ETSI TS 132 663 V6.6.0 (2005-09)

Technical Specification

**Digital cellular telecommunications system (Phase 2+);
Universal Mobile Telecommunications System (UMTS);
Telecommunication management;
Configuration Management (CM);
Kernel CM Integration Reference Point (IRP):
Common Object Request Broker Architecture (CORBA)
Solution Set (SS)
(3GPP TS 32.663 version 6.6.0 Release 6)**

**GLOBAL SYSTEM FOR
MOBILE COMMUNICATIONS**

ETSI

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

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

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Important notice

Individual copies of the present document can be downloaded from:
http://www.etsi.org

The present document may be made available in more than one electronic version or in print. In any case of existing or perceived difference in contents between such versions, the reference version is the Portable Document Format (PDF). In case of dispute, the reference shall be the printing on ETSI printers of the PDF version kept on a specific network drive within ETSI Secretariat.

Users of the present document should be aware that the document may be subject to revision or change of status. Information on the current status of this and other ETSI documents is available at
http://portal.etsi.org/tb/status/status.asp

If you find errors in the present document, please send your comment to one of the following services:
http://portal.etsi.org/chaircor/ETSI_support.asp

Copyright Notification

Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (http://webapp.etsi.org/IPR/home.asp).

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

Foreword

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

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

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

Contents

Foreword

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

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

Version x.y.z

where:

x the first digit:

1 presented to TSG for information;

2 presented to TSG for approval;

3 or greater indicates TSG approved document under change control.

y the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.

z the third digit is incremented when editorial only changes have been incorporated in the document.

Introduction

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

32.661: "Configuration Management (CM); Kernel CM; Requirements".

32.662: "Configuration Management (CM); Kernel CM; Information Service (IS)".

32.663: **"Configuration Management (CM); Kernel CM Integration Reference Point (IRP): Common Object Request Broker Architecture (CORBA) Solution Set (SS)".**

32.664: "Configuration Management (CM); Kernel CM Integration Reference Point (IRP): Common Management Information Protocol (CMIP) Solution Set (SS)".

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.

1 Scope

The purpose of the present document is to define the mapping of the Kernel CM IRP: IS (see 3GPP TS 32.662 [4]) to the protocol specific details necessary for implementation of this IRP in a CORBA/IDL environment.

This Solution Set specification is related to 3GPP TS 32.662 V6.3.X [4].

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 32.600: "Telecommunication management; Configuration Management (CM); Concept and high-level requirements".

[4]   3GPP TS 32.662: "Telecommunication management; Configuration Management (CM); Kernel CM: Information Service (IS)".

[5]   3GPP TS 32.300: "Telecommunication management; Configuration Management (CM); Name convention for Managed Objects".

[6]   Object Management Group 98 (November 1998): "*Notification Service: Joint Revised Submission OMG TC Document telecom/98-11-01*".

[7]   OMG CORBA Services (November 1996): "Common Object Services Specification" (clause 4 contains the Event Service specification).

[8]   The Common Object Request Broker: Architecture and Specification (for specification of valid version, see [1]).

[9]   3GPP TS 32.303: "Telecommunication management; Configuration Management (CM); Notification Integration Reference Point (IRP): Common Object Request Broker Architecture (CORBA) Solution Set (SS)".

[10]   3GPP TS 32.111-3: "Telecommunication management; Fault Management; Part 3: Alarm Integration Reference Point (IRP): Common Object Request Broker Architecture (CORBA) Solution Set (SS)".

[11]   3GPP TS 32.312: "Telecommunication management; Generic Integration Reference Point (IRP) management: Information Service (IS)".

[12]   3GPP TS 32.673: "Telecommunication management; Configuration Management (CM); State Management Integration Reference Point (IRP): Common Object Request Broker Architecture (CORBA) Solution Set (SS)".

[13]   3GPP TS 32.311: "Telecommunication management; Generic Integration Reference Point (IRP) management: Requirements".

3   Definitions and abbreviations

3.1   Definitions

For the purposes of the present document, the terms and definitions given in 3GPP TS 32.101 [1], 3GPP TS 32.102 [2], 3GPP TS 32.600 [3] and 3GPP TS 32.662 [4] apply.

IRP document version number string (or "IRPVersion"): see 3GPP TS 32.311 [13].

3.2   Abbreviations

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

CORBA	Common Object Request Broker Architecture
DN	Distinguished Name
IDL	Interface Definition Language (OMG)
IRP	Integration Reference Point
IS	Information Service
MO	Managed Object
MOC	Managed Object Class
NRM	Network Resource Model
OMG	Object Management Group
SS	Solution Set
VSE	Vendor Specific Extensions

4   Void

5   Architectural features

The overall architectural feature of Kernel Configuration Management IRP is specified in 3GPP TS 32.662 [4]. This clause specifies features that are specific to the CORBA SS.

5.1   Notifications

Notifications are sent according to the Notification IRP: CORBA SS (see 3GPP TS 32.303 [9]).

The contents of the Kernel CM IRP notifications are defined in the present document.

5.2   Filter language

The filter language used in the SS is the Extended Trader Constraint Language (see OMG Notification Service [6]). IRPAgents may throw a FilterComplexityLimit exception when a given filter is too complex.

5.3   Syntax for Distinguished Names and Versions

The format of a Distinguished Name is defined in 3GPP TS 32.300 [5].

The version of this IRP is represented as a string (see also clause 3.1).

6 Mapping

6.1 Void

6.2 Operation and Notification mapping

The Kernel CM IRP: IS (see 3GPP TS 32.662 [4]) defines semantics of operation and notification visible across the Kernel Configuration Management IRP. The following table in this subclause indicates mapping of these operations and notifications to their equivalents defined in this SS.

Table 6.2.1: Mapping from IS Notification/Operation to SS equivalents

IS Operation/ notification (3GPP TS 32.662 [4])	SS Method	Qualifier
getNRMIRPVersion	get_nrm_irp_version	M
notifyObjectCreation	See Notification IRP: CORBA SS [9]	O
notifyObjectDeletion	See Notification IRP: CORBA SS [9]	O
notifyAttributeValueChange	See Notification IRP: CORBA SS [9]	O
notifyStateChange	See Notification IRP: CORBA SS [9]	O
getIRPVersion	get_kernel_cm_irp_versions	M
getOperationProfile	get_kernel_cm_irp operations_profile	O
getNotificationProfile	get_kernel_cm_irp_notification_profile	O
notifyCMSynchronizationRecommended	See Notification IRP: CORBA SS [9]	O

6.3 Operation parameter mapping

The Kernel CM IRP: IS (see 3GPP TS 32.662 [4]) defines semantics of parameters carried in operations across the Kernel Configuration Management IRP. The following tables in this subclause indicate the mapping of these parameters, as per operation, to their equivalents defined in this SS.

Table 6.3.1: Mapping from IS `getNRMIRPVersion` parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
versionNumberList	ManagedGenericIRPConstDefs::VersionNumberSet version_number_list	M
vSEVersionNumberList	ManagedGenericIRPConstDefs::VersionNumberSet vse_version_number_list	M
status	Exceptions: GetNRMIRPVersion	M

Table 6.3.2: Mapping from IS `getIRPVersion` parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
versionNumberList	Return value of type ManagedGenericIRPConstDefs::VersionNumberSet	M
status	exception GetKernelCmIRPVersionsException	M

Table 6.3.3: Mapping from IS `getOperationProfile` parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
irpVersion	ManagedGenericIRPConstDefs::VersionNumber kernel_cm_irp_version	M
operationNameProfile, operationParameterProfile	Return value of type ManagedGenericIRPConstDefs::MethodList	M
status	Exceptions: GetKernelCMIRPOperationsProfileException, ManagedGenericIRPSystem::OperationNotSupported, ManagedGenericIRPSystem::InvalidParameter	M

Table 6.3.4: Mapping from IS `getNotificationProfile` parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
irpVersion	ManagedGenericIRPConstDefs::VersionNumber kernel_cm_irp_version	M
notificationNameProfile, notificationParameterProfile	Return value of type ManagedGenericIRPConstDefs::MethodList	M
status	Exceptions: GetKernelCMIRPNotificationProfileException, ManagedGenericIRPSystem::OperationNotSupported, ManagedGenericIRPSystem::InvalidParameter	M

6.4 Notification attribute mapping

The Kernel CM IRP: IS (see 3GPP TS 32.662 [4]) identifies and defines the semantics of attributes for `notifyObjectCreation`, `notifyObjectDeletion`, `notifyAttributeValueChange`, `notifyStateChange` and `notifyCMSynchronizationRecommended` for use for its IRP. The following table in this subclause shows the mapping of the IS notifications to SS equivalents.

Table 6.4.1: Mapping from IS notifications to SS equivalents

IS notifications in 3GPP TS 32.662 [4]	SS notifications	Qualifier
NotifyObjectCreation	push_structured_event	O
NotifyObjectDeletion	push_structured_event	O
NotifyAttributeValueChange	push_structured_event	O
NotifyStateChange	push_structured_event	O
NotifyCMSynchronizationRecommended	push_structured_event	O

The Kernel CM IRP: IS (see 3GPP TS 32.662 [4]) also qualifies the attributes. The following tables in this subclause show the mapping of these IS attributes to SS equivalents.

Table 6.4.2: Mapping from IS `Notification Header` attributes to SS equivalent

IS Attribute of Notification Header in 3GPP TS 32.662 [4]	SS Attribute	Qualifier
objectClass	KernelCmNotifications::NotificationCommon::MANAGED_OBJECT_CLASS	M
objectInstance	KernelCmNotifications::NotificationCommon::MANAGED_OBJECT_INSTANCE	M
notificationId	KernelCmNotifications::NotificationCommon::NOTIFICATION_ID	M
eventTime	KernelCmNotifications::NotificationCommon::EVENT_TIME	M
systemDN	KernelCmNotifications::NotificationCommon::SYSTEM_DN	O
notificationType	header.fixed_header.event_type.type_name	M

Table 6.4.3: Mapping from IS `notifyObjectCreation` attributes to SS equivalent OBJECT_CREATION

IS Attribute of notifyObjectCreation in 3GPP TS 32.662 [4]	SS Attribute	Qualifier
notificationHeader	See table 6.4.2	M
correlatedNotifications	KernelCmNotifications::MOCreation::CORRELATED_NOTIFICATIONS	O
additionalText	KernelCmNotifications::MOCreation::ADDITIONAL_TEXT	O
sourceIndicator	KernelCmNotifications::MOCreation::SOURCE_INDICATOR	O
attributeList	KernelCmNotifications::MOCreation::InitialAttributeValues (contained in remainder_of_body)	O

**Table 6.4.4: Mapping from IS `notifyObjectDeletion` attributes
to SS equivalent OBJECT_DELETION**

IS Attribute of `notifyObjectDeletion` in 3GPP TS 32.662 [4]	SS Attribute	Qualifier
notificationHeader	See table 6.4.2	M
correlatedNotifications	KernelCmNotifications::MODeletion::CORRELATED_NOTIFICATIONS	O
additionalText	KernelCmNotifications::MODeletion::ADDITIONAL_ TEXT	O
sourceIndicator	KernelCmNotifications::MODeletion::SOURCE_INDICATOR	O
attributeList	KernelCmNotifications::MODeletion::AttributeValues (contained in remainder_of_body)	O

**Table 6.4.5: Mapping from IS `notifyAttributeValueChange` attributes
to SS equivalent ATTRIBUTE_VALUE_CHANGE**

IS Attribute of `notifyAttributeValue Change` in 3GPP TS 32.662 [4]	SS Attribute	Qualifier
notificationHeader	See table 6.4.2	M
correlatedNotifications	KernelCmNotifications::AttributeValueChange::CORRELATED_NOTIFICATIONS	O
additionalText	KernelCmNotifications::AttributeValueChange::ADDITIONAL_TEXT	O
sourceIndicator	KernelCmNotifications::AttributeValueChange::SOURCE_INDICATOR	O
attributeValueChangeDefinition	KernelCmNotifications::AttributeValueChange::ModifiedAttributeSet (contained in remainder_of_body)	M

**Table 6.4.6: Mapping from IS `notifyCMSynchronizationRecommended` attributes
to SS equivalent REQUEST_CM_SYNCHRONIZATION**

IS Attribute of notifyCMSynchronization Recommended in 3GPP TS 32.662 [4]	SS Attribute	Qualifier
notificationHeader	See table 6.4.2	M
baseMOClass	KernelCmNotifications::CMSynchronizationRecommended::BASE_MO_CLASS	M
baseMOInstance	KernelCmNotifications::CMSynchronizationRecommended::BASE_MO_INSTANCE	M
scope	KernelCmNotifications::CMSynchronizationRecommended::SCOPE	M
additionalText	KernelCmNotifications::CMSynchronizationRecommended::ADDITIONAL_TEXT	O

**Table 6.4.7: Mapping from IS `notifyStateChange` attributes
to SS equivalent STATE_CHANGE**

IS Attribute of `notifyStateChange Change` in 3GPP TS 32.662 [4]	SS Attribute	Qualifier
notificationHeader	See table 6.4.2	M
stateChange	StateManagementIRPConstDefs::AttributeNameValue (see note)	M
correlatedNotifications	KernelCmNotifications::StateChange::CORRELATED_NOTIFICATIONS	O
additionalText	KernelCmNotifications::StateChange::ADDITIONAL_TEXT	O
sourceIndicator	KernelCmNotifications::StateChange::SOURCE_INDICATOR	O
NOTE: The stateChange attribute is mapped into name-value pairs that contain the state identifier in the name and the new and optional old state values in the attribute field (See TS 32.673 [12] StateManagementIRPConstDefs IDL *<State Name>OldNewValue* structures).		

7 Use of OMG Structured Event

In CORBA SS, OMG defined `StructuredEvent` (see OMG Notification Service [6]) is used to carry notifications. This clause identifies the OMG defined `StructuredEvent` attributes that carry the attributes of notifications defined in 3GPP TS 32.662 [4].

The composition of OMG Structured Event, as defined in OMG Notification Service [6], is:

```
Header
       Fixed Header
               domain_name
               type_name
               event_name
       Variable Header
Body
       filterable_body_fields
       remainder_of_body
```

The following table in this clause lists all OMG Structured Event attributes in its leftmost column. The second column identifies the SS attributes, if any, that shall be carried there.

Attributes that are denoted as "optional" may be absent from the OMG Structured Event. As an example, if the optional `additionalText` attribute is not used for a particular notification, then the IRPAgent may exclude `additionalText` from the filterable body fields for that particular notification. Individual notifications from the same IRPAgent may include or exclude the same optional attribute.

Table 7.1: Use of OMG Structured Event

SS Attribute	OMG CORBA Structured Event attribute	Comment
There is no corresponding SS attribute	domain_name	It contains the supported SS document version (see clause 4). This version is defined by the string constant KernelCmIRPSystem::VERSION defined in this specification.
eventType	type_name	It is an attribute of notificationHeader. It shall indicate one of the following: Object Creation, Object Deletion, Attribute Value Change, State Change and CM Synchronization Recommended. It is a string. Its value is either defined by KernelCmNotifications::MOCreation::EVENT_TYPE, KernelCmNotifications::MODeletion::EVENT_TYPE, KernelCmNotifications::AttributeValueChange::EVENT_TYPE, KernelCmNotifications::StateChange::EVENT_TYPE or KernelCmNotifications::CMSynchronizationRecommended::EVENT_TYPE
-	event_name	Shall be set to an empty string
There is no corresponding SS attribute	variable Header	
managedObjectClass, managedObjectInstance	One NV pair of filterable_body_fields	NV stands for name-value pair. Order arrangement of NV pairs is not significant. The name of NV-pair is always encoded in string. They are attributes of notificationHeader. Name of NV pair is a string, KernelCmNotifications::<interface>::MANAGED_OBJECT_INSTANCE where <interface> is either MOCreation, MODeletion, AttributeValueChange, StateChange or CMSynchronizationRecommended. Value of NV pair is a string. This string conveys the semantics of both the Managed Object Class and the Managed Object Instance. See corresponding table in Notification IRP: CORBA SS (3GPP TS 32.303 [9]).
notificationId	One NV pair of remainder_of_body	It is an attribute of notificationHeader. Name of NV pair is a string, KernelCmNotifications::<interface>::NOTIFICATION_ID where

SS Attribute	OMG CORBA Structured Event attribute	Comment
		<interface> is either MOCreation, MODeletion, AttributeValueChange, StateChange or CMSynchronizationRecommended. Value of NV pair is a long. See corresponding table in Notification IRP: CORBA SS (3GPP TS 32.303 [9]).
eventTime	One NV pair of filterable_body_fields	It is an attribute of notificationHeader. Name of NV pair is a string, KernelCmNotifications::<interface>::EVENT_TIME where <interface> is either MOCreation, MODeletion, AttributeValueChange, StateChange or CMSynchronizationRecommended. Value of NV pair is a ManagedGenericIRPConstDefs::IRPTime defined in 3GPP TS 32.303 [9]. See corresponding table in Notification IRP: CORBA SS (3GPP TS 32.303 [9]).
systemDN	One NV pair of filterable_body_fields	It is an attribute of notificationHeader. Name of NV pair is a string, KernelCmNotifications::<interface>::SYSTEM_DN where <interface> is either MOCreation, MODeletion, AttributeValueChange, StateChange or CMSynchronizationRecommended. Value of NV pair is a string. See corresponding table in Notification IRP: CORBA SS [9].
correlatedNotifications	One NV pair of remainder_of_body	It is an attribute of the Object Creation, Object Deletion and Attribute Value Change notifications. Name of NV pair is a string, KernelCmNotifications::<interface>::CORRELATED_NOTIFICATIONS where <interface> is either MOCreation, MODeletion, StateChange or AttributeValueChange. Value of NV pair is a NotificationIRPConstDefs::CorrelatedNotificationSet defined in 3GPP TS 32.303 [9].
additionalText	One NV pair of remainder_of_body	It is an attribute of the Object Creation, Object Deletion, Attribute Value Change and CM Synchronization Recommended notifications. Name of NV pair is a string, KernelCmNotifications::<interface>::ADDITIONAL_TEXT where <interface> is either MOCreation, MODeletion, AttributeValueChange, StateChange or CMSynchronizationRecommended. Value of NV pair is a string.
sourceIndicator	One NV pair of remainder_of_body	It is an attribute of the Object Creation, Object Deletion and Attribute Value Change notifications. Name of NV pair is a string, KernelCmNotifications::<interface>::SOURCE_INDICATOR where <interface> is either MOCreation, MODeletion, StateChange or AttributeValueChange. Value of NV pair is a string with values of either KernelCmNotifications::<interface>::RESOURCE_OPERATION, KernelCmNotifications::<interface>::MANAGEMENT_OPERATION or KernelCmNotifications::<interface>::UNKNOWN_OPERATION where <interface> is either MODeletion, MOCreation, StateChange or AttributeValueChange.
attributeList	remainder_of_non_filterable_body (see 3GPP TS 32.303 [9])	Is used to transport attribute information. For Object Creation notification, this is defined by KernelCmNotifications::MOCreation::InitialAttributeValues. For Object Deletion notification, this is defined by KernelCmNotifications::MODeletion::AttributeValues. The name component of InitialAttributeValues and AttributeValues will be set to attribute names defined in KernelCmNRMDefs.
attributeValueChangeDefinition	remainder_of_non_filterable_body (see 3GPP TS 32.303 [9])	For Attribute Value Change notification, this is defined by KernelCmNotifications::AttributeValueChange::ModifiedAttributeSet. The name component of ModifiedAttributeSet will be set to attribute name defined in KernelCmNRMDefs.
StateManagementIRPConstDefs::AttributeNameValue	A set of up to 9 Name-value pairs See table 7.2. All	For state change notifications a series of up to 9 name-value pairs might be sent corresponding with the new and old values of each state/status attribute which has changed it's value.

SS Attribute	OMG CORBA Structured Event attribute	Comment
	these 9 NV pairs are part of the remainder_of_body	The new values of each state/status attributes that have changed are sent. The IRP agent may optionally send the old state/status changes. The name of the name-value pairs are defined by StateManagementIRPConstDefs::AttributeNameValue
baseMOClass	One NV pair of remainder_of_body	It is an attribute of the CMSynchronizationRecommended notification. Name of NV pair is a string, KernelCmNotifications::CMSynchronizationRecommended::BASE_MO_CLASS. Value of NV pair is a string. This string conveys the semantics of the Managed Object Class.
baseMOInstance	One NV pair of remainder_of_body	It is an attribute of the CMSynchronizationRecommended notification. Name of NV pair is a string, KernelCmNotifications::CMSynchronizationRecommended::BASE_MO_INSTANCE. Value of NV pair is a string. This is the DN string of the Managed Object Instance.
scope	One NV pair of remainder_of_body	It is an attribute of the CMSynchronizationRecommended notification. Name of NV pair is a string, KernelCmNotifications::CMSynchronizationRecommended::SCOPE. Value of NV pair is KernelCmConstDefs::ScopePara.

Table 7.2 Name – value pairs for state change notifications

Name	Value
OPERATIONAL_STATE	StateManagementIRPConstDefs::OperationalStateOldNewValue
USAGE_STATE	StateManagementIRPConstDefs::UsageStateOldNewValue
ADMINISTRATIVE_STATE	StateManagementIRPConstDefs::AdministrativeStateOldNewValue
ALARM_STATUS	StateManagementIRPConstDefs::AlarmStatusOldNewValue
PROCEDURAL_STATUS	StateManagementIRPConstDefs::ProceduralStatusOldNewValue
AVAILABILITY_STATUS	StateManagementIRPConstDefs::AvailabilityStatusOldNewValue
CONTROL_STATUS	StateManagementIRPConstDefs::ControlStatusOldNewValue
STANDBY_STATUS	StateManagementIRPConstDefs::StandbyStatusOldNewValue
UNKNOWN_STATUS	StateManagementIRPConstDefs::UnknownStatusOldNewValue

8 Rules for NRM extensions

This clause discusses how the models and IDL definitions provided in the present document can be extended for a particular implementation and still remain compliant with 3GPP SA5's specifications.

8.1 Void

8.2 Extensions not allowed

The IDL specifications in the present document cannot be edited or altered. Any additional IDL specifications shall be specified in separate IDL files.

IDL interfaces (note: not MOCs) specified in the present document may not be subclassed or extended. New interfaces may be defined with vendor-specific methods.

Annex A (normative):
IDL specification (file name "KernelCmConstDefs.idl")

```
//File: KernelCmConstDefs.idl
#ifndef _KERNEL_CM_CONST_DEFS_IDL_
#define _KERNEL_CM_CONST_DEFS_IDL_

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

module KernelCmConstDefs
{
    /**
     * Information about one attribute
     * - name defines the name of the attribute
     * - value defines the value of the attribute
     *
     */
    struct MOAttribute
    {
        string name;
        any value;
    };

    /**
     * A set of attribute names and values
     */
    typedef sequence<MOAttribute> MOAttributeSet;

    /**
     * ScopeType defines the kind of scope to use in a CM synchronization
     * request together with ScopePara.level, in the SCOPE field.
     *
     * ScopePara.level is always >= 0. If a level is bigger than the
     * depth of the tree there will be no exceptions thrown.
     * BASE_ONLY: level ignored, just return the base object.
     * BASE_NTH_LEVEL: return all subordinate objects that are on "level"
     * distance from the base object, where 0 is the base object.
     * BASE_SUBTREE: return the base object and all of its subordinates
     * down to and including the nth level.
     * BASE_ALL: level ignored, return the base object and all of it's
     * subordinates.
     */
    enum ScopeType
    {
        BASE_ONLY,
        BASE_NTH_LEVEL,
        BASE_SUBTREE,
        BASE_ALL
    };

    struct ScopePara
    {
        ScopeType type;
        unsigned long level;
    };

    /* The format of Distinguished Name (DN) is specified in 3GPP TS 32.300
    "Name Conventions for Managed Objects".
    */
    typedef string DN;

    typedef sequence <long> NotifIdSet;

    /*
    This holds identifiers of notifications that are correlated.
    */
    struct CorrelatedNotification
    {
        DN source;  // Contains DN of MO that emitted the set of notifications
                    // DN string format in compliance with Name Convention for
                    // Managed Object.
                    // This may be a zero-length string. In this case, the MO
                    // is identified by the value of the MOI attribute
```

```
                   // of the Structured Event, i.e., the notification.
       NotifIdSet notif_id_set;  // Set of related notification ids
   };

   /*
   Correlated Notification sets are sets of Correlated Notification
   structures.
   */
   typedef sequence <CorrelatedNotification> CorrelatedNotificationSet;

   /*
   This block identifies attributes which are included as part of the Kernel
   CM IRP. These attribute values should not clash with those defined for the
   attributes of notification header (see IDL of Notification IRP).
   */
   interface AttributeNameValue
   {
       const string SOURCE_INDICATOR = "SOURCE";
       const string ADDITIONAL_TEXT = "ADD_TEXT";
       const string CORRELATED_NOTIFICATIONS = "CORREL_NOTIFS";
       const string BASE_MO_CLASS = "BASE_MOC";
       const string BASE_MO_INSTANCE = "BASE_MOI";
       const string SCOPE = "SCOPE";
   };

};

#endif // _KERNEL_CM_CONST_DEFS_IDL_
```

Annex B (normative):
IDL specification (file name "KernelCmIRPSystem.idl")

```
//File: KernelCmIRPSystem.idl
#ifndef _KERNEL_CM_IRP_SYSTEM_IDL_
#define _KERNEL_CM_IRP_SYSTEM_IDL_

#include <ManagedGenericIRPConstDefs.idl>
#include <ManagedGenericIRPSystem.idl>

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

module KernelCmIRPSystem
{
    exception GetKernelCMIRPNotificationProfileException { string reason; };
    exception GetKernelCMIRPOperationsProfileException { string reason; };
    exception GetNRMIRPVersion { string reason; };
    exception GetKernelCMIRPVersionsException { string reason; };

    /**
     * The KernelCmIrpOperations interface.
     * Supports a number of Resource Model versions.
     */
    interface KernelCmIrpOperations
    {
        /*
        Return the list of all supported Kernel CM IRP versions.
        */
        ManagedGenericIRPConstDefs::VersionNumberSet get_kernel_cm_irp_versions (
        )
        raises (GetKernelCMIRPVersionsException);

        /**
         * Get the version(s) of the interface
         *
         * @raises GetNRMIRPVersion when the system for some reason
         *    can not return the supported versions.
         * @returns all supported versions.
         */
        void get_nrm_irp_version
        (
            out ManagedGenericIRPConstDefs::VersionNumberSet version_number_list,
            out ManagedGenericIRPConstDefs::VersionNumberSet vse_version_number_list
        )
        raises (GetNRMIRPVersion);

        /*
        Return the list of all supported operations and their supported
        parameters for a specific KernelCM IRP version.
        */
        ManagedGenericIRPConstDefs::MethodList get_kernel_cm_irp_operations_profile (
            in ManagedGenericIRPConstDefs::VersionNumber kernel_cm_irp_version
        )
        raises (GetKernelCMIRPOperationsProfileException,
                ManagedGenericIRPSystem::OperationNotSupported,
                ManagedGenericIRPSystem::InvalidParameter);

        /*
        Return the list of all supported notifications and their supported
        parameters for a specific KernelCM IRP version.
        */
        ManagedGenericIRPConstDefs::MethodList
            get_kernel_cm_irp_notification_profile
        (
            in ManagedGenericIRPConstDefs::VersionNumber kernel_cm_irp_version
        )
        raises (GetKernelCMIRPNotificationProfileException,
                ManagedGenericIRPSystem::OperationNotSupported,
                ManagedGenericIRPSystem::InvalidParameter);

    };
};
#endif // _KERNEL_CM_IRP_SYSTEM_IDL_
```

Annex C (normative):
IDL specification (file name "KernelCmNotifications.idl")

```
//File: KernelCmNotifications.idl
#ifndef _KERNEL_CM_NOTIFICATIONS_IDL_
#define _KERNEL_CM_NOTIFICATIONS_IDL_

#include <KernelCmConstDefs.idl>
#include <NotificationIRPNotifications.idl>

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

module KernelCmNotifications
{
   /**
    *  This interface defines fields that are common for all
    *  notification types.
    *  All constants in the scope of this interface will be
    *  visible in the interfaces that inherits this.
    *  For instance constant
    *  NotificationCommon::MANAGED_OBJECT_CLASS
    *  can be addressed by MODeletion::MANAGED_OBJECT_CLASS
    */
   interface NotificationCommon: NotificationIRPNotifications::Notify
   {

      /**
       *  This constant defines the name of the
       *  source indicator property.
       */
      const string SOURCE_INDICATOR =
         KernelCmConstDefs::AttributeNameValue::SOURCE_INDICATOR;

      /**
       * Valid values for the SOURCE_INDICATOR
       * property
       */
      const string RESOURCE_OPERATION = "RESOURCE OPERATION";
      const string MANAGEMENT_OPERATION = "MANAGEMENT OPERATION";
      const string UNKNOWN_OPERATION = "UNKNOWN";

      /**
       *  This constant defines the name of the
       *  additional text property.
       *  The data type for the value of this property
       *  is a string.
       */
      const string ADDITIONAL_TEXT =
         KernelCmConstDefs::AttributeNameValue::ADDITIONAL_TEXT;

      /**
       *  This constant defines the name of the
       *  correlated notifications property.
       *  The value part of the property is
       *    KernelCmConstDefs::CorrelatedNotificationSet
       */
      const string CORRELATED_NOTIFICATIONS =
         KernelCmConstDefs::AttributeNameValue::CORRELATED_NOTIFICATIONS;
   };

   /**
    *  Constant definitions for the MO deleted notification
    */
   interface MODeletion : NotificationCommon
   {
      const string EVENT_TYPE = "x7";

      /**
       * This information mapped into the remainder_of_body
       * in the StructuredEvent
       */
      typedef KernelCmConstDefs::MOAttributeSet AttributeValues;
   };
```

```
/**
 * Constant definitions for the MO created notification
 */
interface MOCreation : NotificationCommon
{
    const string EVENT_TYPE = "x6";

    /**
     * This information mapped into the remainder_of_body
     * in the StructuredEvent
     */
    typedef KernelCmConstDefs::MOAttributeSet InitialAttributeValues;
};

/**
 * Constant definitions for the Attribute Value Change
 * notification
 */
interface AttributeValueChange : NotificationCommon
{
    const string EVENT_TYPE = "x8";

    /**
     * Information about modified attributes for
     * one MO instance.
     * - name defines the name of the attribute
     * - new_value defines the new value of the attribute
     * - old_value defines the previous value of the attribute
     *    The value is optional, which means that it may contain
     *    an empty any (null inserted in the any).
     *
     */
    struct ModifiedAttribute
    {
       string name;
       any new_value;
       any old_value;
    };

    /**
     * This information mapped into the remainder_of_body
     * in the StructuredEvent.
     */
    typedef sequence<ModifiedAttribute> ModifiedAttributeSet;
};

/**
 * Constant definitions for the CM Synchronization Recommended notification
 */
interface CMSynchronizationRecommended: NotificationIRPNotifications::Notify
{
    const string EVENT_TYPE = "x9";

    /**
     * This constant defines the name of the
     * additional text property.
     * The data type for the value of this property
     * is a string.
     */
    const string ADDITIONAL_TEXT =
       KernelCmConstDefs::AttributeNameValue::ADDITIONAL_TEXT;

    /**
     * This constant defines the name of the
     * base MO class property.
     * The value part of this property will carry
     * the base MO class name as a string.
     */
    const string BASE_MO_CLASS =
       KernelCmConstDefs::AttributeNameValue::BASE_MO_CLASS;

    /**
     * This constant defines the name of the
     * base MO instance property.
     * The value part of this property will carry
     * the base MO distinguished name as a string.
     */
    const string BASE_MO_INSTANCE =
```

```
        KernelCmConstDefs::AttributeNameValue::BASE_MO_INSTANCE;

     /**
      *  This constant defines the name of the
      *  scope property.
      *  The data type for the value of this property
      *  is KernelCmConstDefs::ScopePara.
      */
     const string SCOPE =
        KernelCmConstDefs::AttributeNameValue::SCOPE;
  };

  /**
   *  Constant definitions for the State Change notification
   */
  interface StateChange : NotificationCommon
  {
     const string EVENT_TYPE = "xA";
  };

};

#endif // _KERNEL_CM_NOTIFICATIONS_IDL_
```

Annex D (informative): Change history

Date	TSG #	TSG Doc.	CR	Rev	Subject/Comment	Cat	Old	New
Sep 2002	SA_17	SP-020466	--	--	Submitted to TSG SA #17 for Approval	--	1.0.0	5.0.0
Mar 2003	SA_19	SP-030143	0001	--	CORBA IDL Compiler Errors	F	5.0.0	5.1.0
Mar 2003	SA_19	SP-030145	0002	--	Add IDL definition of notifyCMSynchronizationRecommended notification for KernelCM IRP	B	5.1.0	6.0.0
Jun 2004	SA_24	SP-040261	0004	--	Add Missing CorrelatedNotificationSetType definition	A	6.0.0	6.1.0
Sep 2004	SA_25	SP-040568	0007	--	Add missing DN definition	A	6.1.0	6.2.0
Sep 2004	SA_25	SP-040568	0009	--	Add missing IDL for get_kernel_CM_IRP_versions	A	6.1.0	6.2.0
Sep 2004	SA_25	SP-040569	0010	--	Add State Management Support to Kernel CM IRP CORBA SS	B	6.1.0	6.2.0
Dec 2004	SA_26	SP-040812	0011	--	Correct the mapping of IS-defined non-filterable parameters to SS-defined non-filterable fields (instead of filterable fields) - Align with IS in 32.662	F	6.2.0	6.3.0
Mar 2005	SA_27	SP-050050	0012	--	IDL incompliant to the style guide	F	6.3.0	6.4.0
Jun 2005	SA_28	SP-050299	0013	--	Correct CORBA SS mapping of notification filterable/non-filterable IS parameters	F	6.4.0	6.5.0
Sep 2005	SA_29	SP-050461	0014	--	Align the CORBA SS IDL with TS 32.150 Style Guide	F	6.5.0	6.6.0

History

Document history		
V6.3.0	December 2004	Publication
V6.4.0	March 2005	Publication
V6.5.0	June 2005	Publication
V6.6.0	September 2005	Publication