Technical Specification Group Services and System Aspects ***TSGS#11(01)010030***
Meeting #11, Palm Springs, CA, USA, 19-22 March 2001

Source: **SA WG5**

Title: **CRs to Configuration Management; Part 6: Basic Configuration Management IRP CORBA Solution Set Version 1:1 (32.106-6)**

Document for: **Approval**

Agenda Item: **7.5.3**

Doc-	Doc-	Spec	CR	R	Pha	Subject	Cat	Version-	Version-	Workite
SP-010030	S5-010045	32.106-6	001		R99	**Remove TimeBase.idl not used in the module NotificationDefs**	F	3.0.0	3.1.0	OAM-CM
SP-010030	S5-010046	32.106-6	002		R99	**Update get_basicCm_IRP_version to be consistent with Alarm IRP and Notification IRP**	F	3.0.0	3.1.0	OAM-CM
SP-010030	S5-010047	32.106-6	003		R99	**Mismatched irpVersion types**	F	3.0.0	3.1.0	OAM-CM
SP-010030	S5-010048	32.106-6	004		R99	**Update Basic CM IRP Iterator to be consistent with Alarm IRP Iterator**	F	3.0.0	3.1.0	OAM-CM
SP-010030	S5-010049	32.106-6	005		R99	**Removing nested IDL modules**	F	3.0.0	3.1.0	OAM-CM
SP-010030	S5-010051	32.106-6	006		R99	**Update Structured Event table to be consistent with Alarm IRP**	F	3.0.0	3.1.0	OAM-CM
SP-010030	S5-010134	32.106-6	007		R99	**UMTS Network Resource Model alignment with TSG RAN specifications**	F	3.0.0	3.1.0	OAM-CM

CR-Form-v3

CHANGE REQUEST

⌘ **32.106-6** **CR** **001** ⌘ rev **-** ⌘ Current version: **3.0.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: ⌘ (U)SIM ☐ ME/UE ☐ Radio Access Network **X** Core Network **X**

Title:	⌘	Remove **TimeBase.idl** not used in the module NotificationDefs
Source:	⌘	SA5
Work item code:	⌘	OAM-CM
Date:	⌘	02/03/2001
Category:	⌘	**F**
Release:	⌘	**R99**

Use one of the following categories:
- ***F*** *(essential correction)*
- ***A*** *(corresponds to a correction in an earlier release)*
- ***B*** *(Addition of feature),*
- ***C*** *(Functional modification of feature)*
- ***D*** *(Editorial modification)*

Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
REL-4	*(Release 4)*
REL-5	*(Release 5)*

Reason for change:	⌘	The `#include "TimeBase.idl"` in the `NotificationDefs` module is not used in the module.
Summary of change:	⌘	Remove `#include "TimeBase.idl"`
Consequences if not approved:	⌘	Mostly, confusion for why it was included. Many IDL compilers will take longer to compile.

Clauses affected:	⌘	Annex B
Other specs affected:	⌘	☐ Other core specifications ⌘ ☐ Test specifications ☐ O&M Specifications
Other comments:	⌘	

Annex B (normative): CORBA IDL, Notification Definitions

```
#ifndef NotificationDefs_idl
#define NotificationDefs_idl

#pragma prefix "3gppsa5.org"

#include <TimeBase.idl>           // CORBA Time Service
#include <NotificationIRPConstDefs.idl>
```

CR-Form-v3

CHANGE REQUEST

⌘ **32.106-6** **CR** **002** ⌘ rev **-** ⌘ Current version: **3.0.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: ⌘ (U)SIM ☐ ME/UE ☐ Radio Access Network **X** Core Network **X**

Title:	⌘	Update **get_basicCm_IRP_version** to be consistent with Alarm IRP and Notification IRP
Source:	⌘	SA5
Work item code:	⌘	OAM-CM
Date:	⌘	02/03/2001
Category:	⌘	**F**
Release:	⌘	**R99**

Use one of the following categories:
F *(essential correction)*
A *(corresponds to a correction in an earlier release)*
B *(Addition of feature),*
C *(Functional modification of feature)*
D *(Editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:
2 (GSM Phase 2)
R96 (Release 1996)
R97 (Release 1997)
R98 (Release 1998)
R99 (Release 1999)
REL-4 (Release 4)
REL-5 (Release 5)

Reason for change:	⌘	3GPP 32.111-3 V3.3.0 (Alarm IRP) and 3GPP 32.106-3 V3.2.0 (Notification IRP) have consistent exceptions for their `get_alarm_IRP_version` and `get_notification_IRP_version` methods. 3GPP 32.106-6 needs to be updated so the `get_basicCm_IRP_version` method matches the other specifications (32.111-3 V3.3.0 and 32.106-3 V3.2.0).
Summary of change:	⌘	Define the `GetBasicCmIRPVersion` exception and raise it in `get_alarm_IRP_version`.
Consequences if not approved:	⌘	Decreased consistency between different specifications. Currently, there is no way to indicate a failure of the `get_BasicCm_IRP_version` method.

Clauses affected:	⌘	Annex A
Other specs affected:	⌘	☐ Other core specifications ☐ Test specifications ☐ O&M Specifications
Other comments:	⌘	

Annex A (normative): CORBA IDL, Access Protocol

```
      /**
       *  A set of strings.
       */
      typedef sequence<string> StringSet;

   };



    exception IllegalFilterFormatException {
       string reason;
    };
    exception IllegalDNFormatException {
       string reason;
    };
    exception IllegalScopeTypeException {
       string reason;
    };
    exception IllegalScopeLevelException {
       string reason;
    };
    exception UndefinedMOException {
       string reason;
    };

   exception UndefinedScopeException {
       string reason;
   };

   exception FilterComplexityLimit {
      string reason;
   };
   exception GetBasicCmIRPVersion {
      string reason;
   };
```

```
    /**
     * The BasicCmIrpOperations interface.
     * Supports a number of Resource Model versions.
     */
    interface BasicCmIrpOperations
    {

       /**
        * Get the version of the interface and all supported resource
        * model versions.
        *
        * @returns all supported versions.
        */
       CommonIRPConstDefs::VersionNumberSet get_basicCm_IRP_version()
          raises (GetBasicCmIRPVersion);
```

CR-Form-v3

CHANGE REQUEST

⌘ **32.106-6** **CR** **003** ⌘ rev **-** ⌘ Current version: **3.0.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: ⌘ (U)SIM ☐ ME/UE ☐ Radio Access Network **X** Core Network **X**

Title:	⌘	Mismatched **`irpVersion`** types
Source:	⌘	SA5
Work item code:	⌘	OAM-CM
Date:	⌘	02/03/2001
Category:	⌘	**F**
Release:	⌘	**R99**

Use one of the following categories:
F *(essential correction)*
A *(corresponds to a correction in an earlier release)*
B *(Addition of feature),*
C *(Functional modification of feature)*
D *(Editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:
2 (GSM Phase 2)
R96 (Release 1996)
R97 (Release 1997)
R98 (Release 1998)
R99 (Release 1999)
REL-4 (Release 4)
REL-5 (Release 5)

Reason for change:	⌘	In 32.106-5 V3.0.0, the `irpVersion` attribute is in the `NotificationIRP`, `AlarmIRP` and `BasicCmIRP` managed objects. Using `NotificationIRP` managed object as an example, `irpVersion` is defined as one or more Notification IRP version entries. In 32.106-3 V3.2.0, version entries are declared as type **`CommonIRPConstDefs::VersionNumberSet`**. In 32.106-6, `irpVersion` is declared as type **`StringSet`**. This CR proposes that all version attributes be of the same type.
Summary of change:	⌘	Change the irpVersion type to be **`CommonIRPConstDefs::VersionNumberSet`**.
Consequences if not approved:	⌘	Application Developers will need to switch between the different types.

Clauses affected:	⌘	6.5.1.7, 6.5.1.8 and 6.5.1.9
Other specs affected:	⌘	☐ Other core specifications ⌘ ☐ Test specifications ☐ O&M Specifications
Other comments:	⌘	

6.5.1.7 MOC NotificationIRP

Table 15: Mapping from NRM MOC NotificationIRP attributes to SS equivalent MOC NotificationIRP attributes

NRM Attributes of MOC NotificationIRP in 3GPP TS 32.106-5 [4]	SS Attributes	SS Type	Qualifier
notificationIRPId	notificationIRPid	string	Read-Only, M
irpVersion	irpVersion	CommonIRPConstDefs::VersionNumberSet~~StringSet~~	Read-Only, M

6.5.1.8 MOC AlarmIRP

Table 16: Mapping from NRM MOC AlarmIRP attributes to SS equivalent MOC AlarmIRP attributes

NRM Attributes of MOC AlarmIRP in 3GPP TS 32.106-5 [4]	SS Attributes	SS Type	Qualifier
alarmIRPId	alarmIRPid	string	Read-Only, M
irpVersion	irpVersion	CommonIRPConstDefs::VersionNumberSet~~StringSet~~	Read-Only, M

6.5.1.9 MOC BasicCmIRP

Table 17: Mapping from NRM MOC BasicCmIRP attributes to SS equivalent MOC BasicCmIRP attributes

NRM Attributes of MOC BasicCmIRP in 3GPP TS 32.106-5 [4]	SS Attributes	SS Type	Qualifier
basicCmIRPId	basicCmIRPid	string	Read-Only, M
irpVersion	irpVersion	CommonIRPConstDefs::VersionNumberSet~~StringSet~~	Read-Only, M

CR-Form-v3

CHANGE REQUEST

⌘ **32.106-6** **CR** **004** ⌘ rev **-** ⌘ Current version: **3.0.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: ⌘ (U)SIM ☐ ME/UE ☐ Radio Access Network **X** Core Network **X**

Title:	⌘	Update Basic CM IRP Iterator to be consistent with Alarm IRP Iterator
Source:	⌘	SA5
Work item code:	⌘	OAM-CM
Date:	⌘	02/03/2001
Category:	⌘	**F**
Release:	⌘	**R99**

Use one of the following categories:
F *(essential correction)*
A *(corresponds to a correction in an earlier release)*
B *(Addition of feature),*
C *(Functional modification of feature)*
D *(Editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:
2 (GSM Phase 2)
R96 (Release 1996)
R97 (Release 1997)
R98 (Release 1998)
R99 (Release 1999)
REL-4 (Release 4)
REL-5 (Release 5)

Reason for change:	⌘	Both Alarm IRP (32.111-3 V3.3.0) and Basic CM define iterators. The Alarm IRP iterator was designed to match the iterator in the T1M1.5 / ITU-T CORBA framework. This CR proposes that the Basic CM iterator be adjusted to match the Alarm IRP iterator format and style.
Summary of change:	⌘	Update the iterator to match the corresponding iterator format given in 32.111-3.
Consequences if not approved:	⌘	Application developers will need to apply different criteria when dealing with 3GPP SA5 iterators.

Clauses affected:	⌘	6.2, 6.3, Annex A
Other specs affected:	⌘	☐ Other core specifications ⌘ ☐ Test specifications ☐ O&M Specifications
Other comments:	⌘	

6.2 Operation and Notification mapping

The IS part of Basic CM IRP: IM (see 3GPP TS 32.106-5 [4]) defines semantics of operation and notification visible across the Basic Configuration Management IRP. Table 1 indicates mapping of these operations and notifications to their equivalents defined in this SS.

Table 1: Mapping from IS Notification/Operation to SS equivalents

IS Operation/ notification (3GPP TS 32.106-5 [4])	SS Method	Qualifier
getMoAttributes	BasicCmIrpOperations::find_managed_objects BasicCmInformationIterator~~Iterator~~::~~get_next_elements~~next_basicCmInformations BasicCmInformationIterator~~Iterator~~::destroy	M
getContainment	BasicCmIrpOperations::find_managed_objects BasicCmInformationIterator~~Iterator~~::~~get_next_elements~~next_basicCmInformations BasicCmInformationIterator~~Iterator~~::destroy	O
getBasicCmIRPVersion	get_basicCm_IRP_version	M
notifyObjectCreation (to convey of a new Managed Object created)	See Notification IRP: CORBA SS [9]	O
notifyObjectDeletion (to convey of a new Managed Object deleted)	See Notification IRP: CORBA SS [9]	O
notifyAttributeValueChange (to convey of a change of one or several attributes of a Managed Object)	See Notification IRP: CORBA SS [9]	O

6.3 Operation parameter mapping

The IS part of Basic CM IRP: IM (see 3GPP TS 32.106-5 [4]) defines semantics of parameters carried in operations across the Basic Configuration Management IRP. Tables 2, 3 and 4 indicate the mapping of these parameters, as per operation, to their equivalents defined in this SS.

The SS operation find_managed_objects is equivalent to the IS operation getMoAttributes when called with ResultContents set to NAMES_AND_ATTRIBUTES. Iterating the BasicCmInformationIterator~~Iterator~~ is used to fetch the result.

Table 2: Mapping from IS getMoAttributes parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
baseObjectInstance	in DN baseObject	M
scope	in searchControl (SearchControl.scope and SearchControl.level)	M
filter	in searchControl (SearchControl.filter)	M
attributeListIn	in requestedAttributes	M
managedObjectClass managedObjectInstance attributeListOut	parameter fetchedElements in the ~~get_next_elements~~next_basicCmInformations in the BasicCmInformationIterator~~Iterator~~ interface.	M
status	exception UndefinedMOException, exception IllegalDNFormatException, exception UndefinedScopeException, exception IllegalScopeTypeException, exception IllegalScopeLevelException, exception IllegalFilterFormatException,	M

	exception FilterComplexityLimit	

The SS operation find_managed_objects is equivalent to the IS operation getContainment when called with ResultContents set to NAMES. Iterating the BasicCmInformationIterator~~Iterator~~ is used to fetch the result.

Table 3: Mapping from IS getContainment parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
baseObjectInstance	in DN baseObject	M
scope	in searchControl (SearchControl.scope and SearchControl.level)	O
Not specified in IS	in searchControl (SearchControl.filter)	M
containment	parameter fetchedElements in the ~~get_next_elements~~next_basicCmInformations in the BasicCmInformationIterator~~Iterator~~ interface.	M
status	exception UndefinedMOException, exception IllegalDNFormatException, exception UndefinedScopeException, exception IllegalScopeTypeException, exception IllegalScopeLevelException, exception IllegalFilterFormatException, exception FilterComplexityLimit	M

Annex A (normative): CORBA IDL, Access Protocol

```
   exception UndefinedScopeException {
       string reason;
   };

   exception FilterComplexityLimit {
      string reason;
   };

   exception NextBasicCmInformations {
      string reason;
   };

   exception InvalidParameter {
      string parameter;
   };



    typedef sequence<Result> ResultSet;


    /**

     The BasicCmInformationIterator is used to iterate through a snapshot of
      Managed Object Information when IRPManager invokes find_managed_objects.
      IRPManager uses it to pace the return of Managed Object Information.

     IRPAgent controls the life-cycle of the iterator. However, a destroy
      operation is provided to handle the case where IRPManager wants to stop
      the iteration procedure before reaching the last iteration.     * Iterator
interface
     *
     *
     */
    interface BasicCmInformationIterator
    {
       exception IllegalCountException {
          string reason;
       };


       /**
        * Gets data from an Iterator.
        * This method returns between 1 and "how_many" elements
        * The IRPAgent may return less than "how_many" items even if
        * there are more items to send. "how_many" shall be non-zero.
        * Return TRUE if there are more elements to return.
        * Return FALSE if thereare no more elements to be returned.
        * Note that the IRPAgent may both provide the last items in the
        * Basic CM operation results and also indicate FALSE for completion.
        *
        * If FALSE is returned, the IRPAgent will automatically destroy the
        * iterator. Otherwise, it is the IRPManager's responsibility to
        * destroy the iterator.
        *
        *
        * @parm howMany how many elements to return in the "fetchedElements" out
```

~~* parameter.~~
~~* @parm fetchedElements the elements.~~
~~* @returns A boolean indicating if any elements are returned.~~
~~* "fetchedElements" is empty when the Iterator is empty.~~
~~*~~
~~* @raises IllegalCountException "howMany" has a value < 0.~~
~~*/~~
~~boolean get_next_elements(in unsigned short howMany,~~
~~out ResultSet fetchedElements)~~
~~raises (IllegalCountException);~~

~~/**~~
~~* Destroys an Iterator. This method shall be used if~~
~~* the iterator is to be removed before all elements~~
~~* are iterated.~~
~~*~~
~~*/~~
~~void destroy();~~

~~};~~

```
/**
 This method returns between 1 and “how_many” Managed Object information.
  The IRPAgent may return less than “how_many” items even if there are
 more items to return. “how_many” must be non-zero. Return TRUE if there
 may be more Managed Object information to return. Return FALSE if there
 are no more Managed Object information to be returned.

  If FALSE is returned, the IRPAgent will automatically destroy the
   iterator.

 @parm how_many how many elements to return in the "fetchedElements" out
   parameter.
 @parm fetchedElements the elements.
 @returns A boolean indicating if any elements are returned.
   "fetchedElements" is empty when the BasicCmInformationIterator is
   empty.
 */

 boolean next_basicCmInformations (
   in unsigned short how_many,
   out ResultSet fetchedElements
 )
 raises (NextBasicCmInformations,InvalidParameter);

 /**
 This method destroys the iterator.
 */

 void destroy ();

}; // end of BasicCmInformationIterator


typedef sequence<string> AttributeNameSet;


 /**
  * The BasicCmIrpOperations interface.
  * Supports a number of Resource Model versions.
  */
```

```
    interface BasicCmIrpOperations
    {

       /**
        * Get the version of the interface and all supported resource
        * model versions.
        *
        * @returns all supported versions.
        */
       CommonIRPConstDefs::VersionNumberSet get_basicCm_IRP_version();


       /**
        * Performs a containment search, using a SearchControl to
        * control the search and the returned results.
        *
        * All MOs in the scope constitute a set that the filter works on.
        * The result BasicCmInformationIteratorIterator contains all matched
MOs,
        * with the amount of detail specified in the SearchControl.
        * For the special case when no managed objects are matched in
        * find_managed_objects, the BasicCmInformationIteratorIterator will be
returned.
        * —Executing
        * the get_next_elementsnext_basicCmInformations in the
        * BasicCmInformationIteratorIterator will return FALSE for
        * completion.
        *
        * @parm baseObject The start MO in the containment tree.
        * @parm searchControl the SearchControl to use.
        * @parm requestedAttributes defines which attributes to get.
        *   If this parameter is empty (""),  all attributes shall
        *   be returned. Note: In R99 this is the only supported semantics.
        *   Note that this argument is only
        *   relevant if ResultContents in the search control is
        *   specifed to NAMES_AND_ATTRIBUTES.
        *
        *
        * @raises UndefinedMOException The MO does not exist.
        * @raises IllegalDNFormatException The dn syntax string is
        * malformed.
        * @raises IllegalScopeTypeException The ScopeType in scope contains
        * an illegal value.
        * @raises IllegalScopeLevelException The scope level is negative
        * (<0).
        * @raises IllegalFilterFormatException The filter string is
        * malformed.
        * @raises FilterComplexityLimit if the filter syntax is correct,
        *   but the filter is too complex to be processed by the IRP agent.
        * @see SearchControl
        * @see BasicCmInformationIteratorIterator
        */
       BasicCmInformationIterator Iterator find_managed_objects(in DN
baseObject,
                                    in SearchControl searchControl,
                                    in AttributeNameSet requestedAttributes)
          raises (UndefinedMOException,
                  IllegalDNFormatException,
                  UndefinedScopeException,
                  IllegalScopeTypeException,
                  IllegalScopeLevelException,
                  IllegalFilterFormatException,
                  FilterComplexityLimit);
```

```
    };
};
#endif
```

CR-Form-v3

CHANGE REQUEST

⌘ **32.106-6** **CR** **005** ⌘ rev **-** ⌘ Current version: **3.0.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: ⌘ (U)SIM ☐ ME/UE ☐ Radio Access Network **X** Core Network **X**

Title:	⌘	**Removing nested IDL modules**
Source:	⌘	SA5
Work item code:	⌘	OAM-CM
Date:	⌘	02/03/2001
Category:	⌘	**F**
Release:	⌘	**R99**

Use one of the following categories:
F *(essential correction)*
A *(corresponds to a correction in an earlier release)*
B *(Addition of feature),*
C *(Functional modification of feature)*
D *(Editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:
2 (GSM Phase 2)
R96 (Release 1996)
R97 (Release 1997)
R98 (Release 1998)
R99 (Release 1999)
REL-4 (Release 4)
REL-5 (Release 5)

Reason for change: ⌘ Annex B, `NotificationDefs`, and Annex C, `NRMDefinitions`, feature nested module definitions.

There are areas of 32.106-6 text that incorrectly only reference one of the module names (as an example, in Table 6, `NotificationDefs::NotificationCommon::MANAGED_OBJECTCLASS` is used instead of `BasicCmIRPSystem::NotificationDefs::NotificationCommon::MANAGED_OBJECTCLASS`). If this CR is not applied, then these references need to be updated via a different CR.

To make the module names more recognizable from Basic CM IRP, change `NotificationDefs` name to `BasicCmNotifiDefs` and change `NRMDefinitions` name to `BasicCmNRMDefs`.

Summary of change: ⌘
1. Remove the outside `BasicCmIRPSystem` modules in Annex B and Annex C.
2. Change name of `NotificationDefs` to `BasicCmNotifDefs`.
3. Update references to `NotificationDefs` constants to `BasicCmNotifDefs`.
4. Change name of `NRMDefinitions` to `BasicCmNRMDefs`.
5. Change `Modeldefs_idl` `#ifndef` constant to match the `BasicCmNRMDefs` module name.

Consequences if not approved: ⌘ Programmers must type in longer names to reference constants and definitions in the IDL, with no clarification benefit. There will be incorrect constant references in text.
If this CR is not applied, then the references mentioned in the 2nd paragraph of

	the “Reason for change” need to be updated via a different CR.

Clauses affected:	⌘	6.4, Annex B, Annex C		
Other specs affected:	⌘	☐ Other core specifications ☐ Test specifications ☐ O&M Specifications	⌘	
Other comments:	⌘			

6.4 Notification attribute mapping

The IS part of Basic CM IRP: IM (see 3GPP TS 32.106-5 [4]) also qualifies the attributes. Tables 6, 7, 8 and 9 show the mapping of these IS attributes to SS equivalents.

Table 6: Mapping from IS `Notification Header` attributes to SS equivalent

IS Attribute of `Notification Header` in 3GPP TS 32.106-5 [4]	SS Attribute	Qualifier
managedObjectClass	~~NotificationDefs~~BasicCmNotifDefs::NotificationCommon::MANAGED_OBJECTCLASS	M
managedObjectInstance	~~NotificationDefs~~BasicCmNotifDefs::NotificationCommon::MANAGED_OBJECT_INSTANCE	M
notificationId	~~NotificationDefs~~BasicCmNotifDefs::NotificationCommon::NOTIFICATION_ID	O
eventTime	~~NotificationDefs~~BasicCmNotifDefs::NotificationCommon::EVENT_TIME	M
systemDN	~~NotificationDefs~~BasicCmNotifDefs::NotificationCommon::SYSTEM_DN	O
eventType	header.fixed_header.event_type.type_name	M
extendedEventType	header.fixed_header.event_name - (always contains an empty string)	M

Table 7: Mapping from IS `notifyObjectCreation` attributes to SS equivalent OBJECT_CREATION

IS Attribute of `notifyObjectCreation` in 3GPP TS 32.106-5 [4]	SS Attribute	Qualifier
notificationHeader	See Table 6	M
correlatedNotifications	~~NotificationDefs~~BasicCmNotifDefs::MOCreation::CORRELATED_NOTIFICATIONS	O
additionalText	~~NotificationDefs~~BasicCmNotifDefs::MOCreation::ADDITIONAL_TEXT	O
sourceIndicator	~~NotificationDefs~~BasicCmNotifDefs::MOCreation::SOURCE_INDICATOR	O
attributeList	remainder_of_body	O

Table 8: Mapping from IS `notifyObjectDeletion` attributes to SS equivalent OBJECT_DELETION

IS Attribute of `notifyObjectDeletion` in 3GPP TS 32.106-5 [4]	SS Attribute	Qualifier
notificationHeader	See Table 6	M
correlatedNotifications	~~NotificationDefs~~BasicCmNotifDefs::MODeletion::CORRELATED_NOTIFICATIONS	O
additionalText	~~NotificationDefs~~BasicCmNotifDefs::MODeletion::ADDITIONAL_ TEXT	O
sourceIndicator	~~NotificationDefs~~BasicCmNotifDefs::MODeletion::SOURCE_INDICATOR	O
attributeList	remainder_of_body (a field of the StructuredEvent)	O

Table 9: Mapping from IS notifyAttributeValueChange attributes to SS equivalent ATTRIBUTE_VALUE_CHANGE

IS Attribute of notifyAttributeValueChange in 3GPP TS 32.106-5 [4]	SS Attribute	Qualifier
notificationHeader	See Table 6	M
correlatedNotifications	~~NotificationDefs~~BasicCmNotifDefs::AttributeValueChange::CORRELATED_NOTIFICATIONS	O
additionalText	~~NotificationDefs~~BasicCmNotifDefs::AttributeValueChange::ADDITIONAL_TEXT	M
sourceIndicator	~~NotificationDefs~~BasicCmNotifDefs::AttributeValueChange::SOURCE_INDICATOR	O
attributeValueChangeDefinition	remainder_of_body	M

Annex B (normative): CORBA IDL, Notification Definitions

```
#ifndef NotificationDefsBasicCmNotifDefs_idl
#define NotificationDefsBasicCmNotifDefs_idl

#pragma prefix "3gppsa5.org"

#include <TimeBase.idl>          // CORBA Time Service
#include <NotificationIRPConstDefs.idl>

module BasicCmIRPSystem
{


module NotificationDefsBasicCmNotifDefs
   {

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


      };

   };
};


#endif
```

Annex C (normative): CORBA IDL, NRM Definitions

```
#ifndef ModelDefsBasicCmNRMDefs_idl
#define ModelDefsBasicCmNRMDefs_idl

#pragma prefix "3gppsa5.org"


module BasicCmIRPSystem
{

   /**
    * This module defines constants for each MO class name and
    * the attribute names for each defined MO class.
    */
   module NRMDefinitionsBasicCmNRMDefs
   {

..
..

      /**
       *  Definitions for MO class GmscFunction
       */
      interface GmscFunction
      {
         const string CLASS = "GmscFunction";

         // Attribute Names
         //
         const string gmscFunctionId = "gmscFunctionId";
         const string userLabel = "userLabel";
      };


   };

};

#endif
```

CR-Form-v3

CHANGE REQUEST

⌘ **32.106-6** **CR** **006** ⌘ rev **-** ⌘ Current version: **3.0.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: ⌘ (U)SIM ☐ ME/UE ☐ Radio Access Network **X** Core Network **X**

Title:	⌘	**Update Structured Event table to be consistent with Alarm IRP**
Source:	⌘	SA5
Work item code:	⌘	OAM-CM
Date:	⌘	02/03/2001
Category:	⌘	**F**
Release:	⌘	**R99**

Use one of the following categories:
F *(essential correction)*
A *(corresponds to a correction in an earlier release)*
B *(Addition of feature),*
C *(Functional modification of feature)*
D *(Editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:

2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
REL-4	*(Release 4)*
REL-5	*(Release 5)*

Reason for change:	⌘	Table 33 in 32.106-6 defines the use of OMG Structured Event. However, it doesn't go to the same amount of detail as in 32.111-3 V3.3.0 to describe attribute ordering and pointing to name constants.
Summary of change:	⌘	Update Table 33 to include additional information as in 32.111-3 V3.3.0.
Consequences if not approved:	⌘	There is the potential of different parameter ordering and different string usages with notifications from different vendors.

Clauses affected:	⌘	7
Other specs affected:	⌘	☐ Other core specifications ⌘ ☐ Test specifications ☐ O&M Specifications
Other comments:	⌘	In case the CR in S5-010049rev2 is not approved, the module references to `BasicCmNotifDefs` and `BasicCmNRMDefs` in this CR have to be updated. (This CR assumes the `BasicCmNotifDefs` and `BasicCmNRMDefs` module names from the CR in S5-010049rev2).

7 Use of OMG Structured Event

Table 33: Use of OMG Structured Event

SS Attribute~~OMG CORBA Structured Event attribute~~	OMG CORBA Structured Event attribute	Comment
There is no corresponding SS attribute~~domain_name~~	domain_name	It contains the version of the supported SS version.~~,~~ This version is defined by constant ~~VERSION, see Annex A (normative): CORBA IDL, Access Protocol~~NotificationIRPConstDefs::CONFIGURATION_IRP_VERSION_1_1 defined in 3GPP TS 32.106-3 [9]
Event Type~~type_name~~	type_name	It is an attribute of notificationHeader. It shall indicate one of the following ITU-T defined semantics: Object Creation, Object Deletion and Attribute Value Change. ~~OBJECT_CREATION~~ ~~OBJECT_DELETION~~ ~~ATTRIBUTE_VALUE_CHANGE~~ It is a string. ~~It is assumed that the types are defined in Annex B (normative): CORBA IDL, Notification Definitions~~It~~'~~s value is either defined by BasicCmNotifDefs::MOCreation::EVENT_TYPE, ~~BasicCmIRPSystem::NotificationDefs~~BasicCmNotifDefs::MODeletion::EVENT_TYPE~~,~~ or ~~BasicCmIRPSystem::NotificationDefs::MOCreation::EVENT_TYPE or BasicCmIRPSystem::NotificationDefs~~BasicCmNotifDefs::AttributeValueChange::EVENT_TYPE
Extended Event Type~~event_name~~	event_name	~~Not used.~~ Shall be set to an empty string~~.~~
There is no corresponding SS attribute~~variable Header~~	variable Header	~~Not used~~
Managed Object Class, Managed Object Instance	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, BasicCmNotifDefs::<interface>::MANAGED_OBJECT_INSTANCE where <interface> is either MOCreation, MODeletion or AttributeValueChange. 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 32.106-3 [9]).
Notification Id	One NV pair of filterable_body_fields	It is an attribute of notificationHeader. Name of NV pair is a string, BasicCmNotifDefs::<interface>::NOTIFICATION_ID where <interface> is either MOCreation, MODeletion or AttributeValueChange. Value of NV pair is a long. See corresponding table in Notification IRP: CORBA SS (3GPP TS 32.106-3 [9]).
Event Time	One NV pair of filterable_body_fields	It is an attribute of notificationHeader. Name of NV pair is a string, BasicCmNotifDefs::<interface>::EVENT_TIME where <interface> is either MOCreation, MODeletion or AttributeValueChange. Value of NV pair is a CommonIRPConstDefs::IRPTime defined in 3GPP TS 32.106-3 [9]. See corresponding table in Notification IRP: CORBA SS (3GPP TS 32.106-3 [9]).
System DN	One NV pair of filterable_body_fields	It is an attribute of notificationHeader. Name of NV pair is a string, BasicCmNotifDefs::<interface>::SYSTEM_DN where <interface> is either MOCreation, MODeletion or AttributeValueChange. Value of NV pair is a string. See corresponding table in Notification IRP: CORBA SS [9].
Correlated Notifications	One NV pair of filterable_body_fields	It is an attribute of the Object Creation, Object Deletion and Attribute Value Change notifications. Name of NV pair is a string,

		BasicCmNotifDefs::<interface>::CORRELATED_NOTIFICATIONS where <interface> is either MOCreation, MODeletion or AttributeValueChange. Value of NV pair is a NotificationIRPConstDefs::CorrelatedNotificationSetType defined in 3GPP TS 32.106-3 [9].
Additional Text	One NV pair of filterable_body_fields	It is an attribute of the Object Creation, Object Deletion and Attribute Value Change notifications. Name of NV pair is a string, BasicCmNotifDefs::<interface>::ADDITIONAL_TEXT where <interface> is either MOCreation, MODeletion or AttributeValueChange. Value of NV pair is a string.
Source Indicator	One NV pair of filterable_body_fields	It is an attribute of the Object Creation, Object Deletion and Attribute Value Change notifications. Name of NV pair is a string, BasicCmNotifDefs::<interface>::SOURCE_INDICATOR where <interface> is either MOCreation, MODeletion or AttributeValueChange. Value of NV pair is a string with values of either BasicCmNotifDefs::<interface>::RESOURCE_OPERATION, BasicCmNotifDefs::<interface>::MANAGEMENT_OPERATION or BasicCmNotifDefs::<interface>::UNKNOWN_OPERATION where <interface> is either MODeletion, MOCreation or AttributeValueChange.
There is no corresponding SS attribute~~remainder_of_body~~		Is used to transport attribute information. For Object Creation notification, this is defined by BasicCmNotifDefs::MOCreation::InitialAttributeValues. For Object Deletion notification, this is defined by BasicCmNotifDefs::MODeletion::AttributeValues. For Attribute Value Change notification, this is defined by BasicCmNotifDefs::AttributeValueChange::ModifiedAttributeSet. The name component of InitialAttributeValues, AttributeValues and ModifiedAttributeSet will be set to attribute names defined in BasicCmNRMDefs.~~, see: Annex B (normative): CORBA IDL, Notification Definitions and subclause 6.4 Notification attribute mapping~~

3GPP TSG SA WG5 Meeting #18 **S5-010134**
Versailles, France, 26 Feb.-2 March. 2001 **S5C010077**

CR-Form-v3

CHANGE REQUEST

⌘ **32.106-6** **CR** **007** ⌘ rev **-** ⌘ Current version: **3.0.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: ⌘ (U)SIM ☐ ME/UE ☐ Radio Access Network **X** Core Network ☐

Title:	⌘	**UMTS Network Resource Model alignment with TSG RAN specifications**
Source:	⌘	SA5
Work item code:	⌘	OAM-CM
Date:	⌘	02/03/2001
Category:	⌘	**F**
Release:	⌘	R99

Use one of the following categories:
F *(essential correction)*
A *(corresponds to a correction in an earlier release)*
B *(Addition of feature),*
C *(Functional modification of feature)*
D *(Editorial modification)*
Detailed explanations of the above categories can be found in 3GPP TR 21.900.

Use one of the following releases:
2 (GSM Phase 2)
R96 (Release 1996)
R97 (Release 1997)
R98 (Release 1998)
R99 (Release 1999)
REL-4 (Release 4)
REL-5 (Release 5)

Reason for change:	⌘	Align 32.106-6 as a consequence of the changes proposed for 32.106-5 CR 001 in S5-010133.
Summary of change:	⌘	The association between NodeBFunction and UtranCell (AssociatedWith-2) is removed. The association AssociatedWith-1 is renamed to AssociatedWith.
Consequences if not approved:	⌘	There is a compatibility problem in the future, as the model does not show the real architecture in the RAN3 specifications. (see further details in S5-010133).

Clauses affected:	⌘	6.5.2.2, 6.5.2.3, 6.5.2.4, Annex C
Other specs affected:	⌘	☐ Other core specifications ⌘ ☐ Test specifications ☐ O&M Specifications 32.106-5 (CR 001 in S5-010133)
Other comments:	⌘	This CR should only be approved and implemented together with the CR in S5-010133 (and vice versa).

6.5.2.2 MOC UtranCell

Table 19: Mapping from NRM MOC UtranCell attributes and associations to SS equivalent MOC UtranCell attributes

NRM Associations/Attributes of MOC UtranCell in 3GPP TS 32.106-5 [4]	SS Attributes	SS Type	Qualifier
utranCellId	utranCellId	string	Read-Only, M
userLabel	userLabel	string	Read-Only, M
AssociatedWith/ utranCell-IubLink	utranCellIubLink	BasicCmIRPSystem::AttributeTypes::MOReference	Read-Only, M

6.5.2.3 MOC NodeBFunction

Table 20: Mapping from NRM MOC NodeBFunction attributes and associations to SS equivalent MOC NodeBFunction attributes

NRM Associations/Attributes of MOC NodeBFunction in 3GPP TS 32.106-5 [4]	SS Attributes	SS Type	Qualifier
nodeBFunctionId	nodeBFunctionId	string	Read-Only, M
userLabel	userLabel	string	Read-Only, M
ConnectedTo/ nodeBFunction-IubLink	nodeBFunctionIubLink	BasicCmIRPSystem::AttributeTypes::MOReference	Read-Only, M

6.5.2.4 MOC IubLink

Table 21: Mapping from NRM MOC IubLink attributes and associations to SS equivalent MOC IubLink attributes

NRM Associations/Attributes of MOC IubLink in 3GPP TS 32.106-5 [4]	SS Attributes	SS Type	Qualifier
iubLinkId	iubLinkId	string	Read-Only, M
userLabel	userLabel	string	Read-Only, M
AssociatedWith/ iubLink-UtranCell	iubLinkUtranCell	BasicCmIRPSystem::AttributeTypes::MOReferenceSet	Read-Only, O
ConnectedTo/ iubLink-NodeBFunction	iubLinkNodeBFunction	BasicCmIRPSystem::AttributeTypes::MOReference	Read-Only, M

Annex C (normative): CORBA IDL, NRM Definitions

```
..
..
      /**
       *  Definitions for MO class UtranCell
       */
      interface UtranCell
      {
         const string CLASS = "UtranCell";

         // Attribute Names
         //
         const string utranCellId = "utranCellId";
         const string userLabel = "userLabel";

         const string utranCellIubLink = "utranCellIubLink";
      };



      /**
       *  Definitions for MO class NodeBFunction
       */
      interface NodeBFunction
      {
         const string CLASS = "NodeBFunction";

         // Attribute Names
         //
         const string nodeBFunctionId = "nodeBFunctionId";
         const string userLabel = "userLabel";
         const string nodeBFunctionIubLink = "nodeBFunctionIubLink";

      };
```