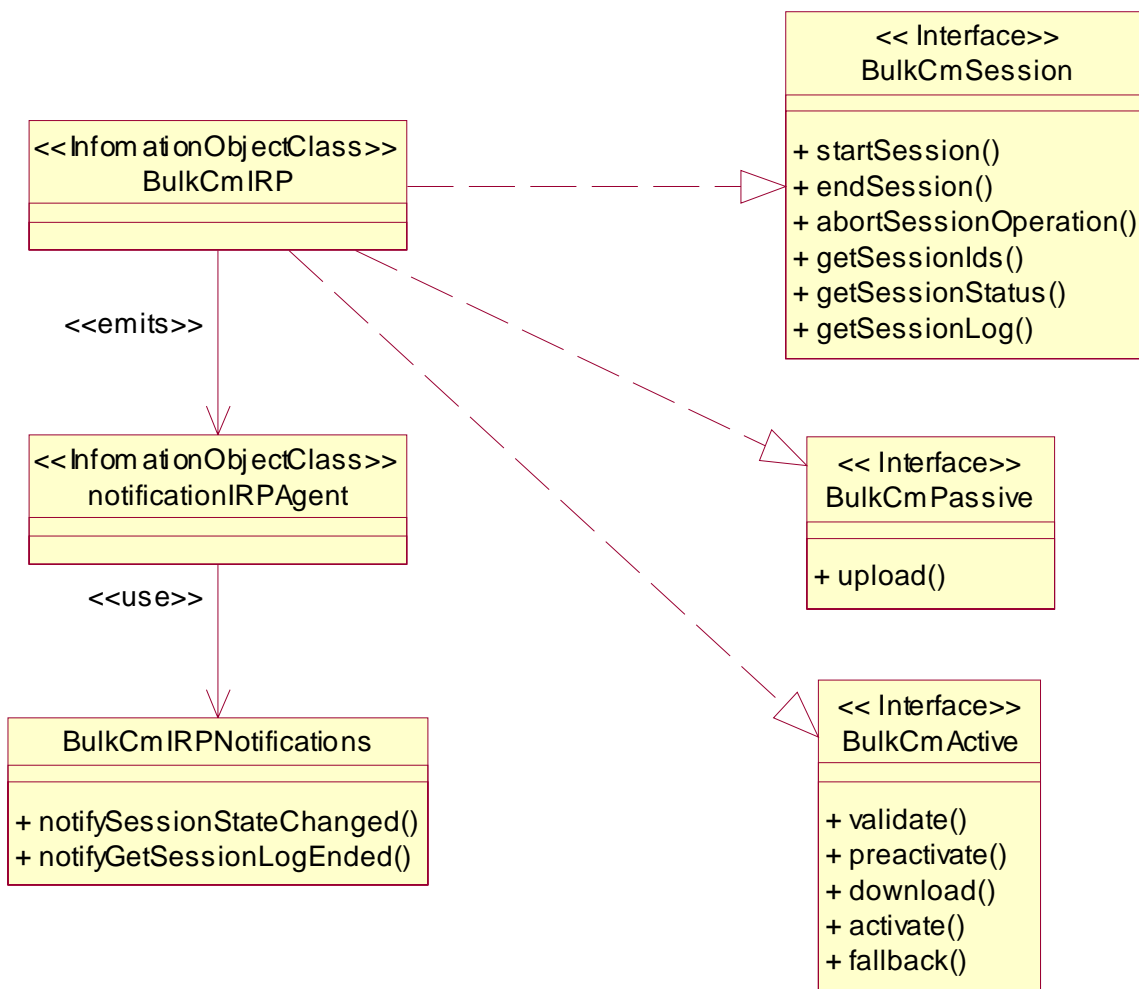


Figure:

## 7.1.2 Suboperations of clause 10



Figure:

## 7.2 Generic rules

- Rule 1: each operation with at least one input parameter supports a pre-condition `valid_input_parameter` which indicates that all input parameters shall be valid with regards to their information type. Additionally, each such operation supports an exception `operation_failed_invalid_input_parameter` which is raised when pre-condition `valid_input_parameter` is false. The exception has the same entry and exit state.
- Rule 2: Each operation with at least one optional input parameter supports a set of pre-conditions `supported_optional_input_parameter_xxx` where "xxx" is the name of the optional input parameter and the pre-condition indicates that the operation supports the named optional input parameter. Additionally, each such operation supports an exception `operation_failed_unsupported_optional_input_parameter_xxx` which is raised when (a) the pre-condition `supported_optional_input_parameter_xxx` is false and (b) the named optional input parameter is carrying information. The exception has the same entry and exit state.
- Rule 3: each operation shall support a generic exception `operation_failed_internal_problem` which is raised when an internal problem occurs and that the operation cannot be completed. The exception has the same entry and exit state.

## 7.3 bulkCmSession Interface

### 7.3.1 Operation startSession (M)

#### 7.3.1.1 Definition

The IRPManager invokes this operation to start a session state machine and initialise temporary entities to be related with bulk data configuration `sessionId` in the IRPAgent.

#### 7.3.1.2 Input parameters

Parameter Name	Qualifier	Information type	Comment
<code>sessionId</code>	M	String identifying the session	Identifies the new session and process to be associated with a bulk data operation e.g. upload or download.

#### 7.3.1.3 Output parameters

Parameter Name	Qualifier	Matching Information	Comment
<code>status</code>	M	ENUM(OperationSucceeded, OperationFailed).	indicates (a) operation is successful and (b) operation failed because of specified or unspecified reasons

### 7.3.1.4 Pre-condition

sessionIdNotInUse

Assertion Name	Definition
sessionIdNotInUse	No state, see clause 9. The supplied sessionId is not already open in the Bulk CM IRP Agent.

### 7.3.1.5 Post-condition

sessionStarted

Assertion Name	Definition
sessionStarted	State = IDLE, see clause 9. The Bulk CM IRP Agent has successfully opened the session and is ready to handle other operations associated with the session.

### 7.3.1.6 Exceptions

#### 7.3.1.6.1 operationFailed

Exception Name	Definition
operationFailed	<b>Condition:</b> Pre-condition is false or post-condition is false. <b>Returned information:</b> The output parameter status. <b>Exit state:</b> Entry state.

## 7.3.2 Operation endSession (M)

### 7.3.2.1 Definition

The IRPManager invokes this operation to end a session state machine and delete all temporary entities and their related bulk data configuration for a specified sessionId in the IRP Agent. If a preactivation had been invoked, endSession should release any internal local resources allocated for the preactivation. The deletion will be rejected if the configuration state is in a working state: e.g. uploading (including getting a log), downloading or activating.

### 7.3.2.2 Input parameters

Parameter Name	Qualifier	Information type	Comment
sessionId	M	String identifying the session	Identifies this specific session and process associated with an earlier bulk data operation e.g. upload or download

### 7.3.2.3 Output parameters

Parameter Name	Qualifier	Matching Information	Comment
status	M	ENUM(OperationSucceeded, OperationFailed).	indicates (a) operation is successful and (b) operation failed because of specified or unspecified reasons

### 7.3.2.4 Pre-condition

sessionInStableState

Assertion Name	Definition
sessionInStableState	The supplied sessionId is open in the Bulk CM IRP Agent and in not in a transition status as defined in clause 9, table 1.

### 7.3.2.5 Post-condition

sessionEnded

Assertion Name	Definition
sessionIdNotInUse	No state, see clause 9. The session is closed and the sessionId is no longer in use.

### 7.3.2.6 Exceptions

#### 7.3.2.6.1 operationFailed

Exception Name	Definition
operationFailed	<p><b>Condition:</b> Pre-condition is false or post-condition is false.</p> <p><b>Returned information:</b> The output parameter status.</p> <p><b>Exit state:</b> Entry state.</p>

## 7.3.3 Operation abortSessionOperation (M)

### 7.3.3.1 Definition

An IRPManager invokes this operation to request an IRPAgent to abort a currently activate asynchronous operation. The abort will cause the session state machine to exit the current state and enter a new state, see clause 9.

### 7.3.3.2 Input parameters

Parameter Name	Qualifier	Information type	Comment
sessionId	M	String identifying the session	Identifies this specific session and process associated with an earlier bulk data operation e.g. upload or download for which the abort is required.

### 7.3.3.3 Output parameters

Parameter Name	Qualifier	Matching Information	Comment
status	M	ENUM(OperationSucceeded, OperationFailed).	indicates (a) start of abort operation is successful and (b) abort operation failed because of specified or unspecified reasons

### 7.3.3.4 Pre-condition

operationInProgress

Assertion Name	Definition
operationInProgress	The supplied sessionId is open in the Bulk CM IRP Agent and an operation is in an 'in progress' state as defined in clause 9, table 1.

### 7.3.3.5 Post-condition

operationAborted

Assertion Name	Definition
operationAborted	State changed from 'in progress' to state as a function of the original state as defined in clause 9.

### 7.3.3.6 Exceptions

#### 7.3.3.6.1 operationFailed

Exception Name	Definition
operationFailed	<b>Condition:</b> Pre-condition is false or post-condition is false. <b>Returned information:</b> The output parameter status. <b>Exit state:</b> Entry state.

### 7.3.4 Operation getSessionIds (M)

#### 7.3.4.1 Definition

An IRPManager invokes this operation to request an IRPAgent to return a list of all its currently open sessionIds.

#### 7.3.4.2 Input parameters

None.

#### 7.3.4.3 Output parameters

Parameter Name	Qualifier	Matching Information	Comment
sessionIdList	M	List of strings identifying sessions	A list of all the sessionIds an IRPAgent currently has open i.e. started with startSession and not ended with endSession operations.
status	M	ENUM(OperationSucceeded, OperationFailed).	indicates (a) operation is successful and (b) operation failed because of specified or unspecified reasons

#### 7.3.4.4 Pre-condition

None.

#### 7.3.4.5 Post-condition

None.

### 7.3.5 Operation getSessionStatus (M)

#### 7.3.5.1 Definition

The IRPManager invokes this operation to request the IRPAgent to send the current state of the bulk configuration data file operation. The IRPAgent returns the current state. See clause 9.

This operation can be invoked in any session state and does not change the session state.

#### 7.3.5.2 Input parameters

Parameter Name	Qualifier	Information type	Comment
sessionId	M	String identifying the session	Identifies this specific session and process associated with an earlier bulk data operation e.g. upload or download for which the current status is required.

### 7.3.5.3 Output parameters

Parameter Name	Qualifier	Matching Information	Comment
sessionState	M	List of ENUM(Idle, Upload In Progress, Upload Failed, Upload Completed, Down Load In Progress, Download Failed, Download Completed, Validation In Progress, Validation Failed, Validation Completed, Preactivation In Progress, Preactivation Failed, Preactivation Partly Realised, Preactivation Completed, Activation In Progress, Activation Failed, Activation Partly Realised, Activation Completed, Fallback In Progress, Fallback Failed, Fallback Partly Realised, Fallback Completed)	Indicates current state of the configuration data file operation. See clause 9, i.e. will be one of: Idle, Upload In Progress, Upload Failed, Upload Completed, Down Load In Progress, Download Failed, Download Completed, Validation In Progress, Validation Failed, Validation Completed, Preactivation In Progress, Preactivation Failed, Preactivation Partly Realised, Preactivation Completed, Activation In Progress, Activation Failed, Activation Partly Realised, Activation Completed, Fallback In Progress, Fallback Failed, Fallback Partly Realised, Fallback Completed.
status	M		Indicates (a) start of operation is successful or (b) operation failed because of specified or unspecified reasons

### 7.3.5.4 Pre-condition

knownSessionID

Assertion Name	Definition
knowSessionID	Session has been successfully started (clause 7.3.1) and not ended (clause 7.3.2).

### 7.3.5.5 Post-condition

None.

### 7.3.5.6 Exceptions

#### 7.3.5.6.1 operationFailed

Exception Name	Definition
operationFailed	<b>Condition:</b> Pre-condition is false. <b>Returned information:</b> The output parameter status. <b>Exit state:</b> Entry state.

## 7.3.6 Operation getSessionLog (M)

### 7.3.6.1 Definition

An IRPManager invokes this operation to request an IRPAgent to provide a log of the results from activities associated with bulk data configuration file sessionId operations.

This operation can be invoked in any session state and does not change the session state.

### 7.3.6.2 Input parameters

Parameter Name	Qualifier	Information type	Comment
sessionId	M	String identifying the session	Identifies this specific session and process associated with an earlier bulk data operation e.g. upload or download for which the current log is required.
logFileReference	M	String of complete path of file and name.	Specifies the address and file name where the result is to be placed in the IRPManager.
contentType	M	Boolean	Identifies if retrieved file should include (a) complete log including errors, (b) only errors.

### 7.3.6.3 Output parameters

Parameter Name	Qualifier	Matching Information	Comment
status	M	ENUM(OperationSucceeded, OperationFailed).	Indicates (a) start of operation is successful and (b) operation failed because of specified or unspecified reasons

### 7.3.6.4 Pre-condition

knownSessionID

Assertion Name	Definition
knowSessionID	Session has been successfully started (clause 7.3.1) and not ended (clause 7.3.2).

### 7.3.6.5 Post-condition

sessionLogWrite

Assertion Name	Definition
sessionLogWrite	The Bulk CM IRP Agent will begin to write contents of log to the specified address and file.

### 7.3.6.6 Exceptions

#### 7.3.6.6.1 operationFailed

Exception Name	Definition
operationFailed	<b>Condition:</b> Pre-condition is false or post-condition is false. <b>Returned information:</b> The output parameter status. <b>Exit state:</b> Entry state.

## 7.4 bulkCmPassive Interface

### 7.4.1 Operation upload (M)

#### 7.4.1.1 Definition

An IRPManager invokes this operation to request the IRPAgent to create a file containing bulk configuration data (clause 10) and transfer the file to the indicated globally unique data file reference.

## 7.4.1.2 Input parameters

Parameter Name	Qualifier	Information type	Comment
SessionId	M	String identifying the session	Identifies this specific session and process associated with the requested bulk data upload.
UploadDataFileReference	M	String of complete path of file and name.	This specifies a globally unique file reference to where the specified scope of bulk data is to be uploaded and stored .
BaseObjectInstance	M	DistinguishedName	The MO where the search starts. This is a full Distinguished Name according to 3GPP TS 32.300 [7].
Scope	M	SEQUENCE < ENUM { BASE_OBJECT_ONLY, NTH_LEVEL_SUBORDINATES, BASE_NTH_LEVEL, BASE_ALL}, Integer>	This parameter defines how many levels of the containment hierarchy to search (i.e. apply the filter defined below). The search starts from the MO given by the baseObjectInstance parameter. The levels of search that may be performed are: the base object alone (default); the n-th level subordinates of the base object; the base object and all of its subordinates down to and including the n-th level; the base object and all of its subordinates.
filter	M	See comment.	This parameter defines a filter test to be applied to the scoped Managed Object(s). If the filter is empty, all of the managed objects included by the scope are selected. The actual syntax and capabilities of the filter is Solution Set specific. However, each Solution Set support a filter consisting of one or several assertions that may be grouped using the logical operators AND, OR and NOT. Each assertion is a logical expression of attribute existence, attribute value comparison ("equal to X, less than Y" etc.) and MO Class.

## 7.4.1.3 Output parameters

Parameter Name	Qualifier	Matching Information	Comment
status	M	ENUM(OperationSucceeded, OperationFailed).	indicates (a) start of operation is successful and (b) operation failed because of specified or unspecified reasons

## 7.4.1.4 Pre-condition

sessionIdle

Assertion Name	Definition
sessionIdle	State as defined in clause 9. The Bulk CM IRP Agent has successfully opened the session and is ready to handle the first operations of the session or repeat this operation.

## 7.4.1.5 Post-condition

uploadInProgress

Assertion Name	Definition
Upload in progress	State = UPLOAD_IN_PROGRESS, as defined in clause 9. The Bulk CM IRP Agent has successfully started the upload of the request configuration data.



## 7.4.1.6 Exceptions

### 7.4.1.6.1 operationFailed

Exception Name	Definition
operationFailed	<b>Condition:</b> Pre-condition is false or post-condition is false. <b>Returned information:</b> The output parameter status. <b>Exit state:</b> Entry state.

## 7.5 bulkCmActive Interface

### 7.5.1 Operation download (M)

#### 7.5.1.1 Definition

An IRPManager invokes this operation to request an IRPAgent to download and administer a file containing bulk configuration data (clause 10). The IRPAgent obtains the configuration data file from the indicated globally unique data file reference.

For checks made during download see subclause 7.5.6.

#### 7.5.1.2 Input parameters

Parameter Name	Qualifier	Information type	Comment
sessionId	M	String identifying the session	Identifies this specific session and process associated with the requested bulk data download.
downloadDataFileReference	M		This specifies a globally unique file reference from where the data to be fetched and download from.

#### 7.5.1.3 Output parameters

Parameter Name	Qualifier	Matching Information	Comment
status	M	ENUM(OperationSucceeded, OperationFailed).	indicates (a) start of operation is successful or (b) operation failed because of specified or unspecified reasons

#### 7.5.1.4 Pre-condition

sessionIdle

Assertion Name	Definition
sessionIdle	State as defined in clause 9. The Bulk CM IRP Agent has successfully opened the session and is ready to handle the first operations of the session or repeat this operation.

#### 7.5.1.5 Post-condition

downloadInProgress

Assertion Name	Definition
downloadInProgress	State = UDOWNLOAD_IN_PROGRESS, as defined in clause 9. The Bulk CM IRP Agent has successfully started the download of the configuration data changes.

## 7.5.1.6 Exceptions

### 7.5.1.6.1 operationFailed

Exception Name	Definition
operationFailed	<p><b>Condition:</b> Pre-condition is false or post-condition is false.</p> <p><b>Returned information:</b> The output parameter status.</p> <p><b>Exit state:</b> Entry state.</p>

## 7.5.2 Operation validate (O)

### 7.5.2.1 Definition

An IRPManager invokes this operation to request an IRPAgent to validate previously downloaded bulk configuration data (clause.10), see subclause 7.5.1. Use of this optional operation enables an IRPManager to detect errors with regard to the previously downloaded bulk configuration data before requesting preactivation or activation. See subclause 7.5.6 for scope and types of errors attempted to be detected.

Specifying an activation mode is optional. There can only be one activation mode for a session. If an activation mode is specified for the validate, it shall be when the first validate operation is requested. If an activation mode was specified for the first validate operation, it is not possible to change the activation mode initially specified with any subsequent validate retries. (If another activation mode is required; a new session, download, validate, preactivate and activate should be started.). If no activation mode is specified for the first validate, it cannot be subsequently specified with any subsequent validate retries. (If specification of an activation mode is required; a new session, download, validate, preactivate and activate should be started.) If an activation mode is specified for the validate, it cannot be specified for the preactivation or activation. If no activation mode is specified for the validate operation, it cannot be specified for the preactivation or activation. See also subclauses 7.5.3 and 7.5.4.

Use of the validate operation shall have no influence on the fallback behaviour of a session.

Invoking the validate operation shall not result in any of the suboperations specified in the downloaded bulk configuration data being applied (clause 10). The operation is essentially passive.

### 7.5.2.2 Input parameters

Parameter Name	Qualifier	Information type	Comment
sessionId	M	String identifying the session	Identifies this specific session and process associated with the requested bulk data download.
ActivationMode	O		Identifies whether a specific activation mode is required. See also subclauses 7.5.3 and 7.5.4. The valid choices are defined in the parameter table in clause 7.5.4.

### 7.5.2.3 Output parameters

Parameter Name	Qualifier	Matching Information	Comment
status	M	ENUM(OperationSucceeded, OperationFailed).	indicates (a) start of operation is successful or (b) operation failed because of specified or unspecified reasons

### 7.5.2.4 Pre-condition

downLoaded

Assertion Name	Definition
downLoaded	State as defined in clause 9. The Bulk CM IRP Agent has successfully opened the session and download had been attempted or repeat this operation.

### 7.5.2.5 Post-condition

validationInProgress

Assertion Name	Definition
validationInProgress	State = VALIDATE_IN_PROGRESS, as defined in clause 9. The Bulk CM IRP Agent has successfully started the validation of the downloaded configuration data.

### 7.5.2.6 Exceptions

#### 7.5.2.6.1 operationFailed

Exception Name	Definition
operationFailed	<p><b>Condition:</b> Pre-condition is false or post-condition is false.</p> <p><b>Returned information:</b> The output parameter status.</p> <p><b>Exit state:</b> Entry state.</p>

## 7.5.3 Operation preactivate (O)

### 7.5.3.1 Definition

An IRPManager invokes this operation to request an IRPAgent to preactivate previously downloaded bulk configuration data (clause 10) that may have optionally been validated (subclause 7.5.3). The principal, but not mandatory, functions of the preactivate operation is to validate the configuration data changes in the context of current operational data and to pre-process the configuration data changes. Use of this optional operation enables the IRPManager to prepare the activation of the downloaded bulk configuration data at the EM or NE level before requesting its effective activation. The actions shall fall short of executing the bulk configuration data changes (clause 10) in the network and impacting service. (The actions may for example be to validate the configuration data changes in the context of current operational data or to pre-process the configuration data changes). Performing such actions prior to activate may help identify any potential problems prior to executing the changes on a live a network and may minimise activation elapse time. See also subclause 7.5.6 for scope of checks during a session and specifically for preactivate.

Specifying an activation mode is optional. There can only be one activation mode for a session. If an activation mode is specified for the preactivation, it shall be when the first preactivate or validate operation is requested. If an activation mode was specified by validate it is not possible to change the activation mode initially specified with any subsequent preactivate or activate operations. If an activation mode was specified for the first preactivate operation, it is not possible to change the activation mode initially specified with any subsequent preactivate retries, activate or activate retries. (If another activation mode is required, a new session, download, validate, preactivate and activate should be started.) If no activation mode is specified for the first preactivate, it cannot be subsequently specified with any subsequent preactivate retries, activation or activation retries. (If specification of an activation mode is required, a new session, download, validate, preactivate and activate should be started.) See also subclauses 7.5.2 and 7.5.4.

See subclause 6.2.4.3 for description of optional verification mode parameter and associated checking.

Selecting a fallback option is optional. There can only be one fallback option for a session.

If the option is selected it shall be initiated when the first preactivation operation is requested. If a fallback option is not requested for the first preactivation, it cannot be subsequently requested for repeated preactivations or activations during the session. If the fallback option was requested, it is not possible to change the fallback option initially selected with any subsequent re- preactivate retries i.e. for a session it is only possible to fallback to the configuration that existed when the first preactivate operation was requested. See also clause 7.5.5. (If a new fallback configuration is required a new session, download, activate and preactivate should be started. The old session can be ended, prior to which fallback can optionally be invoked).

Specifying how preactivate operation retries within a session shall be implemented following a partially successful preactivation (e.g. repeat all preactivation management actions or just the uncompleted delta of management actions that did not previously complete successfully) is beyond the scope of this document. Only the IRPManager can initiate preactivate retries. (The IRPAgent shall not initiate retries autonomously).

### 7.5.3.2 Input parameters

Parameter Name	Qualifier	Information type	Comment
sessionId	M	String identifying the session	Identifies this specific session and process associated with an earlier bulk data download that is required to be activated.
VerificationMode	O		Selects the mode of checking. One of two choices may be selected: "full checking", "limited checking", see subclause 7.5.6.3.
activationMode	O		Identifies whether a specific activation mode is required. See also subclauses 7.5.2 and 7.5.4. The valid choices are defined in the parameter table in clause 7.5.4.
fallbackEnabled	M		Indicates whether or not it is required to initialise and enable fallback option prior to the preactivation. This option is only open for the first preactivate operation of a session. For any subsequent preactivate operation retries within a session the fallbackEnabled parameter must be set to indicate it is not required to initialise fallback otherwise the pre-activate operation retry shall fail.

### 7.5.3.3 Output parameters

Parameter Name	Qualifier	Matching Information	Comment
status	M	ENUM(OperationSucceeded, OperationFailed).	indicates (a) start of operation is successful or (b) operation failed because of specified or unspecified reasons

### 7.5.3.4 Pre-condition

Assertion Name	Definition
downLoaded	State as defined in clause 9. The Bulk CM IRP Agent has successfully opened the session and download had been attempted or repeat this operation.

### 7.5.2.5 Post-condition

preactivationInProgress

Assertion Name	Definition
preactivationInProgress	State = PREACTIVATION_IN_PROGRESS, as defined in clause 9. The Bulk CM IRP Agent has successfully started the validation of the downloaded configuration data.

### 7.5.3.6 Exceptions

#### 7.5.3.6.1 operationFailed

Exception Name	Definition
operationFailed	<b>Condition:</b> Pre-condition is false or post-condition is false. <b>Returned information:</b> The output parameter status. <b>Exit state:</b> Entry state.

## 7.5.4 Operation activate (M)

### 7.5.4.1 Definition

An IRPManager invokes this operation to request an IRPAgent to activate previously downloaded bulk configuration data (clause 10) that may have optionally been checked (subclause 7.5.2 ) and/or been preactivated (subclause 7.5.3). Activate means that operations specified in a previously downloaded configuration data file, for example create, delete and modify of managed objects are carried out on the live network i.e. mobile subscribers are affected by the downloaded configuration data.

An IRPAgent may support an optional activationMode parameter. This enables the IRPManager to indicate to the IRPAgent the preference for how the activation shall be executed. One of two options may be selected: "least service impact" or "least elapse time". If the "least service impact" option is selected the IRPAgent shall optimise the execution of the activation in a way that minimises disruption to network services. Elapse time to complete the activation is of secondary importance. If the "least elapse time" option is selected the IRPAgent shall optimise the execution of the activation in a way that minimises the elapse time for completing the execution of the activation. During the execution, disruption of network services is of secondary importance.

See subclause 7.5.6 for descriptions of checks made during activate execution.

Specifying an activation mode is optional. There can only be one activation mode for a session. If an activation mode is specified for the activation, it shall be when the first activate, validate or preactivate operation is requested. If an activation mode was specified by validate or preactivate operations, it is not possible to change the activation mode initially specified with any subsequent activate operations. If an activation mode was specified for the first activate, it is not possible to change the activation mode initially specified with any subsequent activate retries. (If another activation mode is required, a new session, download, validate, preactivate and activate should be started.) If no activation mode is specified for the first activate, it cannot be subsequently specified with any subsequent activate retries. (If specification of an activation mode is required, a new session, download, validate, preactivate and activate should be started.) See also subclauses 7.5.2 and 7.5.3.

If a preactivation had been invoked, successful completion of activate should release any internal local resources allocated for the preactivation.

Selecting a fallback option is optional. There can only be one fallback option for a session.

If the fallback option is selected it shall be initiated when the first activation or preactivation operation is requested. If a fallback option is not requested for the first activation or preactivation, it cannot be subsequently requested for repeated activations or an activation following a preactivation during the session. If the fallback option was requested, it is not possible change the fallback option initially selected with any subsequent re-activate retries or an activation following a preactivation i.e. for a session it is only possible to fallback to the configuration that existed when the first activate or preactivate operation was requested. See also subclause 7.5.5. (If a new fallback configuration is required a new session, download and activate should be started. The old session can be ended, prior to which fallback can optionally be invoked).

Specifying how activate operation retries within a session shall be implemented following a partially successful activation (e.g. repeat all activation management actions or just the uncompleted delta of management actions that did not previously complete successfully) is beyond the scope of this document. Only the IRPManager can initiate activate retries. (The IRPAgent shall not initiate retries autonomously).

#### 7.5.4.2 Input parameters

Parameter Name	Qualifier	Information type	Comment
sessionId	M	String identifying the session	Identifies this specific session and process associated with an earlier bulk data download that is required to be activated.
activationMode	O		Identifies whether a specific activation mode is required. See also subclauses 7.5.2 and 7.5.3. It may be set to indicate "least service impact" or "least elapse time" types of activation are required.
fallbackEnabled	M		Indicates whether or not it is required to initialise and enable fallback option prior to the activation. This option is only open for the first activate operation of a session. For any subsequent activate operation retries within a session the fallbackEnabled parameter must be set to indicate it is not required to initialise fallback otherwise the activate operation retry shall fail.

#### 7.5.4.3 Output parameters

Parameter Name	Qualifier	Matching Information	Comment
status	M	ENUM(OperationSucceeded, OperationFailed).	indicates (a) start of operation is successful or (b) operation failed because of specified or unspecified reasons

### 7.5.4.4 Pre-condition

downLoaded

Assertion Name	Definition
downLoaded	State as defined in clause 9. The Bulk CM IRP Agent has successfully opened the session and download had been attempted or repeat this operation.

### 7.5.4.5 Post-condition

activationInProgress

Assertion Name	Definition
activationInProgress	State = ACTIVATE_IN_PROGRESS, as defined in clause 9. The Bulk CM IRP Agent has successfully started the activation of the downloaded configuration data.

### 7.5.4.6 Exceptions

#### 7.5.4.6.1 operationFailed

Exception Name	Definition
operationFailed	<b>Condition:</b> Pre-condition is false or post-condition is false. <b>Returned information:</b> The output parameter status. <b>Exit state:</b> Entry state.

## 7.5.5 Operation fallback (M)

### 7.5.5.1 Definition

An IRPManager invokes this operation to request an IRPAgent to recover after a previously ordered activation or preactivation has failed.

If a fallback is requested after a preactivation but before an activation the IRPAgent should as necessary return any internal local resources impacted by the preactivation back to the same state they were in prior to the preactivation being invoked. There is no impact to the operational network resources as the activate operation has not been invoked.

If fallback is requested after an activation the IRPAgent shall instigate activating the fallback area to restore the operational network resources impacted by the configuration changes for the session back to the configuration they were in when the fallback option was selected during the session. If a preactivation was also performed, as necessary the IRPAgent should return any internal local resources impacted by the preactivation back to the same state they were in prior to the preactivation being invoked.

Specifying how fallback operation retries within a session shall be implemented after a fallback fails (e.g. repeat all fallback functions or just the delta of fallback functions that did not previously complete successfully) is beyond the scope of this document. Only the IRPManager can initiate the fallback operation. The IRPAgent shall not initiate fallback or fallback retries autonomously. Within a session the fallback operation shall only be accepted if an initial activate or preactivate operations was performed with fallback option enabled. For further discussion of enabling or not the fallback option see subclause 7.5.4.

### 7.5.5.2 Input parameters

Parameter Name	Qualifier	Information type	Comment
SessionId	M	String identifying the session	Identifies this specific session and process associated with an earlier bulk data operation e.g. upload or download for which the current log is required.

### 7.5.5.3 Output parameters

Parameter Name	Qualifier	Matching Information	Comment
Status	M	ENUM(OperationSucceeded, OperationFailed).	indicates (a) start of operation is successful or (b) operation failed because of specified or unspecified reasons

### 7.5.5.4 Pre-condition

fallbackEnabled

Assertion Name	Definition
fallbackEnabled	State as defined in clause 9. The Bulk CM IRP Agent has successfully opened the session and fallbackEnables=True by either, preactivate or activate operations being successfully invoked, as defined in subclauses 7.5.3 and 7.5.4.

### 7.5.5.5 Post-condition

fallbackInProgress

Assertion Name	Definition
fallbackInProgress	State = FALLBACK_IN_PROGRESS, as defined in clause 9. The Bulk CM IRP Agent has successfully started the fallback.

### 7.5.5.6 Exceptions

#### 7.5.5.6.1 operationFailed

Exception Name	Definition
operationFailed	<b>Condition:</b> Pre-condition is false or post-condition is false. <b>Returned information:</b> The output parameter status. <b>Exit state:</b> Entry state.

## 7.5.6 Validation and Checking Functions

### 7.5.6.1 Download Checks

During download the IRPAgent should check the consistency of imported configuration data against the data schema to ensure there are no errors. The IRPAgent is not required to check the semantic of the downloaded bulk configuration data during the download.

### 7.5.6.2 Validate Checks

During validation the IRPAgent should check the syntax and semantic of previously downloaded bulk configuration data.

### 7.5.6.3 Preactivation Checks

During preactivation the IRPAgent should check the semantic of previously downloaded bulk configuration data, and must also check the syntax if a validate operation has not previously been successfully performed.

An Element Manager should, if technically feasible, send the configuration data changes to all Network Elements (NE) for the NE to verify, to the extent possible, that the activate will successfully execute the configuration data changes. If any elements of configuration change data that will not successfully execute are identified, diagnostic data identifying the NEs and failing configuration data elements will be made available to the Manager.

An IRPAgent may support an optional verification mode parameter, see clause 7.5.2. When the IRPManager does not require extensive checking, this parameter may be used to constrain the scope of validation to avoid performing checks that potentially may require extensive real time to execute, for example checks actively involving entities outside the

IRPAgent such as NE's. The validation mode parameter has two values: "full checking" and "limited checking". In the "full checking" mode, the checking should be as complete as possible with the intent of achieving the greatest assurance that the subsequent activation operation will be successful. In the "limited checking" mode, checking that can be performed by the IRPAgent rapidly is still performed, but further checking that may cause significant delays to execute should be omitted.

#### 7.5.6.4 Activate Checks

During the activation the same checks as for validate and preactivate should be performed if these operations have not previously been successfully performed. These checks may also be repeated if the context may have changed.

## 8 Bulk Configuration Data File

### 8.1 Interface BulkCmIRPNotifications#1

#### 8.1.1 Notification notifySessionStateChanged (M)

##### 8.1.1.1 Definition

The IRPAgent notifies the IRPManager that a state change has occurred on a bulk -configuration data file sessionId operation subscribed to by the IRPManager. E.g. a configuration data file is available for processing after an upload, a download is complete. See clause 9 for a further description of states.

Parameter Name	Qualifiers	Matching Information	Comment
objectClass	O,F	ManagedEntity.objectClass	Notification header - see [3].
objectInstance	O,F	ManagedEntity.objectInstance.	Notification header - see [3].
notificationId	O	This carries the semantics of notification identifier.	Notification header - see [3].
eventTime	M,F	ManagedEntity.creationTime	Notification header - see [3].
systemDN	O,C,F	IRPAgent.systemDN where the IRPAgent is related to the KernelCmIRP.	Notification header - see [3].
NotificationType	M,F	Mapped to notificationType in [3].	Notification header - see [3]. For this notification it indicates notification type is Notify Session State Changed.
sessionId	M	String identifying the session	Identifies this specific session and process associated with an earlier bulk data operation e.g. upload or download for which the current status is required.
sourceIndicator	O		This parameter, when present, indicates the source of the operation that led to the generation of this notification. It can have one of the following values: resource operation: The notification was generated in response to an internal operation of the resource; management operation: The notification was generated in response to a management operation applied across the managed object boundary external to the managed object; unknown: It is not possible to determine the source of the operation.
sessionState	M	ENUM(Upload Failed, Upload Completed, Download Failed, Download Completed, Validation Failed, Validation Completed, Preactivation Failed, Preactivation Partly Realised, Preactivation Completed, Activation Failed, Activation Partly Realised, Activation Completed, Fallback Failed, Fallback Partly Realised, Fallback Completed)	Indicates the state transition that caused the Notification. See clause 9. i.e. Upload Failed, Upload Completed, Download Failed, Download Completed, Validation Failed, Validation Completed, Preactivation Failed, Preactivation Partly Realised, Preactivation Completed, Activation Failed, Activation Partly Realised, Activation Completed, Fallback Failed, Fallback Partly Realised, Fallback Completed. (Note: as per sub-clause 9.2 "in-progress" transition states are not notified)



### 8.1.1.3 Triggering events

State transitions as defined in clause 9.

## 8.1.1 Notification NotifyGetSessionLogEnded (M)

### 8.1.1.1 Definition

The IRPAgent notifies the IRPManager that a requested GetSessionLog for a bulk data configuration file sessionId operation subscribed to by the IRPManager has ended successfully or unsuccessfully.

### 8.1.1.2 Input parameters

Parameter Name	Qualifiers	Matching Information	Comment
objectClass	O,F	ManagedEntity.objectClass	Notification header - see [3].
objectInstance	O,F	ManagedEntity.objectInstance.	Notification header - see [3].
notificationId	O	This carries the semantics of notification identifier.	Notification header - see [3].
eventTime	M,F	ManagedEntity.creationTime	Notification header - see [3].
systemDN	O,C,F	IRPAgent.systemDN where the IRPAgent is related to the KernelCmIRP.	Notification header - see [3].
NotificationType of notificationHeader	M,F	Mapped to notificationType in [3]	Notification header - see [3]. For this notification it indicates notification type is Notify Bulk CM Log State.
SessionId	M	String identifying the session	Identifies this specific session and process associated with an earlier bulk data operation e.g. upload or download for which Log State is required.
SourceIndicator	O		This parameter, when present, indicates the source of the operation that led to the generation of this notification. It can have one of the following values: resource operation: The notification was generated in response to an internal operation of the resource; management operation: The notification was generated in response to a management operation applied across the managed object boundary external to the managed object; unknown: It is not possible to determine the source of the operation.
SessionLogStatus	M	Boolean = GetSessionLog completed successfully or GetSessionLog completed unsuccessfully	Indicates event that caused the Notification i.e. GetSessionLog completed successfully, GetSessionLog completed unsuccessfully.

### 8.1.1.3 Triggering event

Attempt to transfer session log to destinations completed successfully or failed. Session state independent, see clause 9.

---

## 9 State Machine

### 9.1 State Machine Overview

The Bulk CM IRPAgent state machine satisfies the following general requirements and characteristics for Bulk CM IRP:

- 1) Each configuration session is associated with one state machine. The session is identified by the sessionId. If a session is started (startSession operation) an instance of the state machine is created. If the session is ended (endSession operation) the instance of the state machine is deleted.

- 2) Under normal operation without errors the IRPManager is able to supervise a configuration session by just monitoring the state change notifications (notifySessionStateChanged) triggered by the IRPAgent
- 3) Under abnormal conditions where the IRPManager is not notified of a change, the getSessionStatus operation can be invoked to determine current state of the session. The IRPManager does not need to maintain a history of the state machine.
- 4) On the IRPAgent there is only one download configuration data file (clause 10) associated with a session at a time.
- 5) Multi configuration session must be supported by the IRPAgent. E.g. it must be possible to invoke an upload session in parallel with an active activate session.
- 6) The IRPAgent resolves concurrency problems on a "first come - first serve" basis. E.g. an upload and an activation requested on the same configuration data cannot be performed at the same time and in this case the first will be progress to completions and the second request rejected.
- 7) It must be possible to abort a configuration session within a transition state.
- 8) The operator/IRPManager decides on whether or not enabling the fallback option is required before requesting an activation or preactivation. Enabling the fallback option will maintain the disposition of the configuration before the activation or preactivation. The fallback configuration information is established at point before the first activation or preactivation is started. If there are multiple activation or preactivation attempts during a session only one (first) fallback configuration is maintained.
- 9) The session log file can be requested in any state. The uploaded log file contains information which is specific to the configuration session.
- 10) Clause 7.3 defines the valid state machine pre and post conditions for each operation.

## 9.2 State Machine Description

The IRPAgent progresses Bulk CM operations and associated configuration data changes (clause 10) within a session according to the state machine defined here. The IRPManager can manage a configuration session using session state change notifications which are triggered by the IRPAgent. Not all state changes defined here are notified to the IRPManager. The transition states (UPLOAD\_IN\_PROGRESS, DOWNLOAD\_IN\_PROGRESS, VALIDATION\_IN\_PROGRESS, PREACTIVATION\_IN\_PROGRESS, ACTIVATION\_IN\_PROGRESS) are not notified to the IRPManager as they are not required.

If the IRPManager becomes unaware or needs to confirm the current state of a configuration session it can request this by invoking getSessionStatus operation. It is not required to know the history of the state machine. The getSessionStatus operation will provide the "actual" current status.

An IRPManager may request the status when it detects loss of control, for example because of the following reasons:

- 1) Session state change notifications are not being received as expected, e.g. because IRPAgent is blocked in a transition state, e.g. ACTIVATION\_IN\_PROGRESS;
- 2) IRPManager gets disconnected from the IRPAgent, e.g. session state notification are not received.

The session state notification events are considered a subset of the state machine (without transition state). The actual configuration state can be requested via getSessionStatus. Because of this common behaviour it is reasonable to define one interface type for the state machine handling which is used in the session state notification and in the getSessionStatus operation.

The IRPManager will only receive notifications if it registered itself at the IRPAgent with the subscribe operation.

For ease of description the state machine of a configuration session is introduced with the notion of substate machines but state itself are named unique. This kind of notion is not to be interpreted as providing implementation directions.

Within the description of the substate machines it is becoming clear that they have the following state symmetries:

- The state of the UPLOAD\_PHASE, the DOWNLOAD\_PHASE and the VALIDATION\_PHASE are similar.
- The state of the ACTIVATION\_PHASE, PREACTIVATION\_PHASE and the FALLBACK\_PHASE are similar.

The startSession operation creates a state machine. The initial state of the configuration session in the IDLE\_PHASE is IDLE. The endSession deletes a state machine which is not in a transition state, more details are defined in the substate machines.

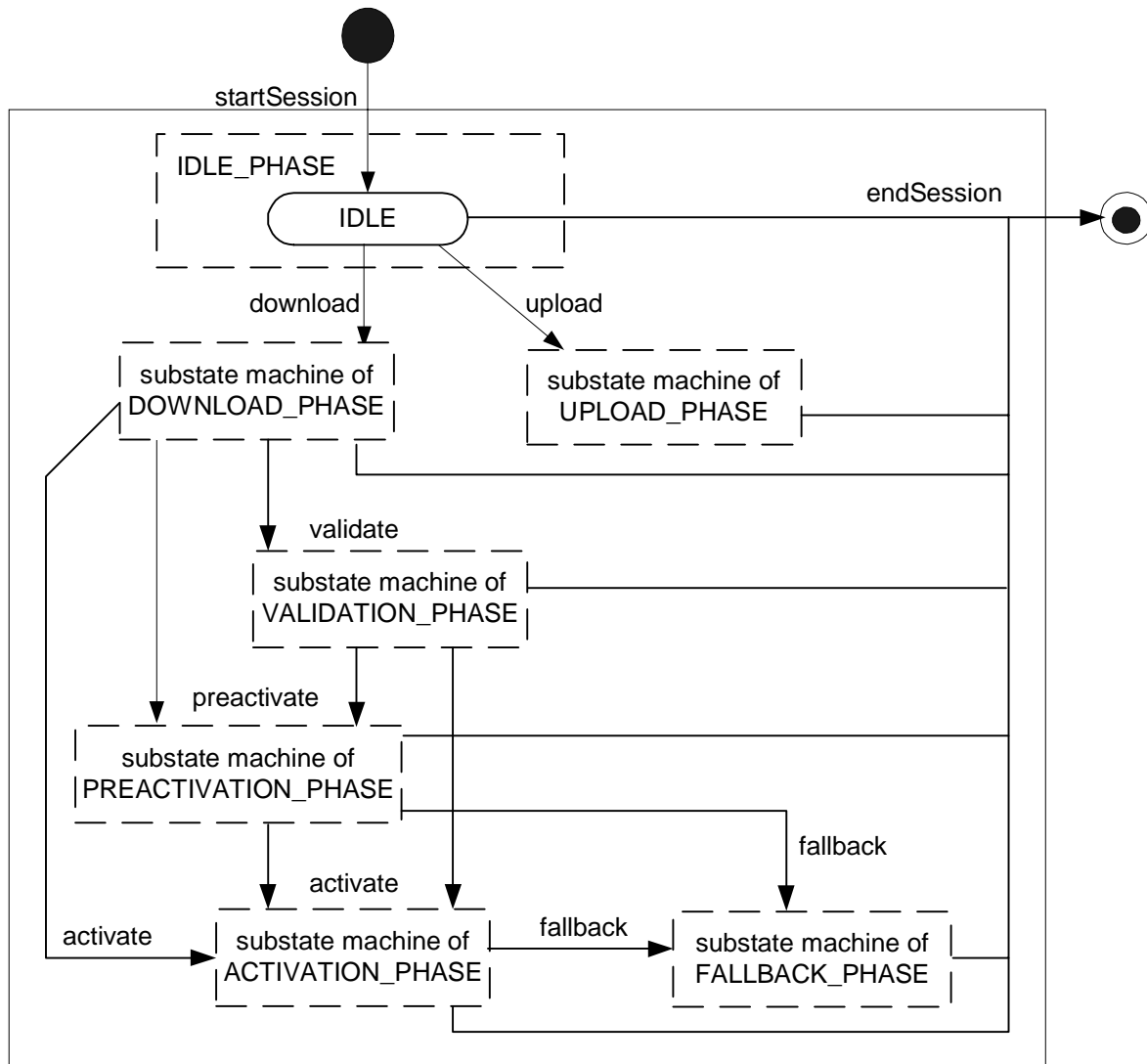


Figure 5: State Machine

The following figures describe the substate machine of a configuration session. The transition states, DOWNLOAD\_IN\_PROGRESS, UPLOAD\_IN\_PROGRESS, VALIDATION\_IN\_PROGRESS, PREAMTIVATION\_IN\_PROGRESS and ACTIVATION\_IN\_PROGRESS, are either left implicit if the IRP Agent finished the processing or explicit via an abortSessionOperation operation from the IRPManager.

In these figures solid transition lines indicate the transition is caused by an external event and dashed transition lines indicate the transition is caused by an internal event or decision as depicted in figure 6.

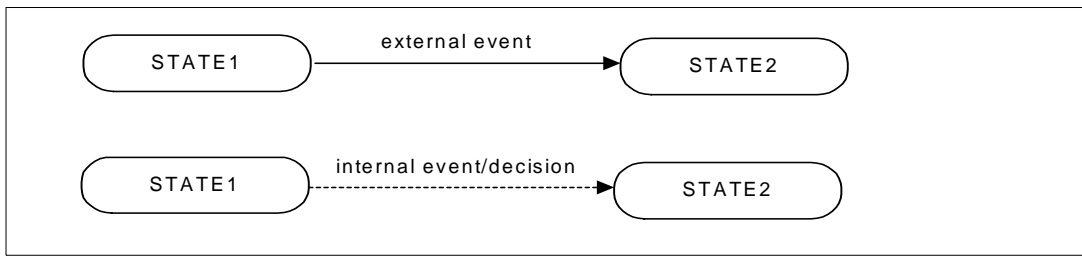


Figure 6: Depicting State Transition Lines for Internal and External Events and Decision

### 9.2.1 Upload Phase

When the upload is triggered the IRP Agent writes the requested configuration data into a configuration data file and copies to the file reference provided by the IRP Manager. If the process succeeds the state UPLOAD\_COMPLETED is indicated. If the upload fails a retry can be triggered in state UPLOAD\_FAILED.

Once a session is associated with an upload none of the other state changes phases outside of the upload phase, i.e., download, validate, preactivate and activate phases cannot be triggered for the session.

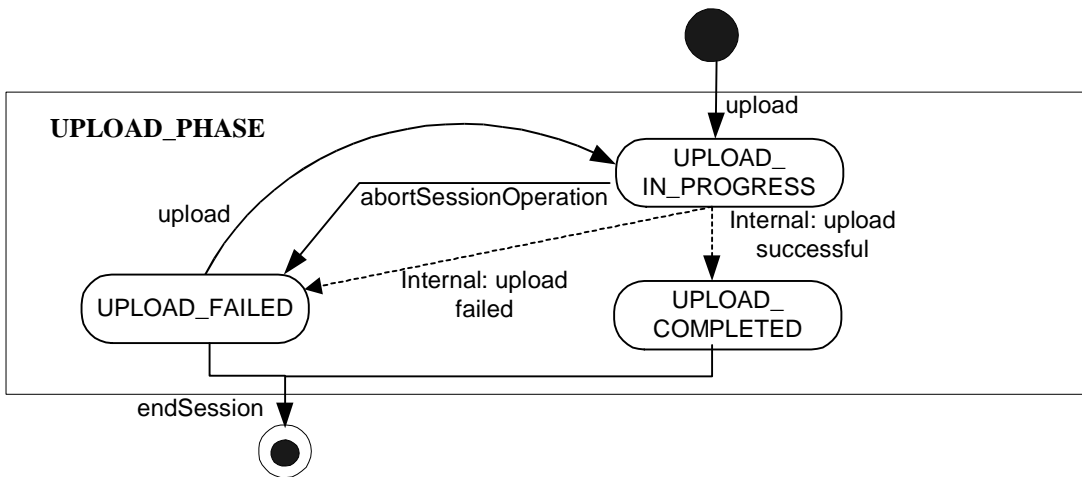


Figure 7: Substate Machine – UPLOAD\_PHASE

### 9.2.2 Download Phase

When the download is triggered the IRP Agent copies the configuration data file (clause 10) from a given file area. The file is parsed and validated. If valid the state DOWNLOAD\_COMPLETED is indicated. If the download fails a retry can be triggered in state DOWNLOAD\_FAILED.

Once a session is associated with a download/validate/preactivate/activation behaviour then an upload phase cannot be triggered within this session.

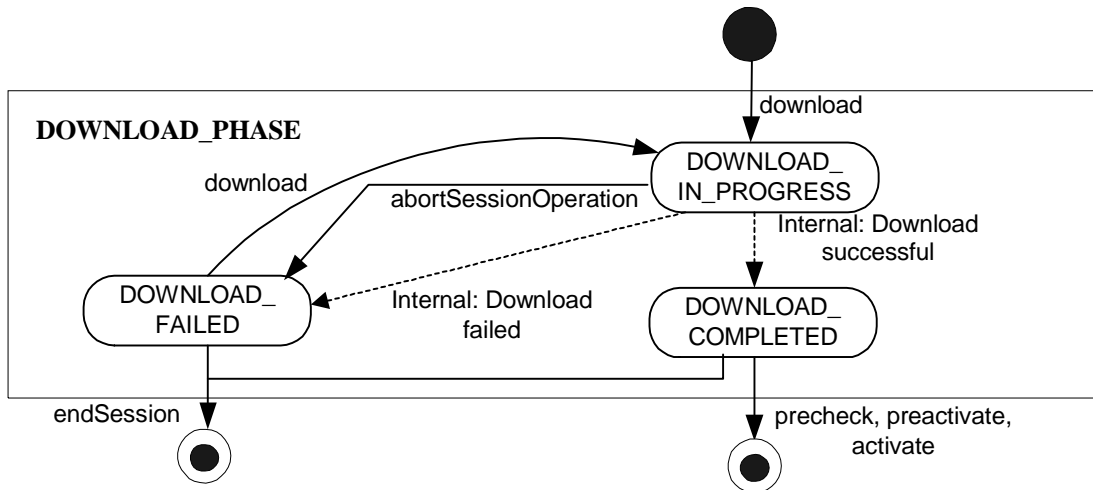


Figure 8: Substate Machine – DOWNLOAD\_PHASE

### 9.2.4 Validation Phase

After a download had been completed the configuration data can be semantically validated before being preactivated or activated into the real subnetwork of an IRP Agent. (see subclause 7.5.6.2). A best effort strategy shall be applied. If validation was successful the state VALIDATION\_COMPLETED is indicated. If the validate fails a retry can be triggered in state VALIDATION\_FAILED.

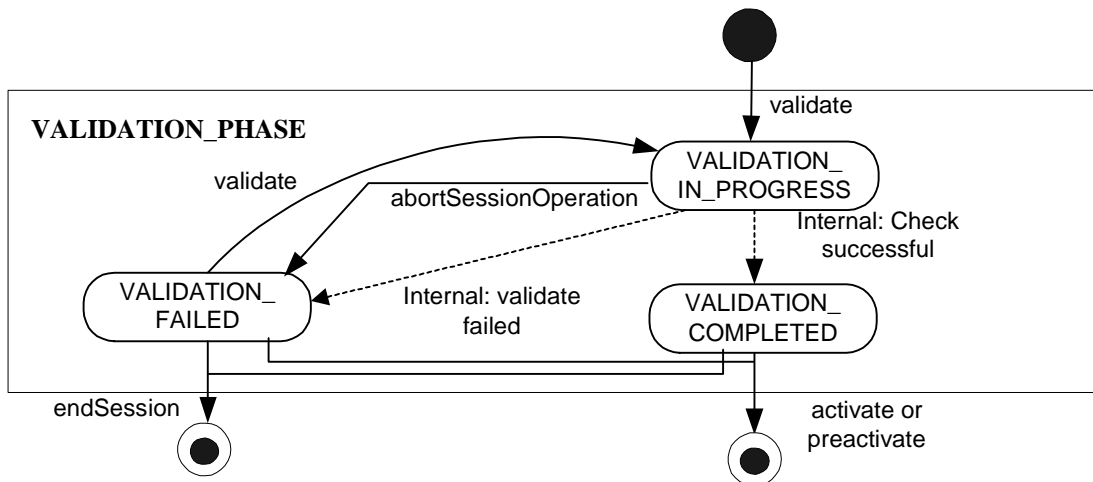


Figure 11: Substate Machine – VALIDATION\_PHASE

### 9.2.5 Preactivation Phase

After a download had been completed and optionally validated the configuration data can be preactivated before being activated into the real subnetwork of an IRPAgent. If the process fully succeeds the preactivation is completed.

For preactivation a best effort strategy shall be employed.

If the IRPAgent is unable to successfully complete all pre-MIB changes that were actioned in the configuration data file (clause10) the state PREACTIVATION\_PARTLY\_REALISED is indicated. This state is not an error condition because the preactivation of configuration data changes follows a best effort strategy. If the preactivation fails completely i.e. there are no pre-MIB changes the state PREACTIVATION\_FAILED is indicated. A retry of the preactivate can be performed in states PREACTIVATION\_PARTLY\_REALISED and PREACTIVATION\_FAILED. The PREACTIVATION\_FAILED state cannot be entered if previously during the session the state had become PREACTIVATION\_PARTLY\_REALISED. The PREACTIVATION\_PARTLY\_REALISED state should be re-entered instead. A retry of the preactivate is allowed so that it is possible to recover after transient condition that caused an preactivate to fail or partly realise are no longer present.

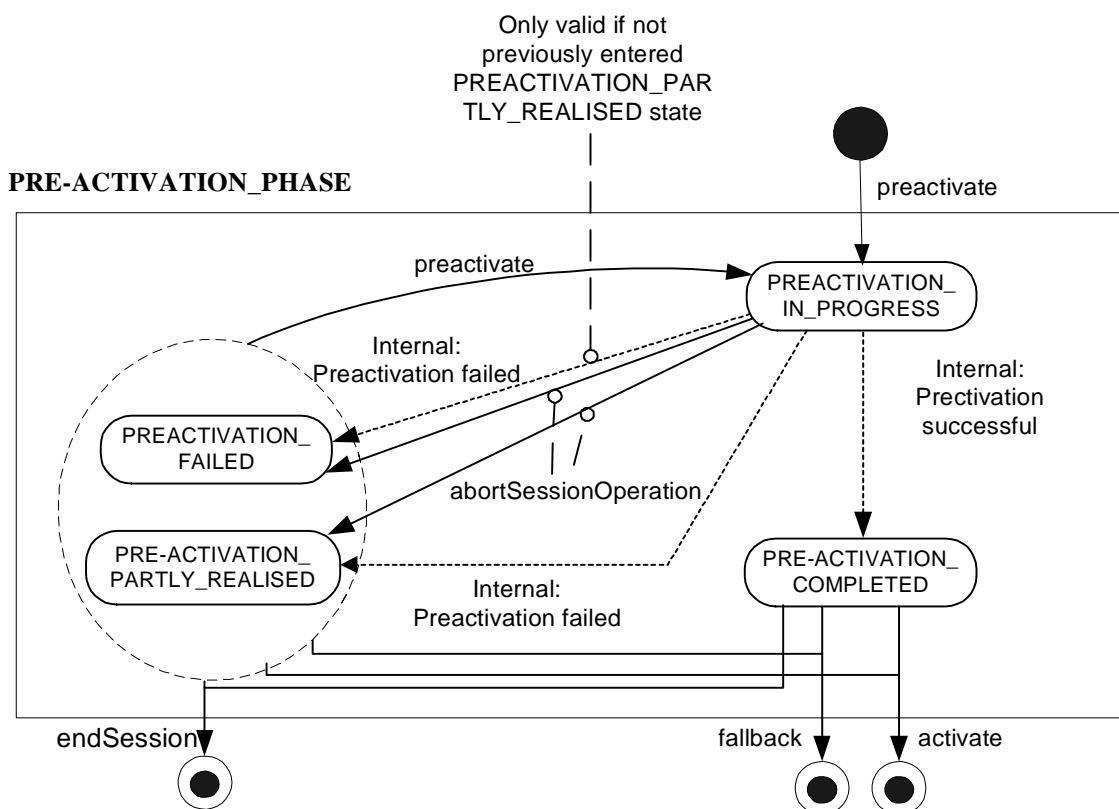


Figure 12: Substate Machine – PREACTIVATION\_PHASE

### 9.2.6 Activation Phase

After a download has been completed and optionally validated and/or preactivated the configuration data can be activated into the real subnetwork of an IRPAgent. If the process fully succeeds the activation is completed.

For activation a best effort strategy shall be employed.

If the IRPAgent is unable to successfully complete all MIB changes and corresponding changes in the network elements that were actioned in the configuration data file (clause 10) the state ACTIVATION\_PARTLY\_REALISED is indicated. This state is not an error condition because the activation of configuration data changes follows a best effort strategy. If the activate fails completely i.e. there are no MIB changes or corresponding changes in the network elements, the state ACTIVATION\_FAILED is indicated. A retry of the activate can be performed in states



### 9.2.7 Fallback Phase

If an activate or preactivate operation was requested with the fallback option enabled and was successfully or partially completed then a fallback operation can be requested. If the process of a fallback fully succeeds then the related MIB and subnetwork is reverted back to its former configuration prior to first configuration data file preactivation or activation of a session.

For fallback a best effort strategy shall be employed.

In case that not all MIB changes and corresponding changes in the network elements that were actioned in configuration data file (clause 8) were successfully reverted back the state FALLBACK\_PARTLY\_REALISED is indicated. This state is not an error condition as the fallback to the former configuration follows a best effort strategy. If the fallback fails completely i.e. no MIB changes or corresponding changes in the network elements can be reverted back then the state FALLBACK\_FAILED is indicated. A retry of fallback can be performed in the states FALLBACK\_PARTLY\_REALISED and FALLBACK\_FAILED. The FALLBACK\_FAILED state cannot be entered if previously during the session the state had become FALLBACK\_PARTLY\_REALISED. The FALLBACK\_PARTLY\_REALISED state should be re-entered instead. A retry of the fallback is allowed so that it is possible to recover after transient condition that caused a fallback to fail or partly realise are no longer present.

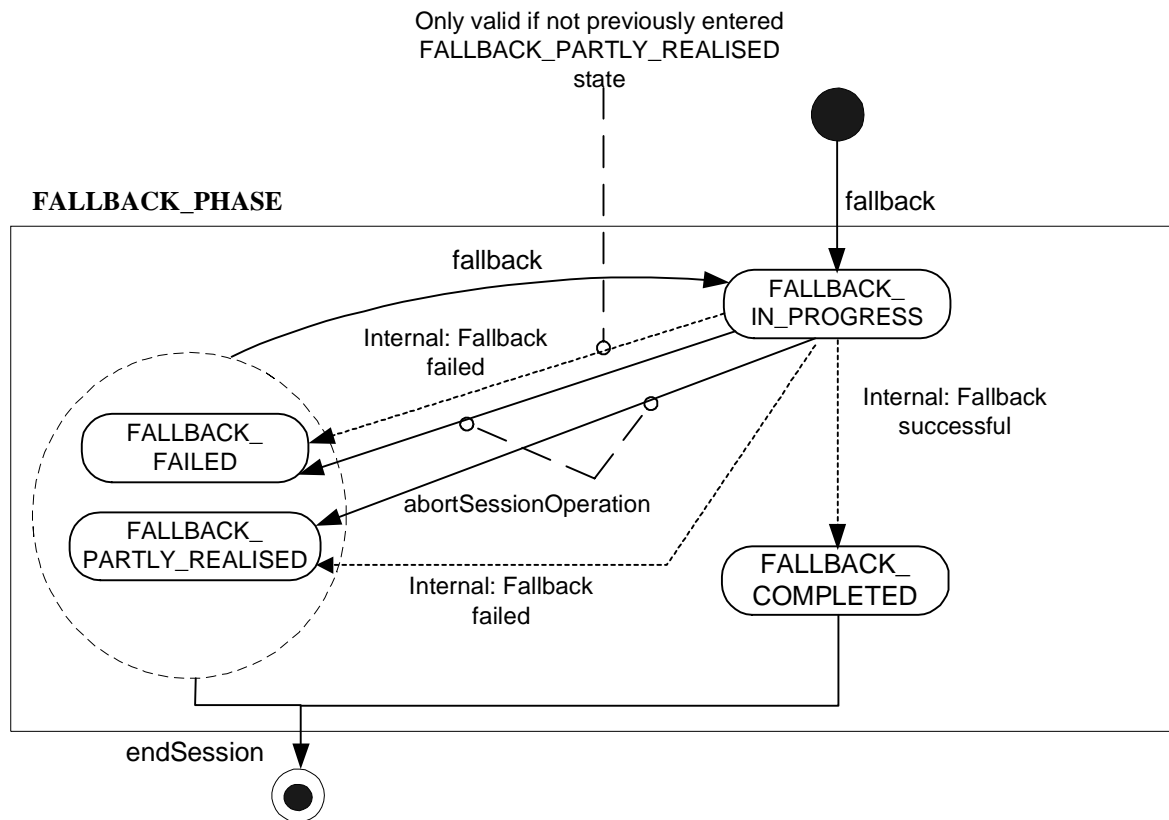


Figure 10: Substate Machine – FALLBACK\_PHASE



## 9.3 State Machine Pre and Post Conditions Tables

For each operation Table 15 identifies the state machine pre and post conditions.

**Table 1: State Machine Pre and Post Conditions**

Operation	Pre-condition	Post Condition
startSession	No state – input sessionId provided by an IRPManager is not already in use in the IRPAgent by this or any other IRPManager	State = IDLE
endSession	not in a Transition status i.e. state <>. * _IN_PROGRESS	sessionId is released - No state.
upload	State = IDLE or UPLOAD_FAILED	Initially while operation is being performed: State= UPLOAD_IN_PROGRESS Finally when operation has completed: State = UPLOAD_COMPLETED or UPLOAD_FAILED
download	State = IDLE or DOWNLOAD_FAILED	Initially while operation is being performed: State= DOWNLOAD_IN_PROGRESS Finally when operation has completed: State = DOWNLOAD_COMPLETED or DOWNLOAD_FAILED
validate	State = DOWNLOAD_COMPLETED or VALIDATION_FAILED	Initially while operation is being performed: State= VALIDATION_IN_PROGRESS Finally when operation has completed: State = VALIDATION_COMPLETED or VALIDATION_FAILED
preactivate	State = DOWNLOAD_COMPLETED or VALIDATION_COMPLETED or PREACTIVATION_PARTLY_REALISED or PREACTIVATION_FAILED	Initially while operation is being performed: State= PREACTIVATION_IN_PROGRESS Finally when operation has completed: State = PREACTIVATION_COMPLETED or PREACTIVATION_PARTLY_REALISED or PREACTIVATION_FAILED
activate	State = DOWNLOAD_COMPLETED or VALIDATION_COMPLETED or ACTIVATION_PARTLY_REALISED or ACTIVATION_FAILED or PREACTIVATION_COMPLETED or PREACTIVATION_PARTLY_REALISED or PREACTIVATION_FAILED	Initially while operation is being performed: State= ACTIVATION_IN_PROGRESS Finally when operation has completed: State = ACTIVATION_COMPLETED or ACTIVATION_PARTLY_REALISED or ACTIVATION_FAILED
fallback	State = PREACTIVATION_COMPLETED or PREACTIVATION_PARTLY_REALISED or ACTIVATION_COMPLETED or ACTIVATION_PARTLY_REALISED or FALLBACK_PARTLY_REALISED or FALLBACK_FAILED or FALLBACK_PARTLY_REALISED or FALLBACK_FAILED	Initially while operation is being performed: State= FALLBACK_IN_PROGRESS Finally when operation has completed: State = FALLBACK_COMPLETED or FALLBACK_PARTLY_REALISED or FALLBACK_FAILED
abortSessionOperation	State = UPLOAD_IN_PROGRESS or DOWNLOAD_IN_PROGRESS or VALIDATION_IN_PROGRESS or PREACTIVATION_IN_PROGRESS or ACTIVATION_IN_PROGRESS or FALLBACK_IN_PROGRESS	State = UPLOAD_FAILED or DOWNLOAD_FAILED or VALIDATE_FAILED or PREACTIVATION_PARTLY_REALISED or PREACTIVATION_FAILED or ACTIVATION_PARTLY_REALISED or ACTIVATION_FAILED or FALLBACK_PARTLY_REALISED or FALLBACK_FAILED
getSessionIds	N/A – State Machine independent	N/A
getSessionStatus	None	None
getSessionLog	None	None
getBulkCmIRPVersion	N/A – State Machine independent	N/A

## 10 Bulk Configuration Data File

The overall management of Bulk CM is controlled by the operations in subclause 7. Unitary management information is aggregated into a configuration data file for bulk CM operations. The file can be used for active and passive CM.

Bulk configuration data files consist of one or more blocks. Each block contains one or more object containment trees defined by a standardised language, for example XML. The basic building block (node) of this tree is a specifically typed MO. This MO is identified by an ID attribute (the Naming attribute used in the RDN), and contains (1) data associated with the MO, and (2) zero or more children nodes. The structure and content of the MO data is constrained by the possible types of contained objects for the CM NRM that is being managed by Bulk CM IRP IS.

The file structure is the same for both upload and download bulk CM operations, apart that for active bulk CM operations, as well as containing MO data the blocks also specify the management actions (sub-operations) associated with each MOs item in the file. The following management actions (sub-operations) on MOs are supported for active bulk CM:

- Create MO. (sub-clause 10.1.1)
- Delete MO. (sub-clause 10.1.2)
- Change one or more existing MO attribute values. (sub-clause 10.1.3)

The rules for ordering management actions in the configuration data file are defined in sub-clause 10.2.

### 10.1 Bulk Configuration Data Management Actions – Sub-operations

By the nature of active Bulk CM IRP, in the download bulk configuration file all sub-operation parameters identified in the following sub-clauses 10.1.1 – 10.1.3 are "input" only. Bulk CM IRP:IS will not generate any explicit notifications or responses for each sub-operation. The resulting session log and output(s) from the associated Bulk CM operations will record and convey the overall result of the sub-operations in the bulk configuration data file. The IRPAgent can record the outcome of relevant sub-operations in the session log. The IRPManager can subsequently get the session log (sub-clause 7.3.6) if it is required to make a detailed analysis.

It should be noted other IRPs can generate notifications as a result of Bulk CM: IS sub-operations if an IRPAgent implements Basic CM IRP. The rules and definitions for these notifications are beyond the scope of this document. The NRMs identified in sub-clause 6.4 and references [4], [5] and [6] give further details of which MOCs may generate Basic CM IRP notifications as a consequence of the sub-operations defined here.

#### 10.1.1 bulkCmCreateMo (Create MO Sub-operation) (M)

The IRPManager associates this sub-operation with an MOI in the configuration data file to request the IRPAgent to create the MOI.

**Table 10.1: bulkCmCreateMo parameters**

Name	Qualifier	Description
objectClass	Input, M	Identifies the NRM MOC within the scope of sub-clause 6.4 that is to be created.
objectInstance	Input, M	Identifies the NRM MOC instance that is to be created.
attributeList	Input, O	Empty, or one or more attribute name and value pairs valid for the MOC. See sub-clause 6.4. If the list is not empty the indicated attributes will be set to their indicated values when the object is created.

## 10.1.2 bulkCmDeleteMo (Delete MO Sub-operation) (M)

The IRPManager associates this sub-operation with an MOI in the configuration data file to request the IRPAgent to delete the MOI.

**Table 10.2: bulkCmDeleteMo parameters**

Name	Qualifier	Description
objectClass	Input, M	Identifies the NRM MOC within the scope of sub-clause 6.3 that is to be deleted.
objectInstance	Input, M	Identifies the NRM MOC instance that is to be deleted.

## 10.1.3 bulkCmChangeMo (Change MO Sub-operation) (M)

The IRPManager associates this sub-operation with an MOI in the configuration data file to request the IRPAgent to change/set one or more attributes of the MOI.

**Table 10.3: bulkCmChangeMo parameters**

Name	Qualifier	Description
objectClass	Input, M	Identifies the NRM MOC within the scope of sub-clause 6.4 that the attributes are to be changed.
objectInstance	Input, M	Identifies the NRM MOC instance for which the attributes are to be changed.
attributeList	Input, M	One or more attribute name and value pairs valid for the MOC. See sub-clause 6.3. The indicated attributes of the MOC instance will be changed/set to their indicated values.

# 10.2 Rules for ordering Management Actions (Sub-operations) in Configuration Data Files

## 10.2.1 Download files

1. The IRP Manager shall enter the management actions into the configuration data file in the order they are to be interpreted and actioned by the IRPAgent following its sequentially step-by-step single pass operation. The IRPManager has overall responsibility for ensuring the correct order of action is given according to the rules in this sub-clause.
2. The IRPAgent shall interpret the management actions in the configuration data file sequentially step-by-step in a single pass operation. The IRPManager has overall responsibility for ensuring the correct order of action is given.
3. The permitted order shall follow NRM hierarchy subtree(s) of the Managed Object instances pertaining to the configuration data file.
4. All delete MOs actions shall precede any Create MOs actions.
5. This document does not specify any limitations on the ordering of change MO attribute actions other than the impacted if the impacted MO does not already exist it needs to be created by a prior create action. The choice of standardised language may recommend or specify some additional constraints e.g. for reasons of efficiency or for compliance with language syntax. Such recommendation and constraints are beyond the scope of this document
6. All necessary MO changes supported by Bulk CM IRP interface-N need to be fully specified in a configuration data file to maintain consistency within the NRM MIB subtree being operated on. (e.g. if an object is to be deleted, all relations and associations shall be removed).
7. All relations to an MO instance shall be removed prior to deleting an MO instance.
8. When part or whole NRM subtree is to be deleted, in the configuration data file the IRPManager shall first action delete of all associated child instances contained in the NRM subtree before actioning delete of MO parents instances i.e. delete actions on MO instances shall be specified in a recursive manner following the NRM hierarchy subtree from the lowest MO instances to the highest MO instances the IRPManager requires to

be deleted. (The IRPAgent will not support autonomous deletion of all MO instance contained in a NRM subtree identified by a single delete action of the highest MO instance of the subtree).

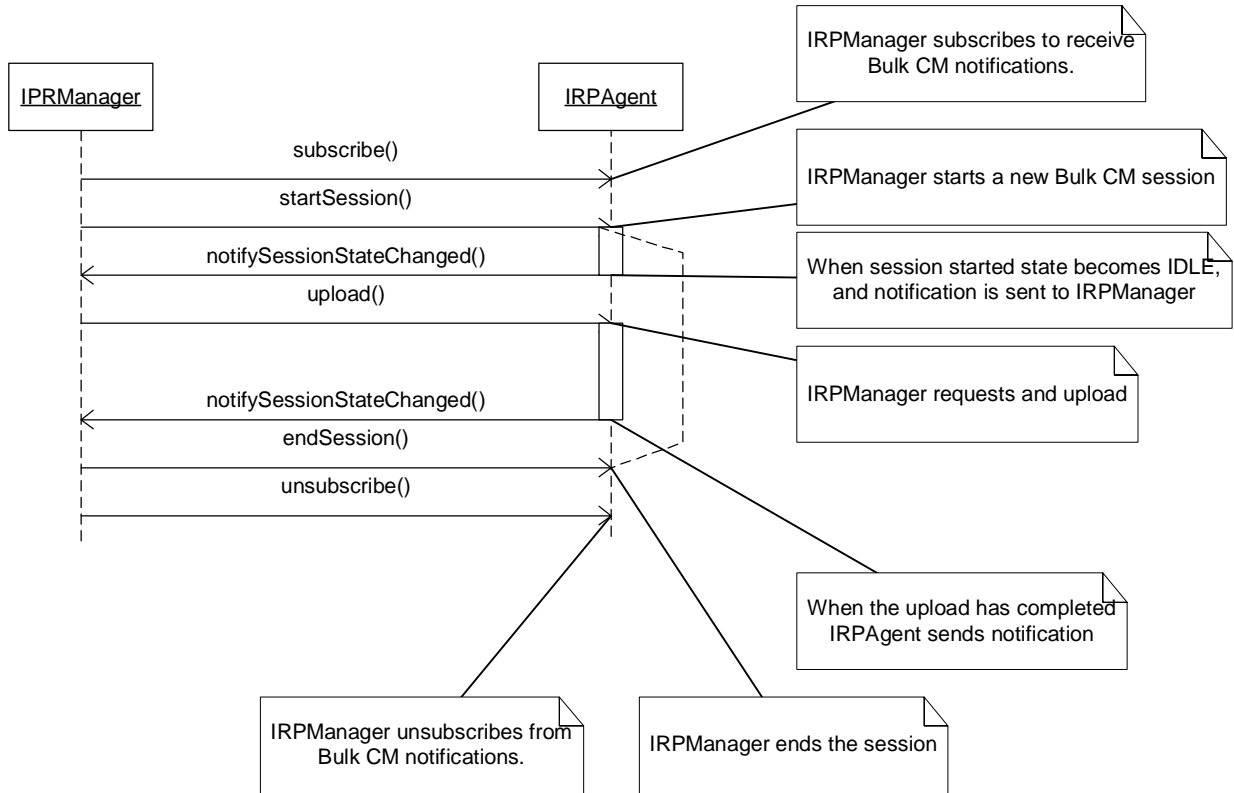
9. When part or a whole NRM subtree is to be created, in the configuration data file the IRPManager shall first action the create action of parents MO instances before actioning the create of any child MO instances contained in the NRM subtree i.e. create actions on MO instances shall be specified in recursive manner following the NRM hierarchy subtree from the highest MO instances to the lowest MO instances the IRPManager requires to be created.

## 10.2.2 Upload files

1. No rules are identified i.e. it is not necessary that they be part of the scope of this document. They may be implementation specific and specified in other document as part of a specific solution.

# Annex A (informative): Scenarios

Supporting background informational only.



**Figure A.1: Example 1: Successful Upload Session**

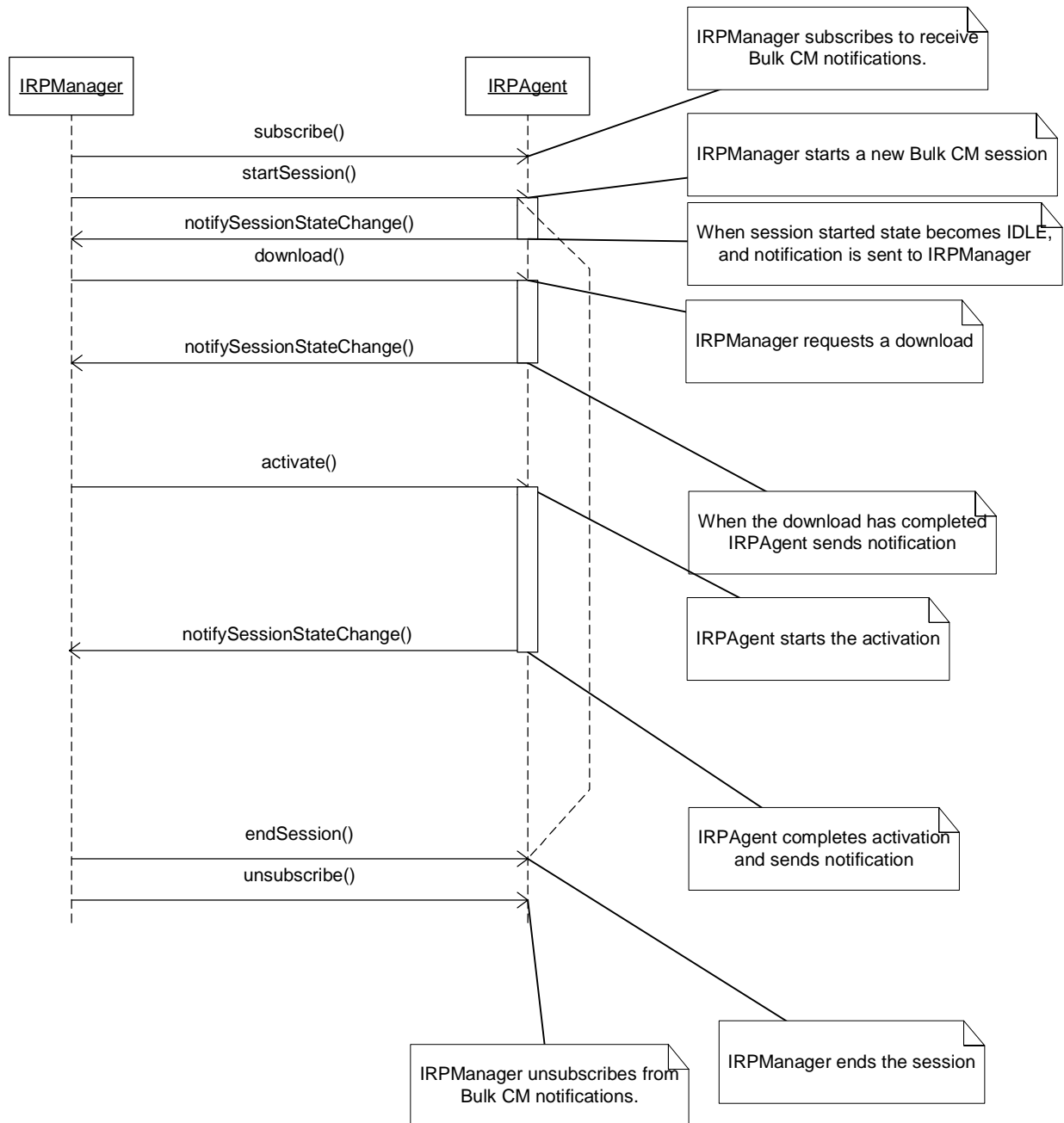


Figure A.2: Example 2: Successful Download and Activation without validation and preactivation.

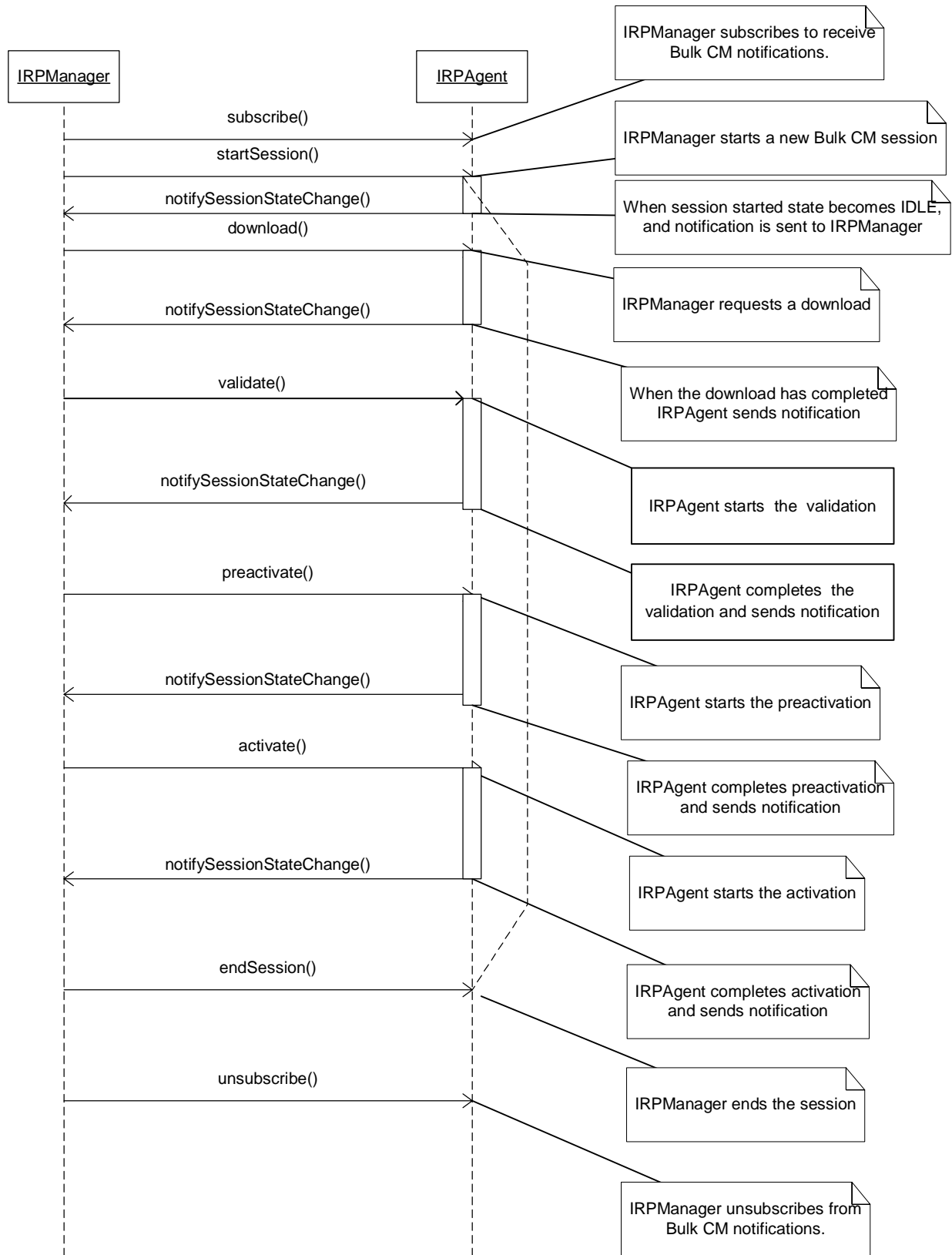


Figure A.3: Example 3: Successful Download and Activation with validation and preactivation.

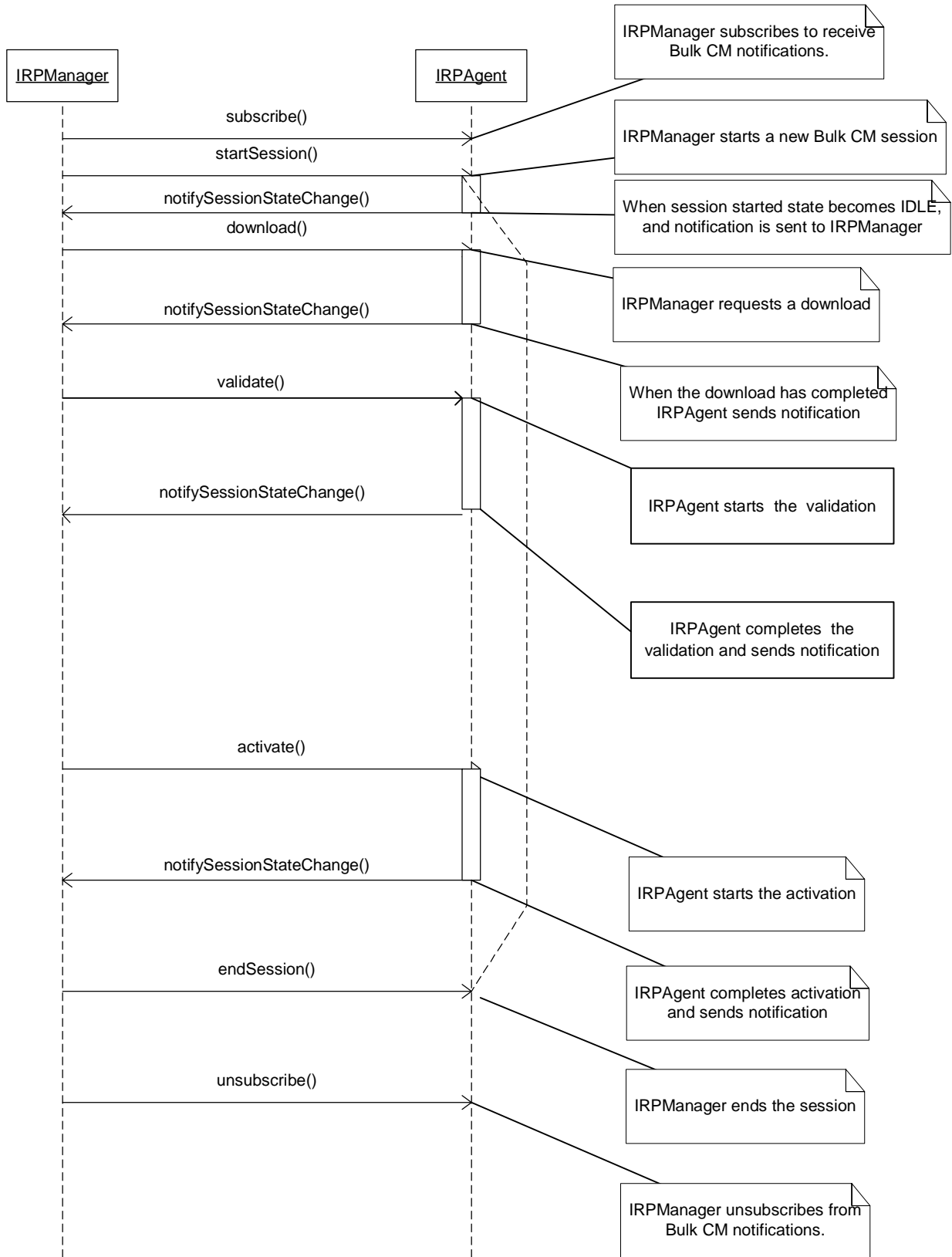


Figure A.4: Example 4: Successful Download and Activation with Validation



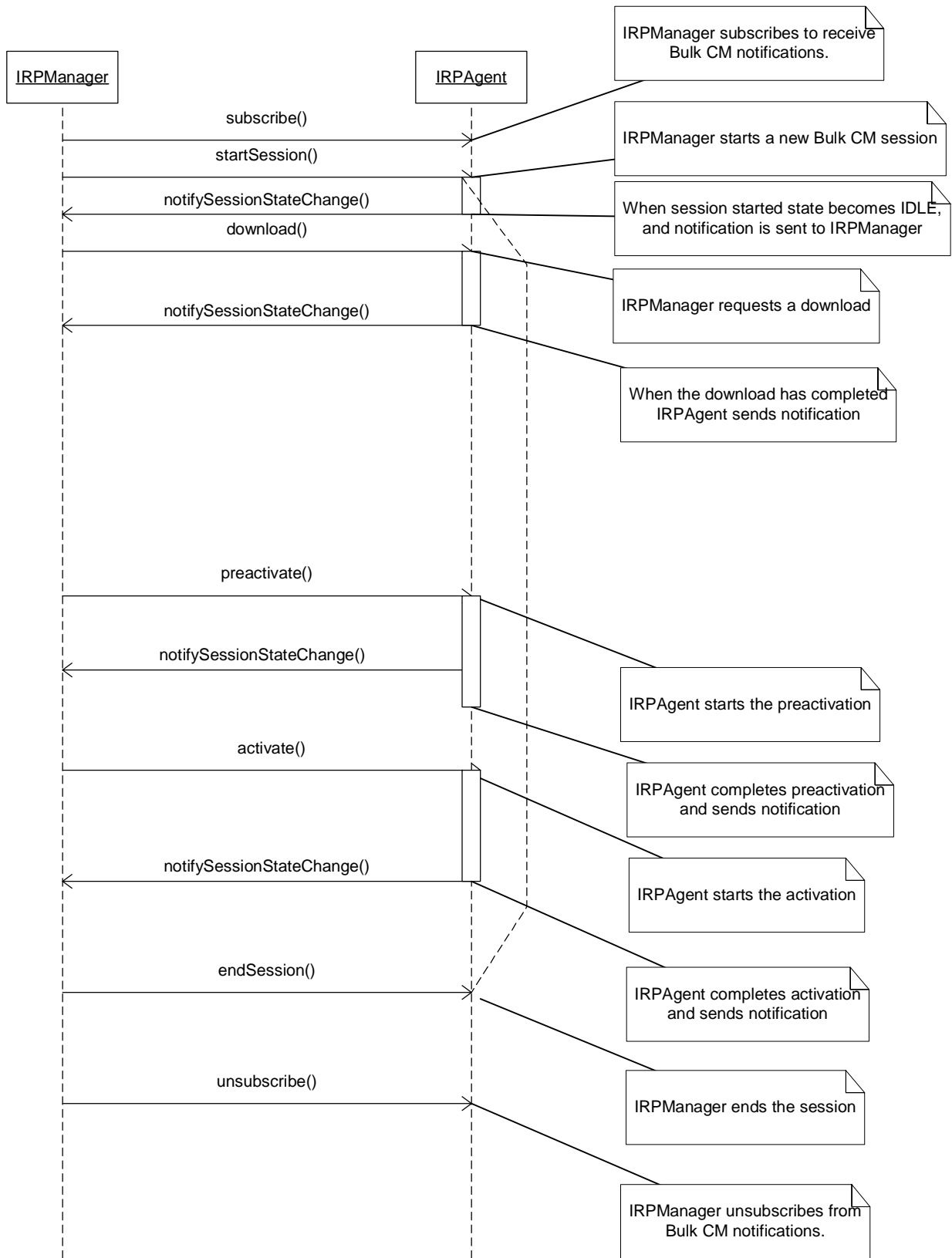


Figure A.5: Example 5: Successful Download and Activation with Preactivation.

## Annex B (informative): Bulk CM Application and Operation Principles

### B1 Key characteristics

1. Bulk CM operations are not transaction based.
2. The state machine does not allow looping. Can only progress forward through main states.
3. If any errors are found in the configuration data, it shall not be possible to fix the configuration data. A new session should be started with new corrected configuration data being downloaded.
4. Non-transitional interface;
5. Sessions may be run in parallel. There shall not be any exclusion of specified changes between parallel sessions. (Editors note: a "lock" option may be added, subject to further contribution).

## Annex C (informative): Change history

Change history							
Date	TSG #	TSG Doc.	CR	Rev	Subject/Comment	Old	New
Jun 2001	S_12	SP-010283	--	--	Approved at TSG SA #12 and placed under Change Control	2.0.0	4.0.0
Sep 2001	S_13	SP-010479	001	--	Correction of State Machine Pre and Post Conditions	4.0.0	4.1.0
Jun 2002	S_16	SP-020296	002	--	Correction of behaviour for IS parameter "saveFallback" of IS operation "activate"	4.1.0	4.2.0
Sep 2002	S_17	SP-020484	003	--	Correction of pre- and post-conditions for the operations getSessionStatus and getSessionLog	4.2.0	4.3.0
Sep 2002	S_17	SP-020486	003	--	Add Bulk CM IRP IS Enhancements for Rel-5	4.3.0	5.0.0
Dec 2002	S_18	SP-020744	006	--	Incomplete getSessionStatus	5.0.0	5.1.0
Mar 2003	--	--	--	--	Editorial (Clause heading missing: 8 Bulk Configuration Data File)	5.1.0	5.1.1

# 3GPP TS 32.642 V5.2.0 (2003-06)

---

*Technical Specification*

**3rd Generation Partnership Project;  
Technical Specification Group Services and System Aspects;  
Telecommunication management;  
Configuration Management (CM);  
UTRAN network resources Integration Reference Point (IRP):  
Network Resource Model (NRM)  
(Release 5)**

---



The present document has been developed within the 3<sup>rd</sup> Generation Partnership Project (3GPP<sup>TM</sup>) and may be further elaborated for the purposes of 3GPP.

The present document has not been subject to any approval process by the 3GPP Organizational Partners and shall not be implemented. This Specification is provided for future development work within 3GPP only. The Organizational Partners accept no liability for any use of this Specification. Specifications and reports for implementation of the 3GPP<sup>TM</sup> system should be obtained via the 3GPP Organizational Partners' Publications Offices.

---

Keywords

---

UMTS, management

**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.

© 2003, 3GPP Organizational Partners (ARIB, CWTS, ETSI, T1, TTA, TTC).  
All rights reserved.

# Contents

Foreword.....	5
Introduction.....	5
1 Scope.....	6
2 References.....	6
3 Definitions and abbreviations.....	7
3.1 Definitions.....	7
3.2 Abbreviations.....	8
4 System overview.....	8
4.1 System context.....	8
4.2 Compliance rules.....	9
5 Modelling approach.....	10
6 IRP Information Model.....	10
6.1 Information entities imported and local labels.....	10
6.2 Class diagram.....	10
6.2.1 Attributes and relationships.....	10
6.2.2 Inheritance.....	11
6.3 Information object classes definition.....	13
6.3.1 RncFunction.....	13
6.3.1.1 Definition.....	13
6.3.1.2 Attributes.....	13
6.3.2 NodeBFunction.....	13
6.3.2.1 Definition.....	13
6.3.2.2 Attributes.....	13
6.3.3 UtranCell.....	14
6.3.3.1 Definition.....	14
6.3.3.2 Attributes.....	14
6.3.4 IubLink.....	15
6.3.4.1 Definition.....	15
6.3.4.2 Attributes.....	15
6.3.5 UtranRelation.....	15
6.3.5.1 Definition.....	15
6.3.5.2 Attributes.....	16
6.3.5.3 Attribute constraints.....	16
6.3.6 ExternalUtranCell.....	16
6.3.6.1 Definition.....	16
6.3.6.2 Attributes.....	16
6.4 Information relationships definition.....	17
6.4.1 ConnectedTo (M).....	17
6.4.1.1 Definition.....	17
6.4.1.2 Roles	17
6.4.1.3 Constraints.....	17
6.4.2 AssociatedWith (M).....	17
6.4.2.1 Definition.....	17
6.4.2.2 Roles	17
6.4.2.3 Constraints.....	17
6.4.3 ExternalUtranNeighbourCellRelation (M).....	18
6.4.3.1 Definition.....	18
6.4.3.2 Roles	18
6.4.3.3 Constraints.....	18
6.4.4 UtranNeighbourCellRelation (M).....	18
6.4.4.1 Definition.....	18
6.4.4.2 Roles	18
6.4.4.3 Constraints.....	18

6.5 Information attributes definition..... 19  
6.5.1 Definition and legal values ..... 19  
6.5.2 Constraints..... 20  
6.6 Particular information configurations ..... 20

**Annex A (informative): Supported UTRAN network configurations .....21**

**Annex B (informative): Change history .....23**

---

## Foreword

This Technical Specification has been produced by the 3<sup>rd</sup> 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

Configuration Management (CM), in general, provides the operator with the ability to assure correct and effective operation of the 3G network as it evolves. CM actions have the objective to control and monitor the actual configuration on the Network Elements (NEs) and Network Resources (NRs), and they may be initiated by the operator or by functions in the Operations Systems (OSs) or NEs.

CM actions may be requested as part of an implementation programme (e.g. additions and deletions), as part of an optimisation programme (e.g. modifications), and to maintain the overall Quality of Service (QoS). The CM actions are initiated either as single actions on single NEs of the 3G network, or as part of a complex procedure involving actions on many resources/objects in one or several NEs.

Configuration Management (CM), in general, provides the operator with the ability to assure correct and effective operation of the 3G network as it evolves. CM actions have the objective to control and monitor the actual configuration on the Network Elements (NEs) and Network Resources (NRs), and they may be initiated by the operator or by functions in the Operations Systems (OSs) or NEs.



---

# 1 Scope

The present document is part of an Integration Reference Point (IRP) named "UTRAN Network Resources IRP", through which an "IRPAgent" (typically an Element Manager or Network Element) can communicate Configuration Management information to one or several "IRPManagers" (typically Network Managers) concerning UTRAN resources. The "UTRAN Network Resources IRP" comprises a set of specifications defining Requirements, a protocol neutral Network Resource Model (NRM) and corresponding Solution Set(s).

The present document

1. specifies the protocol neutral UTRAN Network Resources IRP: Network Resource Model. It reuses relevant parts of the generic NRM in TS 32.622 [16], either by direct reuse or sub-classing, and in addition to that defines UTRAN specific Managed Object Classes.

The Configuration Management (CM) area is very large. The intention is to split the specification of the related interfaces in several IRPs – as described in the Introduction clause above. An important aspect of such a split is that the Network Resource Models (NRMs) defined in different IRPs containing NRMs are consistent, and that NRMs supported by an IRPAgent implementation can be accessed as one coherent model through one IRP Information Service.

To summarize, the present document has the following main purpose:

- (1) to define the applied UTRAN specific Network Resource Model, based on the generic NRM in TS 32.622 [16].

In order to access the information defined by this NRM, an IRP Information Service (IS) is needed, such as the Basic CM IRP: IS (TS 32.602 [17]) or the Bulk CM IRP: IS (TS 32.612 [18]). However, which Information Service that is applicable is outside the scope of this document.

Regarding the support of the State Management IRP: IS (TS 32.672 [8]), all NRMs of one release shall support the same State Management IRP version. This NRM specification is related to 3GPP TS 32.672 V5.0.X.

---

# 2 References

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

- References are either specific (identified by date of publication, edition number, version number, etc.) 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 TS 32.101: "Telecommunication management; Principles and high level requirements".
- [2] 3GPP TS 32.102: "Telecommunication management; Architecture".
- [3] 3GPP TS 23.003: "Numbering, addressing and identification".
- [4] 3GPP TS 25.401: "UTRAN Overall Description"
- [5] 3GPP TS 25.433: "UTRAN Iub Interface NBAP Signalling"
- [6] 3GPP TS 25.423: "UTRAN Iur Interface RNSAP Signalling"
- [7] ITU-T Recommendation X.710 (1991): "Common Management Information Service Definition for CCITT Applications".
- [8] 3GPP TS 32.672: "Telecommunication management; Configuration Management (CM); State Management Integration Reference Point (IRP): Information service".

- [9] Void
- [10] Void
- [11] 3GPP TS 32.111-2: "Telecommunication management; Fault Management; Part 2: Alarm Integration Reference Point (IRP): Information Service".
- [12] Void
- [13] 3GPP TS 32.300: "Telecommunication management; Configuration Management (CM); Name convention for Managed Objects".
- [14] 3GPP TS 32.600: "Telecommunication management; Configuration Management (CM); Concept and high-level requirements".
- [15] 3GPP TS 23.002: "Network Architecture".
- [16] 3GPP TS 32.622: "Telecommunication management; Configuration Management (CM); Generic network resources Integration Reference Point (IRP): Network Resource Model (NRM)".
- [17] 3GPP TS 32.602: "Telecommunication management; Configuration Management (CM); Basic CM Integration Reference Point (IRP) information service".
- [18] 3GPP TS 32.612: "Telecommunication management; Configuration Management (CM); Bulk CM Integration Reference Point (IRP): Information service".

---

## 3 Definitions and abbreviations

### 3.1 Definitions

For the purposes of the present document, the following terms and definitions apply. For terms and definitions not found here, please refer to 3GPP TS 32.101 [1], 3GPP TS 32.102 [2] and 3GPP TS 32.600 [14].

**Association:** In general it is used to model relationships between Managed Objects. Associations can be implemented in several ways, such as:

- (1) name bindings,
- (2) reference attributes, and
- (3) association objects.

This IRP stipulates that containment associations shall be expressed through name bindings, but it does not stipulate the implementation for other types of associations as a general rule. These are specified as separate entities in the object models (UML diagrams).

**Managed Element (ME):** An instance of the Managed Object Class ManagedElement defined in [16].

**Managed Object (MO):** In the context of the present document, a Managed Object (MO) is a software object that encapsulates the manageable characteristics and behaviour of a particular Network Resource. The MO is instance of a MO class defined in a MIM/NRM. This class, called **Information Object Class (IOC)** has attributes that provide information used to characterize the objects that belong to the class (the term "attribute" is taken from TMN and corresponds to a "property" according to CIM). Furthermore, the IOC can have operations that represent the behaviour relevant for that class (the term "operation" is taken from TMN and corresponds to a "method" according to CIM). The IOC may support the emission of notifications that provide information about an event occurrence within a network resource.

**Management Information Model (MIM):** Also referred to as NRM – see the definition below.

**Network Resource Model (NRM):** A model representing the actual managed telecommunications network resources that a System is providing through the subject IRP. An NRM identifies and describes the IOCs, their associations, attributes and operations. The NRM is also referred to as "MIM" (see above), which originates from the ITU-T TMN.

**Node B:** A logical node responsible for radio transmission/reception in one or more cells to/from the User Equipment. It terminates the Iub interface towards the RNC.

## 3.2 Abbreviations

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

CIM	Common Information Model
DN	Distinguished Name (see 3GPP TS 32.300 [13])
EM	Element Manager
FM	Fault Management
IOC	Information Object Class
IRP	Integration Reference Point
Iub	Interface between RNC and Node B
ME	Managed Element
MIM	Management Information Model
MO	Managed Object
NE	Network Element
NM	Network Manager
NR	Network Resource
NRM	Network Resource Model
PM	Performance Management
RDN	Relative Distinguished Name (see 3GPP TS 32.300 [13])
RNC	Radio Network Controller
TMN	Telecommunications Management Network
UML	Unified Modelling Language
UMTS	Universal Mobile Telecommunications System
UTRAN	Universal Terrestrial Radio Access Network

---

## 4 System overview

### 4.1 System context

Figure 4.1 and 4.2 identify system contexts of the IRP defined by the present document in terms of its implementation called IRPAgent and the user of the IRPAgent, called IRPManager. For a definition of IRPManager and IRPAgent, see 3GPP TS 32.102 [2].

The IRPAgent implements and supports this IRP. The IRPAgent can reside in an Element Manager (EM; for definition see 3GPP TS 32.101 [1]) or a Network Element (NE) (see also [2] clause 8). In the former case, the interface (represented by a thick dotted line) between the EM and the NEs is not the subject of this IRP.

An IRPManager using this IRP shall choose one of the two System Contexts defined here, for each NE. For instance, if an EM is responsible for managing a number of NEs, the NM shall access this IRP through the EM and not directly to those NEs. For another IRP though, the System Context may be different.

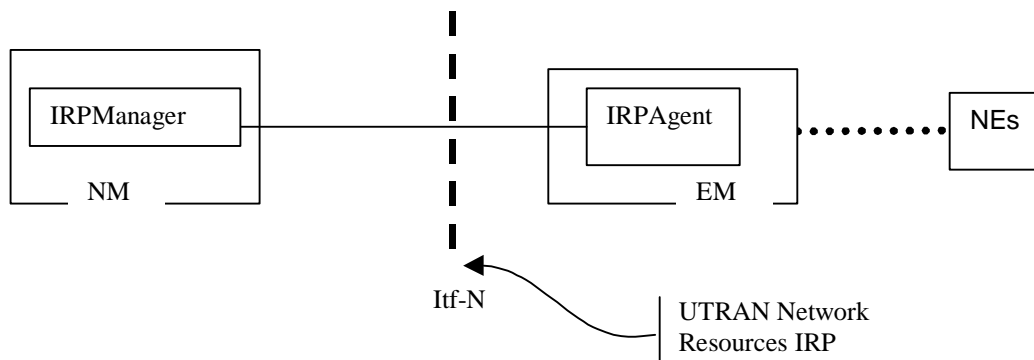


Figure 4.1: System Context A

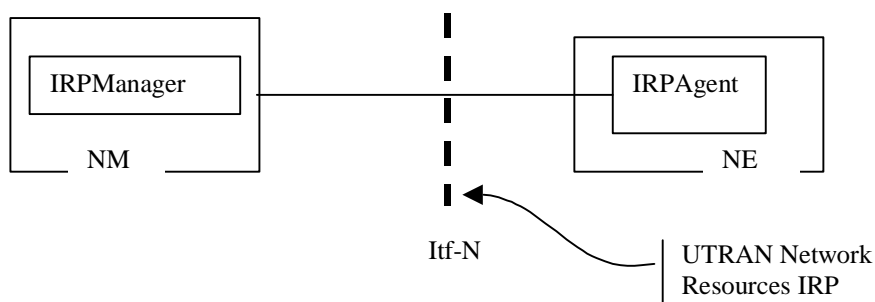


Figure 4.2: System Context B

## 4.2 Compliance rules

For general definitions of compliance rules related to qualifiers (Mandatory/Optional/Conditional) for *operations, notifications and parameters* (of operations and notifications) please refer to 3GPP TS 32.102 [2].

The following defines the meaning of Mandatory and Optional IOC attributes and associations between IOCs, in Solution Sets to the IRP defined by the present document:

- The IRPManager shall support all mandatory attributes/associations. The IRPManager shall be prepared to receive information related to mandatory as well as optional attributes/associations without failure; however the IRPManager does not have to support handling of the optional attributes/associations.
- The IRPAgent shall support all mandatory attributes/associations. It may support optional attributes/associations.

An IRPAgent that incorporates vendor-specific extensions shall support normal communication with a 3GPP SA5-compliant IRPManager with respect to all Mandatory and Optional managed object classes, attributes, associations, operations, parameters and notifications without requiring the IRPManager to have any knowledge of the extensions.

Given that

- rules for vendor-specific extensions remain to be fully specified, and

- many scenarios under which IRPManager and IRPAgent interwork may exist,

it is recognised that in Release 4/5 the IRPManager, even though it is not required to have knowledge of vendor-specific extensions, may be required to be implemented with an awareness that extensions can exist and behave accordingly.

---

## 5 Modelling approach

The modelling approach adopted and used in this IRP is described in the Generic Network Resources IRP: NRM [16].

---

## 6 IRP Information Model

### 6.1 Information entities imported and local labels

None.

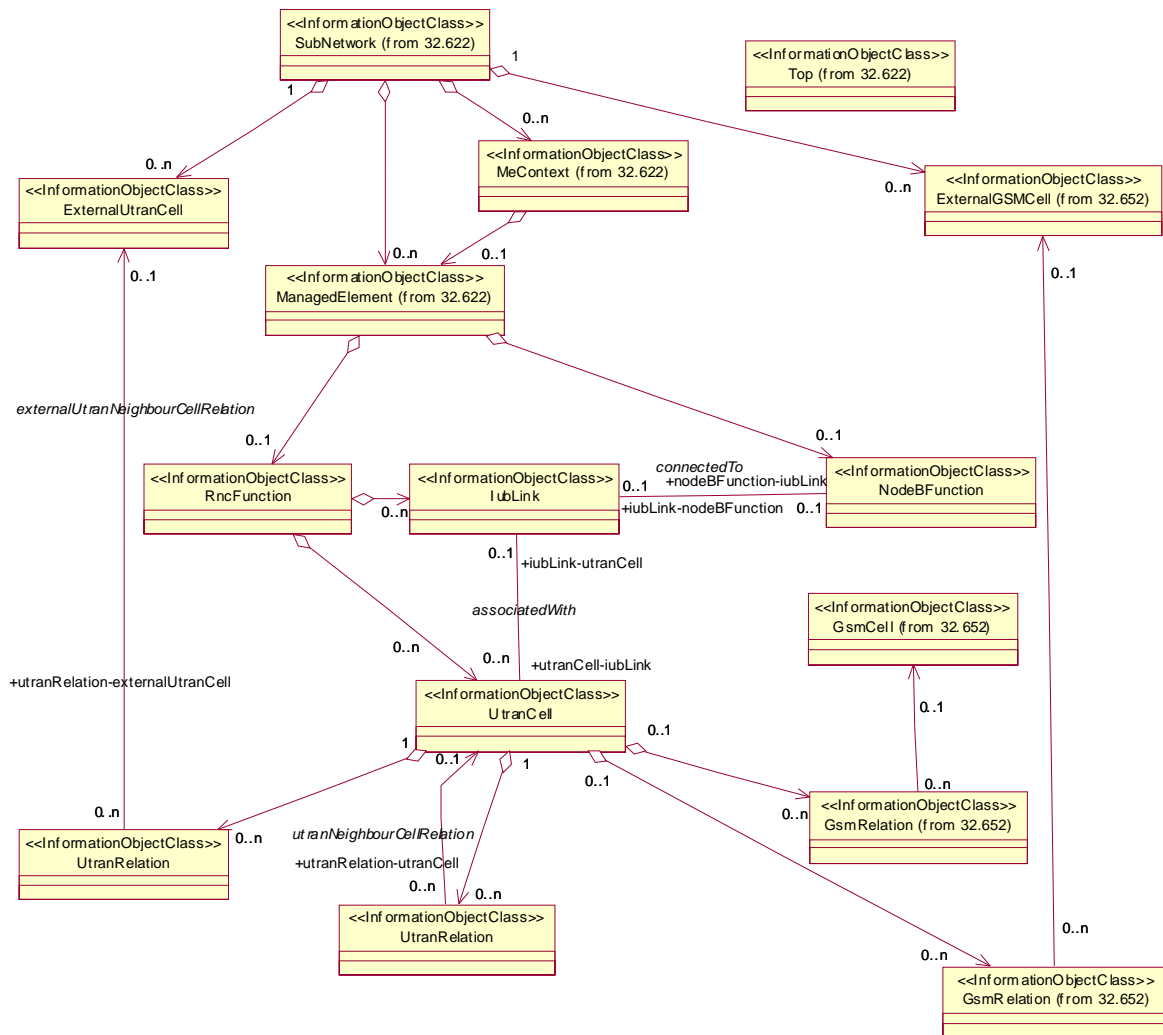
### 6.2 Class diagram

#### 6.2.1 Attributes and relationships

This subclause depicts the set of IOCs that encapsulate information relevant for this service. This subclause provides the overview of all information object classes in UML. Subsequent subclauses provide more detailed specification of various aspects of these information object classes.

Figure 6.1 show the name-containment relation and other types of relations of the UTRAN NRM.

NOTE: The name-containment relations between IOCs are indicated by UML "unidirectional aggregation by reference" ("hollow diamonds").



- NOTE 1: The listed cardinality numbers represent transient as well as steady-state numbers, and reflect all managed object creation and deletion scenarios.
- NOTE 2: The relation between GsmRelation and GsmCell is optional. It may be present if both the UtranCell and the GsmCell are managed by the same management node.
- NOTE 3: The UtranRelation and GsmRelation can be name-contained under IOCs defined in other NRMs.

**Figure 6.1: UTRAN NRM Containment/Naming and Association diagram**

Each IOC is identified with a Distinguished Name (DN) according to 3GPP TS 32.300 [13] that expresses its containment hierarchy. As an example, the DN of an IOC representing a cell could have a format like:

SubNetwork=Sweden,MeContext=MEC-Gbg-1,ManagedElement=RNC-Gbg-1, RncFunction=RF-1,UtranCell=Gbg-1.

### 6.2.2 Inheritance

This sub-clause depicts the inheritance relationships that exist between IOCs.

Figure 6.2 shows the inheritance hierarchy for the UTRAN NRM.

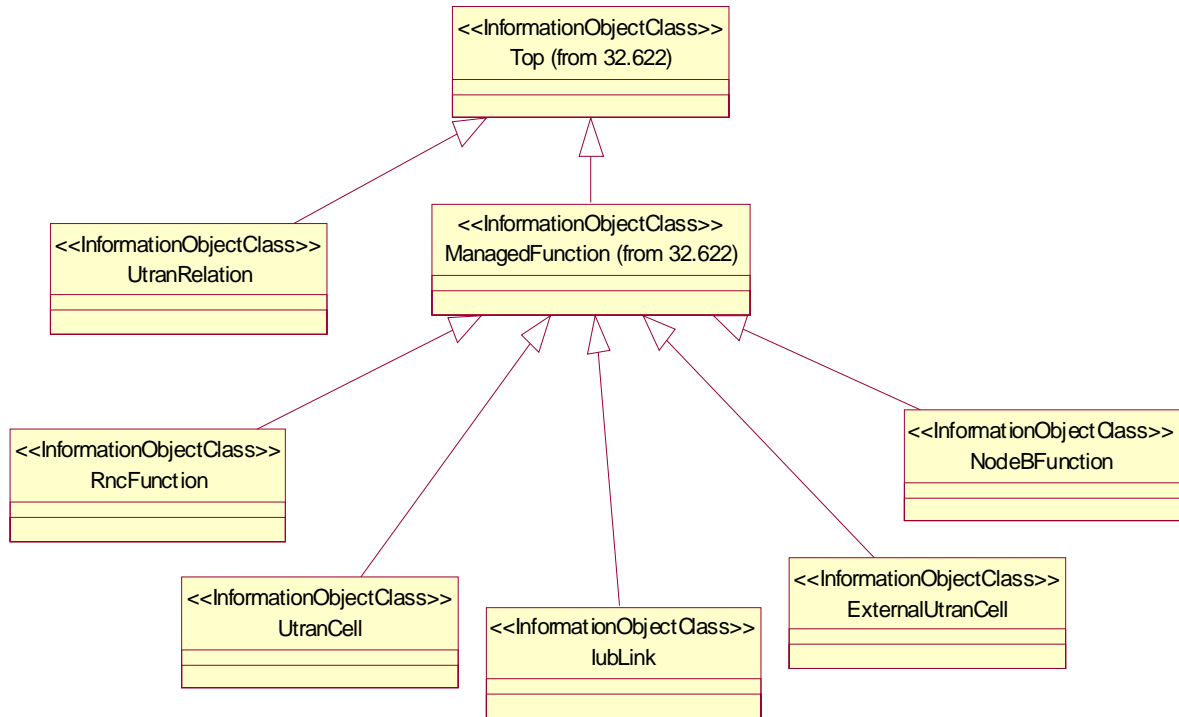
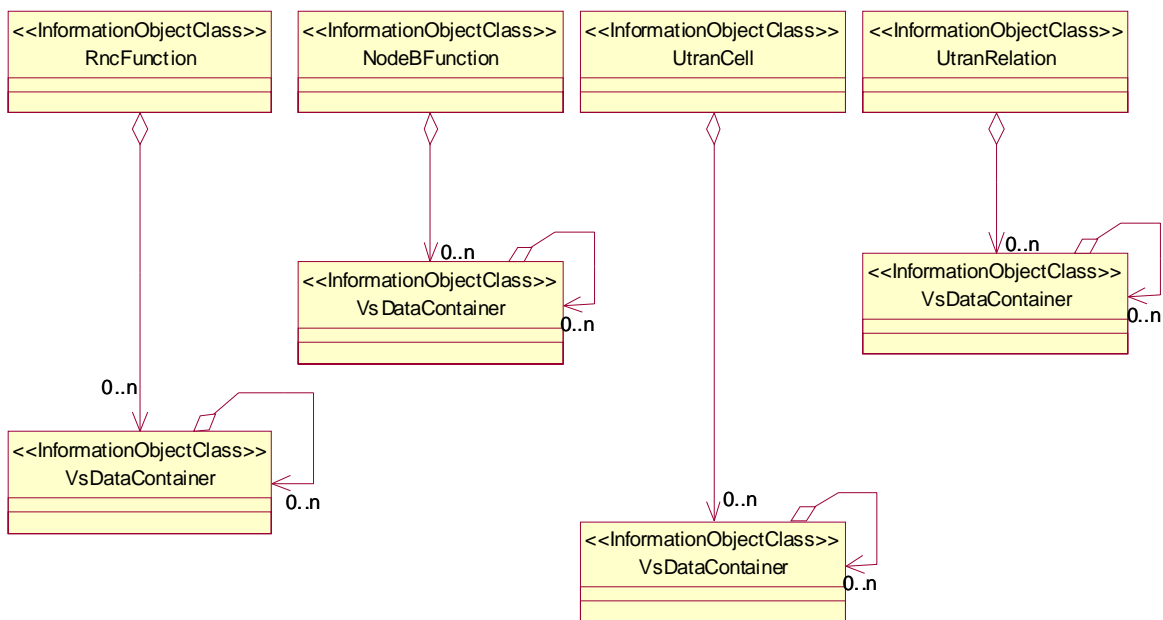


Figure 6.2: UTRAN NRM Inheritance Hierarchy



NOTE 1: The listed cardinality numbers represent transient as well as steady-state numbers, and reflect all managed object creation and deletion scenarios.

NOTE 2: Each instance of the vsDataContainer shall only be contained under one IOC. The vsDataContainer can be contained under IOCs defined in other NRMs.

Figure 6.3: vsDataContainer Containment/Naming and Association in UTRAN NRM diagram

The vsDataContainer is only used for the Bulk CM IRP.

## 6.3 Information object classes definition

### 6.3.1 RncFunction

#### 6.3.1.1 Definition

This IOC represents RNC functionality. For more information about the RNC, see 3GPP TS 23.002 [15].

#### 6.3.1.2 Attributes

**Table 6.1: Attributes of RncFunction**

Attribute name	Visibility	Support Qualifier	Read Qualifier	Write Qualifier
rncFunctionId	+	M	M	-
userLabel	+	M	M	M
mcc	+	M	M	M
mnc	+	M	M	M
rnclid	+	M	M	M

**Table 6.2: Notifications of RncFunction**

Name	Qualifier	Notes
notifyAckStateChanged	See Alarm IRP (3GPP TS 32.111-2 [11])	
notifyAttributeValueChange	O	
notifyChangedAlarm	See Alarm IRP (3GPP TS 32.111-2 [11])	
notifyClearedAlarm	See Alarm IRP (3GPP TS 32.111-2 [11])	
notifyNewAlarm	See Alarm IRP (3GPP TS 32.111-2 [11])	
notifyObjectCreation	O	
notifyObjectDeletion	O	
notifyComments	See Alarm IRP (3GPP TS 32.111-2 [11])	
notifyAlarmListRebuilt	See Alarm IRP (3GPP TS 32.111-2 [11])	

### 6.3.2 NodeBFunction

#### 6.3.2.1 Definition

This IOC represents Node B functionality. For more information about the Node B, see 3GPP TS 23.002 [15].

#### 6.3.2.2 Attributes

**Table 6.3: Attributes of NodeBFunction**

Attribute name	Visibility	Support Qualifier	Read Qualifier	Write Qualifier
nodeBFunctionId	+	M	M	-
userLabel	+	M	M	M
nodeBFunction-lubLink	+	M	M	-



**Table 6.4 Notifications of NodeBFunction**

Name	Qualifier	Notes
notifyAckStateChanged	See Alarm IRP (3GPP TS 32.111-2 [11])	
notifyAttributeValueChange	O	
notifyChangedAlarm	See Alarm IRP (3GPP TS 32.111-2 [11])	
notifyClearedAlarm	See Alarm IRP (3GPP TS 32.111-2 [11])	
notifyNewAlarm	See Alarm IRP (3GPP TS 32.111-2 [11])	
notifyObjectCreation	O	
notifyObjectDeletion	O	
notifyComments	See Alarm IRP (3GPP TS 32.111-2 [11])	
notifyAlarmListRebuilt	See Alarm IRP (3GPP TS 32.111-2 [11])	

### 6.3.3 UtranCell

#### 6.3.3.1 Definition

This IOC represents a radio cell controlled by the RNC. For more information about radio cells, see 3GPP TS 23.002 [15].

#### 6.3.3.2 Attributes

**Table 6.5: Attributes of UtranCell**

Attribute name	Visibility	Support Qualifier	Read Qualifier	Write Qualifier
utranCellId	+	M	M	-
userLabel	+	M	M	M
cld	+	M	M	M
localCellId	+	M	M	M
uarfcnUI	+	M	M	M
uarfcnDI	+	M	M	M
primaryScramblingCode	+	M	M	M
primaryCpichPower	+	M	M	M
maximumTransmissionPower	+	M	M	M
primarySchPower	+	M	M	M
secondarySchPower	+	M	M	M
bchPower	+	M	M	M
lac	+	M	M	M
rac	+	M	M	M
sac	+	M	M	M
ura	+	M	M	M
utranCell-lubLink	+	M	M	-

**Table 6.6: Additional attributes of UtranCell for the support of the State Management IRP**

Attribute Name	Support Qualifier	READ	WRITE
operationalState	O	M	-
NOTE: No state propagation shall be implied.			

**Table 6.7: Notifications of UtranCell**

Name	Qualifier	Notes
notifyAckStateChanged	See Alarm IRP (3GPP TS 32.111-2 [11])	
notifyAttributeValueChange	O	
notifyChangedAlarm	See Alarm IRP (3GPP TS 32.111-2 [11])	
notifyClearedAlarm	See Alarm IRP (3GPP TS 32.111-2 [11])	
notifyNewAlarm	See Alarm IRP (3GPP TS 32.111-2 [11])	
notifyObjectCreation	O	
notifyObjectDeletion	O	
notifyComments	See Alarm IRP (3GPP TS 32.111-2 [11])	
notifyAlarmListRebuilt	See Alarm IRP (3GPP TS 32.111-2 [11])	

## 6.3.4 IubLink

### 6.3.4.1 Definition

This IOC represents the logical link to a Node B as seen from the RNC. For more information about the RNC, see 3GPP TS 23.002 [15].

### 6.3.4.2 Attributes

**Table 6.8: Attributes of IubLink**

Attribute name	Visibility	Support Qualifier	Read Qualifier	Write Qualifier
IubLinkId	+	M	M	-
userLabel	+	M	M	M
IubLink-UtranCell	+	M	M	M
IubLink-NodeBFunction	+	M	M	-

**Table 6.9: Notifications of IubLink**

Name	Qualifier	Notes
notifyAckStateChanged	See Alarm IRP (3GPP TS 32.111-2 [11])	
notifyAttributeValueChange	O	
notifyChangedAlarm	See Alarm IRP (3GPP TS 32.111-2 [11])	
notifyClearedAlarm	See Alarm IRP (3GPP TS 32.111-2 [11])	
notifyNewAlarm	See Alarm IRP (3GPP TS 32.111-2 [11])	
notifyObjectCreation	O	
notifyObjectDeletion	O	
notifyComments	See Alarm IRP (3GPP TS 32.111-2 [11])	
notifyAlarmListRebuilt	See Alarm IRP (3GPP TS 32.111-2 [11])	

## 6.3.5 UtranRelation

### 6.3.5.1 Definition

The "UtranRelation" IOC contains radio network related parameters for the relation to the "UtranCell" or "ExternalUtranCell" IOC.

**NOTE:** In handover relation terms, the cell containing the UTRAN Relation object is the source cell for the handover. The cell referred to in the UTRAN relation object is the target cell for the handover. This defines a one-way handover relation where the direction is *from* source cell *to* target cell.

## 6.3.5.2 Attributes

Table 6.10: Attributes of UtranRelation

Attribute name	Visibility	Support Qualifier	Read Qualifier	Write Qualifier
utranRelationId	+	M	M	-
adjacentCell	+	M	M	M
uarfcnUI	+	O	M	-
uarfcnDI	+	O	M	-
primaryScramblingCode	+	O	M	-
primaryCpichPower	+	O	M	-
lac	+	O	M	-

Table 6.11: Notifications of UtranRelation

Name	Qualifier	Notes
notifyAttributeValueChange	O	
notifyObjectCreation	O	
notifyObjectDeletion	O	

## 6.3.5.3 Attribute constraints

The optionally attributes uarfcnUI, uarfcnDI, primaryScramblingCode, primaryCpichPower and lac shall be included if the EM does not guarantee consistency between the cell definition and what is broadcast on system information. Otherwise they shall not be included.

## 6.3.6 ExternalUtranCell

## 6.3.6.1 Definition

This IOC represents a radio cell controlled by another IRPAgent. This IOC has necessary attributes for inter-system handover. It contains a subset of the attributes of related IOCs controlled by another IRPAgent. The way to maintain consistency between the attribute values of these two IOCs is outside the scope of this document.

## 6.3.6.2 Attributes

Table 6.12: Attributes of ExternalUtranCell

Attribute name	Visibility	Support Qualifier	Read Qualifier	Write Qualifier
externalUtranCellId	+	M	M	-
userLabel	+	M	M	M
cId	+	M	M	M
mcc	+	M	M	M
mnc	+	M	M	M
rnclId	+	M	M	M
uarfcnUI	+	M	M	M
uarfcnDI	+	M	M	M
primaryScramblingCode	+	M	M	M
primaryCpichPower	+	M	M	M
lac	+	M	M	M
rac	+	M	M	M

Table 6.13: Notifications of ExternalUtranCell

Name	Qualifier	Notes
notifyAttributeValueChange	O	
notifyObjectCreation	O	
notifyObjectDeletion	O	

## 6.4 Information relationships definition

### 6.4.1 ConnectedTo (M)

#### 6.4.1.1 Definition

This represents a bi-directional relationship between the IubLink and Node B (through the NodeBFunction). The role of the relation shall be mapped to a reference attribute of the IOC. The names of the reference attribute and the role are the same.

#### 6.4.1.2 Roles

**Table 6.14: Roles of the relation ConnectedTo**

Name	Definition
iubLink-nodeBFunction	This role (when present) represents IubLink capability to identify one NodeBFunction. When the role is absent, the IubLink.iubLink-nodeBFunction shall contain no information. When present, it shall contain one NodeBFunction DN.
nodeBFunction-iubLink	This role (when present) represents NodeBFunction capability to identify one IubLink. When the role is absent, the NodeBFunction.nodeBFunction-iubLink shall contain no information. When present, it shall contain one IubLink DN.

#### 6.4.1.3 Constraints

When a particular IubLink identifies a particular NodeBFunction, that particular NodeBFunction must identify the particular IubLink.

### 6.4.2 AssociatedWith (M)

#### 6.4.2.1 Definition

This represents a bi-directional relation between the IubLink and UtranCell. The role of the relation shall be mapped to a reference attribute of the IOC. The name of the reference attribute shall be the role name.

#### 6.4.2.2 Roles

**Table 6.15: Roles of the relation AssociatedWith**

Name	Definition
iubLink-utranCell	This role (when present) represents IubLink capability to identify the set of related UtranCell. IubLink.iubLink-utranCell shall carry the set of UtranCell DN(s).
utranCell-iubLink	This role (when present) represents UtranCell capability to identify one related IubLink. When the role is absent, the UtranCell.utranCell-iubLink shall contain no information. When it is present, it shall contain one IubLink DN.

#### 6.4.2.3 Constraints

When a particular IubLink identifies a particular UtranCell, that particular UtranCell must have identified the particular IubLink.

## 6.4.3 ExternalUtranNeighbourCellRelation (M)

### 6.4.3.1 Definition

This represents a unidirectional relation from UtranRelation to the ExternalUtranCell. The role of the relation shall be mapped to a reference attribute, named adjacentCell, of the IOC.

### 6.4.3.2 Roles

**Table 6.16: Roles of the relation ExternalUtranNeighbourCellRelation**

Name	Definition
utranRelation-externalUtranNeighbourCell	This role (when present) represents UtranRelation capability to identify one ExternalUtranCell. When this role is present, the UtranRelation.adjacentCell shall contain one ExternalUtranNeighbourCell DN.

### 6.4.3.3 Constraints

This role (for a particular UtranRelation) shall be present if the UtranNeighbourCellRelation of this particular UtranRelation is absent. This role shall be absent if the UtranNeighbourCellRelation of this particular UtranRelation is present.

## 6.4.4 UtranNeighbourCellRelation (M)

### 6.4.4.1 Definition

This represents the unidirectional relation from the UtranRelation to UtranCell. The role of the relation shall be mapped to a reference attribute, named adjacentCell, of the IOC.

### 6.4.4.2 Roles

**Table 6.17: Roles of the relation UtranNeighbourCellRelation**

Name	Definition
utranRelation-utranNeighbourCell	This role (when present) represents UtranRelation capability to identify one UtranCell. When this role is present, the UtranRelation.adjacentCell shall contain one UtranCell DN.

### 6.4.4.3 Constraints

This role (for a particular UtranRelation) shall be present if the ExternalUtranNeighbourCellRelation of this particular UtranRelation is absent. This role shall be absent if the ExternalUtranNeighbourCellRelation of this particular UtranRelation is present.

## 6.5 Information attributes definition

### 6.5.1 Definition and legal values

The table below defines the attributes that are present in several Information Object Classes (IOCs) of this TS.

**Table 6.18: Attributes**

Attribute Name	Definition	Legal Values
adjacentCell	It carries the DN of the UtranCell or the ExternalUtranCell.	
bchPower	The power of the broadcast channel in the cell (Ref. 3GPP TS 25.433 [5]).	Type: Numeric value Range: (-35..+15 dB) Steps of 0.1dB
cld	The attribute is the identifier of a cell in one RNC (Ref. 3GPP TS 25.401 [4]), 3GPP TS 25.433 [5]).	Type: Integral numeric value Range: (0...65535)
externalUtranCellId	An attribute whose "name+value" can be used as an RDN when naming an instance of the object class. This RDN uniquely identifies the object instance within the scope of its containing (parent) object instance.	
iubLinkId	An attribute whose "name+value" can be used as an RDN when naming an instance of the object class. This RDN uniquely identifies the object instance within the scope of its containing (parent) object instance.	
lac	IOCs <b>UtranCell</b> and <b>ExternalUtranCell</b> : Location Area Code, LAC (Ref. 3GPP TS 23.003 [3]). IOC <b>UtranRelation</b> : Location Area Code, LAC (Ref. 3GPP TS 23.003 [3]), for another UTRAN cell or the external UTRAN Cell that is broadcast in the system information in the Cell.	Type: Integral numeric value Range: (1.. 65533, 65535)
localCellId	Local Cell id is used to uniquely identify the set of resources defined in a Node B to support a cell (as defined by a Cid Ref. 3GPP TS 25.401 [4]), 3GPP TS 25.433 [5]). It must be unique in Node B at a minimum, but may be unique in UTRAN. It can be used to tie the cell in the RNC to a specific set of resources in the Node B.	Type: Integral numeric value Range: (0...268435455)
maximumTransmissionPower	The maximum transmission power of a cell, DL Power (Ref. 3GPP TS 25.433 [5]).	Type: Numeric value Range: (0,..50 dBm) Steps of 0.1 dB
mcc	Mobile Country Code, MCC (part of the PLMN Id, Ref. 3GPP TS 23.003 [3]).	
mnc	Mobile Network Code, MNC (part of the PLMN Id, Ref. 3GPP TS 23.003 [3]).	
nodeBFunctionId	An attribute whose "name+value" can be used as an RDN when naming an instance of the object class. This RDN uniquely identifies the object instance within the scope of its containing (parent) object instance.	
primaryCpichPower	IOCs <b>UtranCell</b> and <b>ExternalUtranCell</b> : The power of the primary CPICH channel in the cell (Ref. 3GPP TS 25.433 [5]). IOC <b>UtranRelation</b> : The power of the primary CPICH channel in the cell (Ref. 3GPP TS 25.433 [5]), for another UTRAN cell or the external UTRAN Cell that is broadcast in the system information in the Cell.	Type: Numeric value Range: (-10,..,50 dBm) Steps of 0.1 dB
primarySchPower	The power of the primary synchronisation channel in the cell, DL Power (Ref. 3GPP TS 25.433 [5]).	Type: Numeric value Range: (-35..+15 dB) Steps of 0.1dB
primaryScramblingCode	IOCs <b>UtranCell</b> and <b>ExternalUtranCell</b> : The primary DL scrambling code used by the cell (Ref. 3GPP TS 25.433 [5]). IOC <b>UtranRelation</b> : The primary DL scrambling code used by the cell (Ref. 3GPP TS 25.433 [5]), for another UTRAN cell or the external UTRAN Cell that is broadcast in the system information in the Cell.	Type: Integral numeric value Range: (0 – 511)

Attribute Name	Definition	Legal Values
rac	Routing Area Code, RAC (Ref. 3GPP TS 23.003 [3]).	Type: Integral numeric value Range: (0..255)
rncFunctionId	An attribute whose "name+value" can be used as an RDN when naming an instance of the object class. This RDN uniquely identifies the object instance within the scope of its containing (parent) object instance.	
rncId	<b>IOC ExternalUtranCell:</b> Unique RNC ID for the associated RNC (Ref. 3GPP TS 23.003 [3]). <b>IOC RncFunction:</b> Unique RNC ID (Ref. 3GPP TS 23.003 [3]).	
sac	Service Area Code, SAC (Ref. 3GPP TS 23.003 [3]).	Type: Integral numeric value Range: (0.. 65535)
secondarySchPower	The power of the secondary synchronisation channel in the cell, DL Power (Ref. 3GPP TS 25.433 [5]).	Type: Numeric value Range: (-35..+15 dB) Steps of 0.1dB
uarfcnDl	<b>IOCs UtranCell and ExternalUtranCell:</b> The DL UTRA absolute Radio Frequency Channel number, UARFCN (Ref. 3GPP TS 25.433 [5]). <b>IOC UtranRelation:</b> The DL UTRA absolute Radio Frequency Channel number, UARFCN (Ref. 3GPP TS 25.433 [5]), for another UTRAN cell or the external UTRAN Cell that is broadcast in the system information in the Cell.	The channel number should correspond to a frequency in the downlink band, range 2110 MHz – 2170 MHz, or 1930 MHz – 1990 MHz for ITU Region 2. (Ref. 3GPP TS 25.101).  Type: Integral numeric value Range: (10562 - 10838) or (9662 - 9938)
uarfcnUl	<b>IOCs UtranCell and ExternalUtranCell:</b> The UL UTRA absolute Radio Frequency Channel number, UARFCN (Ref. 3GPP TS 25.433 [5]). <b>IOC UtranRelation:</b> The UL UTRA absolute Radio Frequency Channel number, UARFCN (Ref. 3GPP TS 25.433 [5]) for another UTRAN cell or the external UTRAN Cell, that is broadcast in the system information in the Cell.	The channel number should correspond to a frequency in the uplink band, range 1920 MHz – 1980 MHz, or 1850 MHz - 1910 MHz for ITU Region 2. (Ref. 3GPP TS 25.101)  Type: Integral numeric value Range: (9612 - 9888) or (9262 – 9538)
ura	UTRAN Registration Area, URA (Ref. 3GPP TS 25.423 [6]).	Type: Integral numeric value Range: (0..65535)
userLabel	A user-friendly (and user assigned) name of the associated object. Inherited from ManagedFunction.	
utranCellId	An attribute whose "name+value" can be used as an RDN when naming an instance of the object class. This RDN uniquely identifies the object instance within the scope of its containing (parent) object instance.	
utranRelationId	An attribute whose "name+value" can be used as an RDN when naming an instance of the object class. This RDN uniquely identifies the object instance within the scope of its containing (parent) object instance.	

## 6.5.2 Constraints

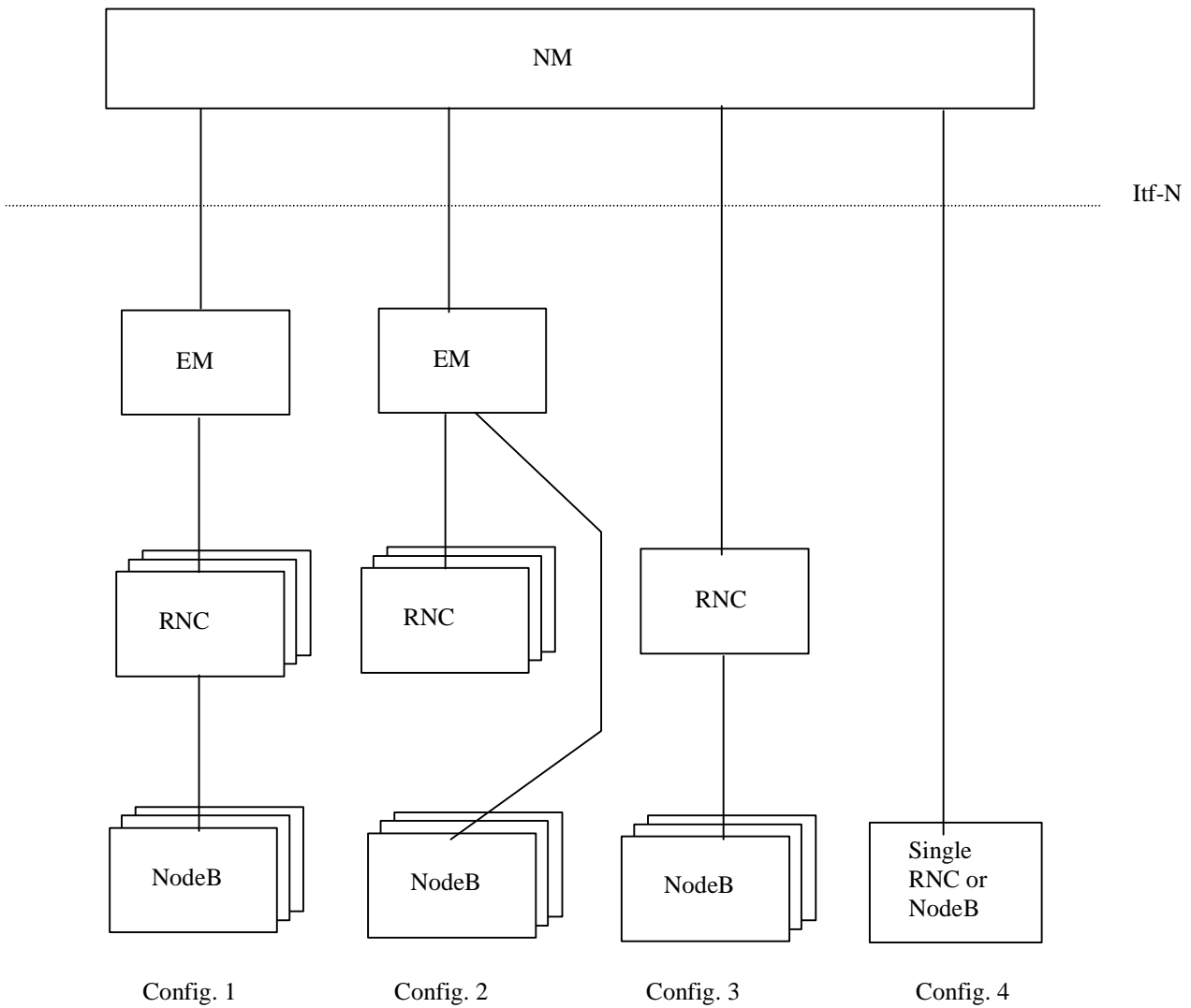
None.

## 6.6 Particular information configurations

Not applicable.

## Annex A (informative): Supported UTRAN network configurations

Figure A.1 depicts four typical network configurations, which are supported by the UTRAN NRM over the Itf-N. However, this does not preclude support for other configurations.



**Figure A.1: Typical network configurations supported by the UTRAN NRM**

Table A.1 shows the possible number of instances for each network configuration (counted from left to right in figure A.1.):



Table A.1: Number of instances for each example configuration in figure A.1

IOC	Config. 1	Config. 2	Config. 3	Config. 4
SubNetwork	1	1	1	0..1
ManagementNode	1	1	0	0
ManagedElement	1..N	1..N	1..N	1
MeContext	0..M	0..M	0..M	0..1
RncFunction	0..P	0..P	0..1	0..1
NodeBFunction	0..Q	0..Q	0..(N-1)	0..1
IubLink	0..Q	0..Q	0..(N-1)	0
UtranCell	0..R	0..R	0..R	0..R
IRPAgent	1	1	1	1
NotificationIRP	1	1	1	1
AlarmIRP	0..1	0..1	0..1	0..1
BasicCmIRP	0..1	0..1	0..1	0..1

## Annex B (informative): Change history

Change history							
Date	TSG #	TSG Doc.	CR	Rev	Subject/Comment	Old	New
Jun 2001	S_12	SP-010283	--	--	Approved at TSG SA #12 and placed under Change Control	2.0.0	4.0.0
Jun 2002	S_16	SP-020303	001	--	Corrections of reference in figure 6.2 and of attribute descriptions in UtranRelation in 32.642 (UTRAN network resources IRP: NRM)	4.0.0	4.1.0
Jun 2002	S_16	SP-020304	002	--	Correction of supported IRP in system context	4.0.0	4.1.0
Sep 2002	S_17	SP-020490	003	--	UML corrections	4.1.0	4.2.0
Sep 2002	S_17	SP-020492	004	--	Add the new IRP IS methodology defined in 32.102	4.2.0	5.0.0
Sep 2002	S_17	SP-020492	005	--	Add State Management	4.2.0	5.0.0
Dec 2002	S_18	SP-020748	006	--	Inclusion of valid values and ranges for UTRAN Cell parameters	5.0.0	5.1.0
Jan 2003	--	--	--	--	Accepted all revision marks	5.1.0	5.1.1
Jun 2003	S_20	SP-030282	008	--	Include notification tables	5.1.1	5.2.0
Jun 2003	S_20	SP-030282	010	--	Correction of UML diagram vsDataContainer Containment/Naming and Association in UTRAN NRM	5.1.1	5.2.0
Jun 2003	S_20	SP-030283	012	--	Deletion of UTRAN attribute relationType	5.1.1	5.2.0