Technical Specification Group Services and System Aspects **_TSGS#25(04)0562_**
Meeting #25, Palm Springs, CA, USA, 13 - 16 September 2004

Source: **SA5 (Telecom Management)**

Title: **Rel-6 CR 32.303 Update 32.303 using IDL Style Guide**

Document for: **Approval**

Agenda Item: **7.5.3**

Doc-1st-Level	Doc-2nd-Level	Spec	CR	Rev	Phase	Subject	Cat	Ver-Cur	Wi
SP-040562	S5-046859	32.303	012	--	Rel-6	Update 32.303 using IDL Style Guide	F	6.0.0	OAM-NIM

3GPP TSG-SA5 (Telecom Management) **S5-046859**
Meeting #39, Montreal, CANADA, Aug 16ñ20, 2004

CR-Form-v7

CHANGE REQUEST

⌘	**32.303**	**CR**	**012**	⌘**rev**	**-**	⌘ Current version:	**6.0.0** ⌘

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

Proposed change affects: UICC apps⌘ □ ME □ Radio Access Network ☒ Core Network ☒

Title:	⌘	Update 32.303 using IDL Style Guide	
Source:	⌘	**SA5** (edwin.tse@ericsson.com)	
Work item code:	⌘	OAM-NIM	***Date:*** ⌘ 20/08/2004
Category:	⌘	**F**	***Release:*** ⌘ Rel-6

Use one of the following categories:
F *(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)*
Rel-6	*(Release 6)*

Reason for change:	⌘	Update Notification IRP CORBA SS so that other Yyy Interface IRP CORBA IDL(s) can re-use its constructs.
Summary of change:	⌘	Correction of TS number in clause Reference. Correct title of clause 5.3. Add a new A.5 in Annex A.
Consequences if not approved:	⌘	IDL statements of Notification IRP CORBA SS will not be using IDL Style Guide.

Clauses affected:	⌘	3, 5.3, Annex A (including a new A.5)			
		Y	N		
Other specs affected:	⌘		X	Other core specifications	⌘
			X	Test specifications	
			X	O&M Specifications	
Other comments:	⌘				

How to create CRs using this form:

Change in Clause 3.1

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in 3GPP TS 32.101 [7], 3GPP TS 32.102 [8] and 3GPP TS 32.301 [9] and the following apply:

IRP document version number string (or "IRPVersion"): See 3GPP TS 32.311 [11].

End of change in Clause 3.1

Change in Clause 5.3

5.3 Notification p~~P~~arameter mapping

Notification IRP: IS (3GPP TS 32.302 [5]) defines the semantics of common attributes carried in notifications. This SS does not provide the mapping of these attributes to their CORBA SS equivalents. Other IRPs such as Alarm IRP: IS (3GPP TS 32.111-2 [6]) identify and qualify these common attributes for use in their environment. Their corresponding SS documents define the mapping of these attributes to their SS equivalents.

End of change in clause 5.3

Annex A (normative): IDL specifications

A.1 IDL specification (file name "ManagedGenericIRPConstDefs.idl")

```
//File: ManagedGenericIRPConstDefs.idl

#ifndef ~~ManagedGenericIRPConstDefs_idl~~_MANAGEDGENERICIRPCONSTDEFS_IDL_
#define ~~ManagedGenericIRPConstDefs_idl~~_MANAGEDGENERICIRPCONSTDEFS_IDL_

#include "TimeBase.idl"

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

/* ## Module: ManagedGenericIRPConstDefs
This module contains definitions commonly used among all IRPs such as Alarm IRP.
================================================================
*/
module ManagedGenericIRPConstDefs
{
   /*
   Definition imported from CosTime.
   The time refers to time in Greenwich Time Zone.
   It also consists of a time displacement factor in the form of minutes of
   displacement from the Greenwich Meridian.
   */
   typedef TimeBase::UtcT IRPTime;

   enum Signal {~~OK, Failure, PartialFailure~~OK, FAILURE, PARTIALFAILURE};

   /*
   The VersionNumber is a string that identifies the IRP specification name
   and its version number.  See definition "IRP document version number
   string" or "IRPVersion".

   The VersionNumberSet is a sequenece of such VersionNumber.  It is returned
   by get_XXX_IRP_versions().  The sequence order has no significance.
   */
   typedef string VersionNumber;
   typedef sequence <VersionNumber> VersionNumberSet;

   typedef string MethodName;
   typedef string ParameterName;
   typedef sequence <ParameterName> ParameterList;

   /*
   The Method defines the structure to be returned as part of
   get_supported_operations_profile().  The name shall be the actual method
   name (ex. "attach_push", "change_subscription_filter", etc.)
   The parameter_list contains a list of strings.  Each string shall be
   the actual parameter name (ex. "manager_reference", "filter", etc.)
   */
   struct Method
   {
      MethodName name;
      ParameterList parameter_list;
   };

   /*
   List of all methods and their associated parameters.
   */
   typedef sequence <Method> MethodList;

   /*
   StringTypeOpt is a type carrying an optional parameter.
   If the boolean is TRUE, then the value is present.
   Otherwise the value is absent.
   */
   union StringTypeOpt switch (boolean)
   {
      case TRUE: string value;
   };

   /*
   ShortTypeOpt is a type carrying an optional parameter.
   If the boolean is TRUE, then the value is present.
   Otherwise the value is absent.
   */
   union ShortTypeOpt switch (boolean)
   {
      case TRUE: short value;
   };

   /*
   UnsignedShortTypeOpt is a type carrying an optional parameter.
   If the boolean is TRUE, then the value is present.
   Otherwise the value is absent.
   */
   union UnsignedShortTypeOpt switch (boolean)
   {
      case TRUE: unsigned short value;
   };

   /*
   LongTypeOpt is a type carrying an optional parameter.
   If the boolean is TRUE, then the value is present.
   Otherwise the value is absent.
   */
   union LongTypeOpt switch (boolean)
   {
      case TRUE: long value;
   };

   /*
   UnsignedLongTypeOpt is a type carrying an optional parameter.
   If the boolean is TRUE, then the value is present.
   Otherwise the value is absent.
   */
   union UnsignedLongTypeOpt switch (boolean)
   {
      case TRUE: unsigned long value;
   };
};

#endif  // _MANAGEDGENERICIRPCONSTDEFS_IDL_
```

A.2 IDL specification (file name "ManagedGenericIRPSystem.idl")

```
//File: ManagedGenericIRPSystem.idl

#ifndef ~~ManagedGenericIRPSystem_idl~~_MANAGEDGENERICIRPSYSTEM_IDL_
#define ~~ManagedGenericIRPSystem_idl~~_MANAGEDGENERICIRPSYSTEM_IDL_

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

/* ## Module: ManagedGenericIRPSystem
This module contains definitions commonly used among all IRPs such as Alarm IRP.
================================================================
*/
module ManagedGenericIRPSystem
{
   /*
   Exception thrown when an unsupported optional parameter
   is passed with information.
   The parameter shall be the actual unsupported parameter name.
   */
   exception ParameterNotSupported { string parameter; };

   /*
   Exception thrown when an invalid parameter value is passed.
   The parameter shall be the actual parameter name.
   */
   exception InvalidParameter { string parameter; };

   /*
   Exception thrown when a valid but unsupported parameter value is passed.
   The parameter shall be the actual parameter name.
   */
   exception ValueNotSupported { string parameter; };

   /*
   Exception thrown when an unsupported optional method is called.
   */
   exception OperationNotSupported {};
};

#endif   // _MANAGEDGENERICIRPSYSTEM_IDL_
```

A.3 IDL specification (file name "NotificationIRPConstDefs.idl")

```
//File: NotificationIRPConstDefs.idl

#ifndef ~~NotificationIRPConstDefs_idl~~_NOTIFICATIONIRPCONSTDEFS_IDL_
#define ~~NotificationIRPConstDefs_idl~~_NOTIFICATIONIRPCONSTDEFS_IDL_

#include "ManagedGenericIRPConstDefs.idl"

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


/* ## Module: NotificationIRPConstDefs
This module contains definitions specific for Notification IRP.
================================================================
*/
module NotificationIRPConstDefs
{


   /*
   Define the parameters (in the notification header) specified in
   the Notification IRP: IS.
   */
   interface AttributeNameValue
   {
      const string NOTIFICATION_ID = "a";
      const string EVENT_TIME = "b";
      const string SYSTEM_DN = "c";
      const string MANAGED_OBJECT_CLASS = "d";
      const string MANAGED_OBJECT_INSTANCE = "e";
   };

   /*
   It defines the notification categories.
   A notification category is identified by the IRP name and its version number.
   */
   typedef ManagedGenericIRPConstDefs::VersionNumberSet NotificationCategorySet;

   /*
   It defines the notification types of a particular notification category.
   */
   typedef sequence <string> NotificationTypePerNotificationCategory;

   /*
   This sequence identifies all notification types of all notification
   categories identified by NotificationCategorySet.  The number of elements
   in this sequence shall be identical to that of NotificationCategorySet.
   */
   typedef sequence <NotificationTypePerNotificationCategory>
```

```
       NotificationTypesSet;

   /*
   It defines a sequence of SubscriptionIds.
   */
   typedef string SubscriptionId;
   typedef sequence <SubscriptionId> SubscriptionIdSet;

   /*
   This indicates if the subscription is Active (not suspended), Suspended,
   or Invalid.
   */
   enum SubscriptionState {~~Active, Suspended, Invalid~~ACTIVE, SUSPENDED, INVALID};
};

#endif  // _NOTIFICATIONIRPCONSTDEFS_IDL_
```

A.4 IDL specification (file name "NotificationIRPSystem.idl")

```
//File: NotificationIRPSystem.idl

#ifndef ~~NotificationIRPSystem_idl~~_NOTIFICATIONIRPSYSTEM_IDL_
#define ~~NotificationIRPSystem_idl~~_NOTIFICATIONIRPSYSTEM_IDL_

#include "CosNotifyChannelAdmin.idl"
#include "ManagedGenericIRPConstDefs.idl"
#include "ManagedGenericIRPSystem.idl"
#include "NotificationIRPConstDefs.idl"

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

/* ## Module: NotificationIRPSystem
This module implements capabilities of Notification IRP.
================================================================
*/
module NotificationIRPSystem
{
   /*
   System fails to complete the operation.  System can provide reason
   to qualify the exception.  The semantics carried in reason
   is outside the scope of this IRP.
   */
   exception GetNotificationIRPVersions { string reason; };
   exception GetNotificationIRPOperationsProfile { string reason; };
   exception GetNotificationIRPNotificationProfile { string reason; };
   exception Attach { string reason; };
   exception DetachException { string reason; };
   exception GetSubscriptionStatus { string reason; };
   exception ChangeSubscriptionFilter { string reason; };
   exception GetNotificationCategories { string reason; };

   exception GetSubscriptionIds { string reason; };

   exception AlreadySubscribed {};
   exception AtLeastOneNotificationCategoryNotSupported {};

   interface NotificationIRP
   {
      /*
      Return the list of all supported Notification IRP versions
      Each IRPVersion is defined by the rule in TS 32.311 clause titled
      "IRP document version number string"
      */
      ManagedGenericIRPConstDefs::VersionNumberSet get_notification_IRP_versions
      (
      )
      raises (GetNotificationIRPVersions);

      /*
      Return the list of all supported operations and their supported
      parameters for a specific Notification IRP version.
      */
      ManagedGenericIRPConstDefs::MethodList
          get_notification_IRP_operations_profile (
              in ManagedGenericIRPConstDefs::VersionNumber
                  notification_irp_version
      )
      raises (GetNotificationIRPOperationsProfile,
              ManagedGenericIRPSystem::OperationNotSupported,
              ManagedGenericIRPSystem::InvalidParameter);

      /*
      Return the list of all supported notifications.
      Agent should always throw a ManagedGenericIRPSystem::OperationNotSupported
      exception.
      Similar method, such as get_alarm_IRP_notification_profile,
      is supported in other IRP versions such as Alarm IRP.
      */
      ManagedGenericIRPConstDefs::MethodList
          get_notification_IRP_notification_profile (
              in ManagedGenericIRPConstDefs::VersionNumber
                  notification_irp_version
      )
      raises (GetNotificationIRPNotificationProfile,
              ManagedGenericIRPSystem::OperationNotSupported,
              ManagedGenericIRPSystem::InvalidParameter);

      /*
      Obtain the list of all supported notification categories.
      */
      NotificationIRPConstDefs::NotificationCategorySet
          get_notification_categories (
              out NotificationIRPConstDefs::NotificationTypesSet
                  notification_type_list
      )
      raises (GetNotificationCategories,
          ManagedGenericIRPSystem::OperationNotSupported);

      NotificationIRPConstDefs::SubscriptionId attach_push (
         in string manager_reference,
         in unsigned long time_tick,
         in NotificationIRPConstDefs::NotificationCategorySet
             notification_categories,
         in ManagedGenericIRPConstDefs::StringTypeOpt filter
      )
      raises (Attach, ManagedGenericIRPSystem::ParameterNotSupported,
              ManagedGenericIRPSystem::InvalidParameter, AlreadySubscribed,
              AtLeastOneNotificationCategoryNotSupported);

      NotificationIRPConstDefs::SubscriptionId attach_push_b (
         in string manager_reference,
         in unsigned long time_tick,
         in NotificationIRPConstDefs::NotificationCategorySet
             notification_categories,
         in ManagedGenericIRPConstDefs::StringTypeOpt filter,
         out CosNotifyChannelAdmin::SequenceProxyPushSupplier system_reference
      )
      raises (Attach, ManagedGenericIRPSystem::OperationNotSupported,
              ManagedGenericIRPSystem::ParameterNotSupported,
              ManagedGenericIRPSystem::InvalidParameter,
              AlreadySubscribed, AtLeastOneNotificationCategoryNotSupported);

      NotificationIRPConstDefs::SubscriptionId attach_pull (
         in string manager_reference,
         in unsigned long time_tick,
         in NotificationIRPConstDefs::NotificationCategorySet
             notification_categories,
         in ManagedGenericIRPConstDefs::StringTypeOpt filter,
         out CosNotifyChannelAdmin::SequenceProxyPullSupplier system_reference
      )
      raises (Attach, ManagedGenericIRPSystem::OperationNotSupported,
              ManagedGenericIRPSystem::ParameterNotSupported,
              ManagedGenericIRPSystem::InvalidParameter,
              AlreadySubscribed, AtLeastOneNotificationCategoryNotSupported);

      /*
      Replace the present filter constraint with the one provided.
      */
      void change_subscription_filter (
         in NotificationIRPConstDefs::SubscriptionId subscription_id,
         in string filter
      )
      raises (ChangeSubscriptionFilter,
          ManagedGenericIRPSystem::OperationNotSupported,
          ManagedGenericIRPSystem::InvalidParameter);

      /*
      Check the current state of the subscription.
      */
      NotificationIRPConstDefs::NotificationCategorySet get_subscription_status
      (
         in NotificationIRPConstDefs::SubscriptionId subscription_id,
         out ManagedGenericIRPConstDefs::StringTypeOpt filter_in_effect,
         out NotificationIRPConstDefs::SubscriptionState subscription_state,
         out unsigned long time_tick
      )
      raises (GetSubscriptionStatus,
          ManagedGenericIRPSystem::OperationNotSupported,
          ManagedGenericIRPSystem::InvalidParameter);

      NotificationIRPConstDefs::SubscriptionIdSet get_subscription_ids (
         in string manager_reference
      )
      raises (GetSubscriptionIds,
          ManagedGenericIRPSystem::OperationNotSupported,
          ManagedGenericIRPSystem::InvalidParameter);

      /*
      Terminates the subscription with the agent.
      */
      void detach (
         in string manager_reference,
         in NotificationIRPConstDefs::SubscriptionId subscription_id
      )
      raises (DetachException,
          ManagedGenericIRPSystem::ParameterNotSupported,
          ManagedGenericIRPSystem::InvalidParameter);
   };
};

#endif  // _NOTIFICATIONIRPSYSTEM_IDL_
```

Change in (add a new A.5) Annex A

A.5 IDL specification (file name "NotificationIRPNotifications.idl")

```
//File: NotificationIRPNotifications.idl

#ifndef _NOTIFICATIONIRPNOTIFICATIONS_IDL_
#define _NOTIFICATIONIRPNOTIFICATIONS_IDL_

#include <NotificationIRPConstDefs.idl>

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

module NotificationIRPNotifications
{

    interface Notify
    {
         /**
          Notification IRP IS defines 6 attributes for the notification header.
          They are: objectClass, objectInstance, notificationId, eventTime,
          systemDN and notificationType.

          The first 2 attributes are mapped into 1 name-value pair
          of the filterable body of the CORBA structured event.  The name of
          the mapped IDL construct is MANAGED_OBJECT_INSTANCE.  The const
          string of this mapped IDL construct is defined here.

          The notificationId, eventTime and systemDN are respectively mapped
          into 3 name-value pairs of the filterable body.  The const string(s) of
          these 3 mapped IDL constructs are defined here.

          The notificationType is not mapped into any name-value pair of the
          filterable body but is mapped into the type_name position-dependent
          field of the CORBA structured-event.  There is no need for a const string
          definition for it.
          */

         const string MANAGED_OBJECT_INSTANCE =
             NotificationIRPConstDefs::AttributeNameValue::MANAGED_OBJECT_INSTANCE;

         const string NOTIFICATION_ID =
             NotificationIRPConstDefs::AttributeNameValue::NOTIFICATION_ID;

         const string EVENT_TIME =
             NotificationIRPConstDefs::AttributeNameValue::EVENT_TIME;

         const string SYSTEM_DN =
             NotificationIRPConstDefs::AttributeNameValue::SYSTEM_DN;

    };
};

#endif  // _NOTIFICATIONIRPNOTIFICATIONS_IDL_
```

End of change in Annex A

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
Sep 2001	S_13	SP-010522	001	--	Eliminate guesses on IDL file names in Notification IRP: CORBA SS	4.0.0	4.1.0
Mar 2002	S_15	SP-020038	002	--	Addition of missing generic CORBA exception "ValueNotSupported" into CORBA module "ManagedGenericIRPSystem"	4.1.0	4.2.0
Mar 2002	S_15	--	--	--	Automatic upgrade to Rel-5 (no Rel-5 CR)	4.2.0	5.0.0
Sep 2002	S_17	SP-020482	004	--	Corrections to CORBA IDL specification "NotificationIRPSystem"	5.0.0	5.1.0
Sep 2002	S_17	SP-020479	005	--	Add optional parameters in CORBA Solution Set	5.0.0	5.1.0
Sep 2002	--	--	--	--	Corrected history box CR# & TS version#	5.1.0	5.1.1
Dec 2002	--	--	--	--	Cosmetics	5.1.1	5.1.2
Mar 2003	S_19	SP-030137	008	--	Remove unused suspend_subscription and resume_subscription methods	5.1.2	5.2.0
Mar 2003	S_19	SP-030137	009	--	Corrections of CORBA IDL syntax errors	5.1.2	5.2.0
Mar 2003	S_19	SP-030064	010	--	Update the usage IRP_VERSION in line with adopted release 5 policy - alignment with 32.111-3	5.1.2	5.2.0
Mar 2004	S_23	SP-040105	--	--	Automatic upgrade to Rel-6 (no CR)	5.2.0	6.0.0