Technical Specification Group Services and System Aspects
Meeting #10, Bangkok, Thailand, 11-14 December 2000

TSGS#10(00)0519

Source: **SA WG5**

Title: **CRs to Telecommunication Management; Configuration Management; Part 3: Notification Integration Reference Point; CORBA Solution Set version 1:1 (32.106-3)**

Document for: **Approval**

Agenda Item: **7.5.3**

Doc-1st-Level	Doc-2nd-Level	Spec	CR	Rev	Phase	Cat	Subject	Version-Current	Version-New	Work item
SP-000519	S5-000486	32.106-3	001	1	R99	F	Add pragma statement to Notification IRP IDL	3.1.0	3.2.0	OAM-CM
SP-000519	S5-000482	32.106-3	002		R99	F	Correction of IDL Errors	3.1.0	3.2.0	OAM-CM
SP-000519	S5-000483	32.106-3	003		R99	F	Spelling Errors in the CORBA IDL	3.1.0	3.2.0	OAM-CM
SP-000519	S5-000485	32.106-3	004		R99	F	Ensure consistency with IDL exception	3.1.0	3.2.0	OAM-CM

3GPP TSG-SA5 (Telecom Management)
Meeting #15, Girdwood, Alaska, USA, 16 - 20 October 2000

SA5#15(00)0486
Tdoc S5C000158

CHANGE REQUEST

Please see embedded help file at the bottom of this page for instructions on how to fill in this form correctly.

32-106-3	**CR**	**001R1**	Current Version:	V3.1.0

GSM (AA.BB) or 3G (AA.BBB) specification number ↑ *↑ CR number as allocated by MCC support team*

For submission to: **SA#10** *list expected approval meeting # here ↑*	for approval	**X**	strategic		*(for SMG use only)*
	for information		non-strategic		

Form: CR cover sheet, version 2 for 3GPP and SMG *The latest version of this form is available from: ftp://ftp.3gpp.org/Information/CR-Form-v2.doc*

Proposed change affects: *(at least one should be marked with an X)*	(U)SIM		ME		UTRAN / Radio	**X**	Core Network	**X**

Source: **SA5#15** **Date:** 20/10/2000

Subject: Add pragma statement to Notification IRP IDL

Work item: OAM-CM

Category: *(only one category shall be marked with an X)*

F	Correction	**X**	**Release:**	Phase 2	
A	Corresponds to a correction in an earlier release			Release 96	
B	Addition of feature			Release 97	
C	Functional modification of feature			Release 98	
D	Editorial modification			Release 99	**X**
				Release 00	

Reason for change:

In order to avoid potential CORBA name clashes, this CR adds the use of the **"#pragma prefix" directive in Notification IRP CORBA IDL** in annex A.

Lack of 3GPP specified "#pragma prefix" may cause **interworking problems between Agent and Manager when different development tools are used**.

CORBA Common Object Request Broker Architecture
IDL Interface Definition Language
IRP Integration Reference Point

Clauses affected: **Annex A**

Other specs affected:

Other 3G core specifications		→ List of CRs:	
Other GSM core specifications		→ List of CRs:	
MS test specifications		→ List of CRs:	
BSS test specifications		→ List of CRs:	
O&M specifications		→ List of CRs:	

Other comments:

Annex A (normative): Notification IRP CORBA IDL

```
/* ## Module: CommonIRPConstDefs
This module contains definitions commonly used among all IRPs such as Alarm IRP.
=================================================================================
*/

#ifndef CommonIRPConstDefs_idl
#define CommonIRPConstDefs_idl
#include <TimeBase.idl>
#pragma prefix "3gppsa5.org"

module CommonIRPConstDefs {

  /*
  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};

  typedef sequence <string> VersionNumberSet;

};

#endif


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

#ifndef  NotificationIRPConstDefs_idl
#define  NotificationIRPConstDefs_idl
#pragma prefix "3gppsa5.org"

module NotificationIRPConstDefs {

  /*
  This is a string sequence identifying notification categories.
  A notification category is identified by the IRP name and its version.
  */
  typedef sequence <string> NotificationCategorySet;

  /*
  This is a sequence of strings identifying event types of a particular
  notification category.
  */
  typedef sequence <string> EventTypesPerNotificationCategory;

  /*
  This sequence identifies all event types of all notification categories
```

```
  identified by NotificationCategorySet.  The number of elements in this
  sequence shall be identical to that of NotificationCategorySet.
  */
  typedef sequence <EventTypesPerNotificationCategory> EventTypesSet;

  /*
  This is a sequence of strings identifying extended event types of
  a particular notification category.
  */
  typedef sequence <string> ExtendedEventTypePerNotificationCategory;
```

========== end of page 17 ==========

========== page 19 ==========

```
  const string NV_SPECIFIC_PROBLEM = "i";
  const string NV_ADDITIONAL_TEXT = "j";
  const string NV_ALARM_id = "k";
  const string NV_ACK_USER_ID = "l";
  const string NV_ACK_TIME = "m";
  const string NV_ACK_SYSTEM_ID = "n";
  const string NV_ACK_STATE = "o";
  const string NV_BACKED_UP_STATUS = "p";
  const string NV_BACK_UP_OBJECT = "q";
  const string NV_THRESHOLD_INFO = "r";
  const string NV_TREND_INDICATION = "s";
  const string NV_STATE_CHANGE_DEFINITIONS = "t";
  const string NV_MONITORED_ATTRIBUTES = "u";
  const string NV_PROPOSED_REPAIRED_ACTIONS = "v";


  /*
  This indicates if the subscription is active (not suspended) or inactive.
  */
  enum SubscriptionState {Inactive, Active, DontKnow};

};

#endif


/* ## Module: NotificationIRPSystem
       This module implements capabilities of IRPAgent specified in Notification
       IRP: Information Service version 1 and its equivalents in Notification
    IRP: CORBA Solution Set version 1:1.
 ================================================================
*/

#ifndef  NotificationIRPSystem_idl
#define  NotificationIRPSystem_idl

#include "CosNotifyComm.idl"
#include "CosNotifyChannelAdmin.idl"
#include "NotificationIRPConstDefs.idl"
#include "CommonIRPConstDefs.idl"
#pragma prefix "3gppsa5.org"

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 Attach { string reason; };
  exception DetachException { string reason; };

  exception GetSubscriptionStatus { string reason; };
  exception GetSubscriptionIds { string reason; };
  exception ChangeSubscriptionFilter { string reason; };
  exception GetNotificationCategories { string reason; };

  exception ParameterNotSupported { string parameter; };
    // name of the unsupported parameter as defined in IDL
  exception InvalidParameter { string parameter; };
    // name of the parameter as defined in IDL
  exception OperationNotSupported {};
  exception AlreadySubscribed {};
  exception AtLeastOneNotificationCategoryNotSupported {};
```

========================= end of page 19 =====================

3GPP TSG-SA5 (Telecom Management)
Meeting #15, Girdwood, Alaska, USA, 16 - 20 October 2000

SA5#15(00)0482
Tdoc **S5C000152**

CHANGE REQUEST

Please see embedded help file at the bottom of this page for instructions on how to fill in this form correctly.

32.106-3	CR	002	Current Version:	V3.1.0

GSM (AA.BB) or 3G (AA.BBB) specification number ↑ *↑ CR number as allocated by MCC support team*

For submission to:	SA#10	for approval	X	strategic		(for SMG use only)
list expected approval meeting # here ↑		for information		non-strategic		

Form: CR cover sheet, version 2 for 3GPP and SMG *The latest version of this form is available from: ftp://ftp.3gpp.org/Information/CR-Form-v2.doc*

Proposed change affects: *(at least one should be marked with an X)* (U)SIM [ ] ME [ ] UTRAN / Radio [X] Core Network [X]

Source: SA5#15 **Date:** 20/10/2000

Subject: Correction of IDL Errors

Work item: OAM-CM

Category: *(only one category shall be marked with an X)*

F	Correction	X	**Release:**	Phase 2	
A	Corresponds to a correction in an earlier release			Release 96	
B	Addition of feature			Release 97	
C	Functional modification of feature			Release 98	
D	Editorial modification			Release 99	X
				Release 00	

Reason for change:

To correct an IDL listed in Annex A:
`NotificationCategorySet` shall be listed as
`NotificationIRPConstDefs::NotificationCategorySet`.

This correction also changes some of the text in subclause 5.2 (Tables 2, 3, 4) explaining the corrected data item.

In addition, some IDL compilers don't allow type names and variable names to be the same except for an upper case or lower case initial letter.

To correct the usage of `NotifIDSet` and `notifIDSet`,
the `NotifIDSet` is being changed to `NotifIDSetType`.

Clauses affected: 5.2 (Tables 2, 3 & 4) and Annex A (normative)

Other specs affected:

Other 3G core specifications	[ ]	→ List of CRs:
Other GSM core specifications	[ ]	→ List of CRs:
MS test specifications	[ ]	→ List of CRs:
BSS test specifications	[ ]	→ List of CRs:
O&M specifications	[ ]	→ List of CRs:

Other comments:

5.2 Operation parameter mapping

3G TS 32.106-2 [5] defines semantics of parameters carried in operations across the Notification IRP. Table 2 through table 12 indicate the mapping of these parameters, as per operation, to their equivalents defined in this SS.

Table 2: Mapping from IS `subscribe` parameters to SS `attach_push` equivalents

IS Operation parameter	SS Method parameter	Qualifier
managerReference	Object manager_reference	M
timeTick	long time_tick	O
notification Categories	~~NotificationCategorySet~~NotificationIRPConstDefs::NotificationCategorySet notification_category_set	O
filter	string filter (See NOTE)	O
subscriptionId	Return value of type SubscriptionId	M
status	Attach, ParameterNotSupported, InvalidParameter, AlreadySubscribed, AtLeastOneNotificationCategoryNotSupported	M
NOTE: The grammar of the filter string is extended_TCL defined by OMG Notification Service (OMG TC Document telecom [2]). This grammar shall be the only one used for Alarm IRP: CORBA SS.		

Table 3: Mapping from IS `subscribe` parameters to SS `attach_push_b` equivalents

IS Operation parameter	SS Method parameter	Qualifier
managerReference	Object manager_reference	M
timeTick	long time_tick	O
notification Categories	~~NotificationCategorySet~~NotificationIRPConstDefs::NotificationCategorySet notification_category_set	O
filter	string filter	O
subscriptionId	Return value of type SubscriptionId	M
Not specified in IS	CosNotifyChannelAdmin::SequenceProxyPushSupplier system_reference (See NOTE)	M
status	Attach, OperationNotSupported, ParameterNotSupported, InvalidParameter, AlreadySubscribed, AtLeastOneNotificationCategoryNotSupported	M
NOTE: IRPAgent provides this reference to which IRPManager can invoke methods to manage the subscription. Valid methods are not defined in this IRP. OMG CORBA Notification Service defines these methods. Read interface SequencePushSupplier:proxySupplier, CosNotifyComm::SequencePushSupplier{}. IRPManager is expected to invoke connect_sequence_push_consumer() of this interface to connect its own cosNotifyComm::sequencePushConsummer with this reference. After successful connection, IRPAgent pushes sequence of Structured Events towards IRPManager.		

Table 4: Mapping from IS `subscribe` parameters to SS `attach_pull` equivalents

IS Operation parameter	SS Method parameter	Qualifier
managerReference	Object manager_reference	M
timeTick	long time_tick	O
notification Categories	~~NotificationCategorySet~~NotificationIRPConstDefs::NotificationCategorySet notification_category_set	O
filter	string filter	O
subscriptionId	Return value of type SubscriptionId	M
Not specified in IS.	CosNotifyChannelAdmin::SequenceProxyPullSupplier system_reference	M
status	Attach, OperationNotSupported, ParameterNotSupported, InvalidParameter, AlreadySubscribed, AtLeastOneNotificationCategoryNotSupported	M

Annex A (normative): Notification IRP CORBA IDL

```
/* ## Module: CommonIRPConstDefs
This module contains definitions commonly used among all IRPs such as Alarm
IRP.
================================================================================
=
*/

#ifndef CommonIRPConstDefs_idl
#define CommonIRPConstDefs_idl
#include <TimeBase.idl>

module CommonIRPConstDefs {

  /*
  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};

  typedef sequence <string> VersionNumberSet;

};

#endif


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

#ifndef  NotificationIRPConstDefs_idl
#define  NotificationIRPConstDefs_idl

module NotificationIRPConstDefs {

  /*
  This is a string sequence identifying notification categories.
  A notification category is identified by the IRP name and its version.
  */
  typedef sequence <string> NotificationCategorySet;

  /*
  This is a sequence of strings identifying event types of a particular
  notification category.
  */
  typedef sequence <string> EventTypesPerNotificationCategory;

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

  /*
  This is a sequence of strings identifying extended event types of
  a particular notification category.
  */
  typedef sequence <string> ExtendedEventTypePerNotificationCategory;
```

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

    typedef sequence <long> ~~NotifIDSet~~NotifIDSetType;

    /*
    This holds identifiers of notifications that are correlated.
    */

    struct CorrelatedNotification {
         string 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 parameter-
  attribute
                        //  of the Structured Event, i.e., the notification.
         ~~NotifIDSet~~NotifIDSetType notifIDSet;
    };

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

    /*
    This is a sequence of strings identifying Subscription Ids.
    */
    typedef string SubscriptionId;
    typedef sequence <SubscriptionId> SubscriptionIdSet;

    /*
    This block encapsulates valid strings carried in domain_name of
    structured event header.  It carries the name of IRP and its
    corresponding CORBA SS version number.  They are the returned
    values for get_XXX_IRP_version() as well.
    */
    const string ALARM_IRP_VERSION_1_1 = "1f1";  //alarm IRP 1:1
    const string CONFIGURATION_IRP_VERSION_1_1 = "1c1"; //CM IRP 1:1


    /*
    This string is used as return value for get_notification_irp_version()
    */
    const string NOTIFICATION_IRP_VERSION_1_1 = "1n1"; //Notification IRP 1:1


    /*
    This block encapsulates string used in the name of the Name Value
    pair of the structured event.
    */

    const string NV_NOTIFICATION_ID = "a";
    const string NV_CORRELATED_NOTIFICATIONS = "b";
    const string NV_EVENT_TIME = "c";
    const string NV_SYSTEM_DN = "d";
    const string NV_MANAGED_OBJECT_CLASS = "e";
    const string NV_MANAGED_OBJECT_INSTANCE = "f";
    const string NV_PROBABLE_CAUSE = "g";
    const string NV_PERCEIVED_SEVERITY = "h";
    const string NV_SPECIFIC_PROBLEM = "i";
    const string NV_ADDITIONAL_TEXT = "j";
    const string NV_ALARM_id = "k";
```

```
  const string NV_ACK_USER_ID = "l";
  const string NV_ACK_TIME = "m";
  const string NV_ACK_SYSTEM_ID = "n";
  const string NV_ACK_STATE = "o";
  const string NV_BACKED_UP_STATUS = "p";
  const string NV_BACK_UP_OBJECT = "q";
  const string NV_THRESHOLD_INFO = "r";
  const string NV_TREND_INDICATION = "s";
  const string NV_STATE_CHANGE_DEFINITIONS = "t";
  const string NV_MONITORED_ATTRIBUTES = "u";
  const string NV_PROPOSED_REPAIRED_ACTIONS = "v";


  /*
  This indicates if the subscription is active (not suspended) or inactive.
  */
  enum SubscriptionState {Inactive, Active, DontKnow};

};

#endif


/* ## Module: NotificationIRPSystem
       This module implements capabilities of IRPAgent specified in Notification
       IRP: Information Service version 1 and its equivalents in Notification
    IRP: CORBA Solution Set version 1:1.
 ================================================================
*/

#ifndef  NotificationIRPSystem_idl
#define  NotificationIRPSystem_idl

#include "CosNotifyComm.idl"
#include "CosNotifyChannelAdmin.idl"
#include "NotificationIRPConstDefs.idl"
#include "CommonIRPConstDefs.idl"


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 Attach { string reason; };
  exception DetachException { string reason; };

  exception GetSubscriptionStatus { string reason; };
  exception GetSubscriptionIds { string reason; };
  exception ChangeSubscriptionFilter { string reason; };
  exception GetNotificationCategories { string reason; };

  exception ParameterNotSupported { string parameter; };
    // name of the unsupported parameter as defined in IDL
  exception InvalidParameter { string parameter; };
    // name of the parameter as defined in IDL
  exception OperationNotSupported {};
  exception AlreadySubscribed {};
  exception AtLeastOneNotificationCategoryNotSupported {};


     interface NotificationIRPOperations {

    /* ## Operation: attach_push
    */
    NotificationIRPConstDefs::SubscriptionId attach_push (
```

```
      in Object manager_reference,
      in long time_tick,
      in
NotificationCategorySetNotificationIRPConstDefs::NotificationCategorySet
_________notification_category_set,
      in string filter
    )
    raises (Attach, ParameterNotSupported, InvalidParameter, AlreadySubscribed,
            AtLeastOneNotificationCategoryNotSupported);

    /* ## Operation: attach_push_b
    */
          NotificationIRPConstDefs::SubscriptionId attach_push_b (
      in Object manager_reference,
      in long time_tick,
      in
NotificationCategorySetNotificationIRPConstDefs::NotificationCategorySet
_________notification_category_set,
      in string filter,
      out CosNotifyChannelAdmin::SequenceProxyPushSupplier system_reference
    )
    raises
(Attach,OperationNotSupported,ParameterNotSupported,InvalidParameter,AlreadySub
scribed,AtLeastOneNotificationCategoryNotSupported);

    /* ## Operation: attach_pull
    */
          NotificationIRPConstDefs::SubscriptionId attach_pull (
       in Object manager_reference,
       in long time_tick,
       in
NotificationCategorySetNotificationIRPConstDefs::NotificationCategorySet
_________notification_category_set,
       in string filter,
       out CosNotifyChannelAdmin::SequenceProxyPullSupplier system_reference
    )
    raises (Attach, OperationNotSupported, ParameterNotSupported,
            InvalidParameter, AlreadySubscribed,
            AtLeastOneNotificationCategoryNotSupported);

    /* ## Operation: detach
    */
          void detach (
      in Object manager_reference,
      in string subscription_id
    )
    raises (DetachException,InvalidParameter);

    /* ## Operation: get_notification_IRP_version
    */
    CommonIRPConstDefs::VersionNumberSet get_notification_IRP_version ()
    ;

    /* ## Operation: get_subscription_status
    */
          NotificationIRPConstDefs::NotificationCategorySet
get_subscription_status (
      in string subscription_id,
      out string filter_in_effect,
      out NotificationIRPConstDefs::SubscriptionState subscription_state,
      out long time_tick
    )
    raises (GetSubscriptionStatus,OperationNotSupported,InvalidParameter);

    /* ## Operation: get_subscription_ids
    */
    NotificationIRPConstDefs::SubscriptionIdSet get_subscription_ids (
      in Object manager_reference
    )
```

```
    raises (GetSubscriptionIds,OperationNotSupported,InvalidParameter);

    /* ## Operation: change_subscription_filter
    */
          void change_subscription_filter (
               in string subscription_id,
               in string filter
    )
    raises (ChangeSubscriptionFilter,OperationNotSupported,InvalidParameter);

    /* ## Operation: get_notification_categories
    */
    NotificationIRPConstDefs::NotificationCategorySet
         get_notification_categories (
      out NotificationIRPConstDefs::EventTypesSet event_type_list,
      out NotificationIRPConstDefs::ExtendedEventTypesSet
         extended_event_type_list
    )
    raises (GetNotificationCategories,OperationNotSupported);
     };

};

#endif
```

3GPP TSG-SA5 (Telecom Management)
Meeting #15, Girdwood, Alaska, USA, 16 - 20 October 2000

SA5#15(00)0483
Tdoc **S5C000153**

CHANGE REQUEST

Please see embedded help file at the bottom of this page for instructions on how to fill in this form correctly.

32.106-3	**CR**	**003**	Current Version:	V3.1.0

GSM (AA.BB) or 3G (AA.BBB) specification number ↑ *↑ CR number as allocated by MCC support team*

For submission to: **SA#10** *list expected approval meeting # here ↑*	for approval	X	strategic		*(for SMG use only)*
	for information		non-strategic		

Form: CR cover sheet, version 2 for 3GPP and SMG *The latest version of this form is available from: ftp://ftp.3gpp.org/Information/CR-Form-v2.doc*

Proposed change affects: *(at least one should be marked with an X)* (U)SIM [ ] ME [ ] UTRAN / Radio [X] Core Network [X]

Source: **SA5#15** **Date:** 20/10/2000

Subject: Spelling Errors in the CORBA IDL

Work item: OAM-CM

Category: *(only one category shall be marked with an X)*

F	Correction	X
A	Corresponds to a correction in an earlier release	
B	Addition of feature	
C	Functional modification of feature	
D	Editorial modification	

Release:

Phase 2	
Release 96	
Release 97	
Release 98	
Release 99	X
Release 00	

Reason for change: Spelling corrections in the CORBA IDL in order to remove confusion on the names:

`NV_ALARM_id`	is being changed to	`NV_ALARM_ID`
`NV_STATE_CHANGE_DEFINITIONS`	is being changed to	`NV_STATE_CHANGE_DEFINITION`
`NV_PROPOSED_REPAIRED_ACTIONS`	is being changed to	`NV_PROPOSED_REPAIR_ACTIONS`

Clauses affected: Annex A

Other specs affected:

Other 3G core specifications	[ ]	→ List of CRs:
Other GSM core specifications	[ ]	→ List of CRs:
MS test specifications	[ ]	→ List of CRs:
BSS test specifications	[ ]	→ List of CRs:
O&M specifications	[ ]	→ List of CRs:

Other comments:

Annex A (normative): Notification IRP CORBA IDL

```
  const string NV_NOTIFICATION_ID = "a";
  const string NV_CORRELATED_NOTIFICATIONS = "b";
  const string NV_EVENT_TIME = "c";
  const string NV_SYSTEM_DN = "d";
  const string NV_MANAGED_OBJECT_CLASS = "e";
  const string NV_MANAGED_OBJECT_INSTANCE = "f";
  const string NV_PROBABLE_CAUSE = "g";
  const string NV_PERCEIVED_SEVERITY = "h";
  const string NV_SPECIFIC_PROBLEM = "i";
  const string NV_ADDITIONAL_TEXT = "j";
  const string NV_ALARM_IDid = "k";
  const string NV_ACK_USER_ID = "l";
  const string NV_ACK_TIME = "m";
  const string NV_ACK_SYSTEM_ID = "n";
  const string NV_ACK_STATE = "o";
  const string NV_BACKED_UP_STATUS = "p";
  const string NV_BACK_UP_OBJECT = "q";
  const string NV_THRESHOLD_INFO = "r";
  const string NV_TREND_INDICATION = "s";
  const string NV_STATE_CHANGE_DEFINITIONS DEFINITION = "t";
  const string NV_MONITORED_ATTRIBUTES = "u";
  const string NV_PROPOSED_REPAIRED_ACTIONS = "v";
```

3GPP TSG-SA5 (Telecom Management)
Meeting #15, Girdwood, Alaska, USA, 16-20 October 2000

SA5#15(00)0485
S5C000157

CHANGE REQUEST

Please see embedded help file at the bottom of this page for instructions on how to fill in this form correctly.

32.106-3	**CR**	**004**	Current Version:	V.3.1.0

GSM (AA.BB) or 3G (AA.BBB) specification number ↑ *↑ CR number as allocated by MCC support team*

For submission to: **SA#10**	for approval	**X**	strategic		*(for SMG use only)*
list expected approval meeting # here ↑	for information		non-strategic		

Form: CR cover sheet, version 2 for 3GPP and SMG *The latest version of this form is available from: ftp://ftp.3gpp.org/Information/CR-Form-v2.doc*

Proposed change affects: *(at least one should be marked with an X)* (U)SIM [ ] ME [ ] UTRAN / Radio [X] Core Network [X]

Source: **SA5#15** **Date:** 20/10/2000

Subject: Ensure consistency with IDL exception

Work item: OAM-CM

Category: *(only one category shall be marked with an X)*

F	Correction	**X**	**Release:**	Phase 2	
A	Corresponds to a correction in an earlier release			Release 96	
B	Addition of feature			Release 97	
C	Functional modification of feature			Release 98	
D	Editorial modification			Release 99	**X**
				Release 00	

Reason for change:

Ensure consistency of text with IDL exception.

The method defined in IDL for getNotificationIRPVersion() does not throw exception. Table 6 indicates that an InvalidParameter and getNotificationIRPVersion exceptions can be thrown.

Corrections proposed:

1. Add getNotificationIRPVersion exception to IDL to make it consistent with other methods and consistent with Alarm IRP's getAlarmIRPVersion().
2. Remove InvalidParameter exception from Table 6 to make it consistent with IDL.

Clauses affected: 5.2 (Table 6), Annex A

Other specs affected:

Other 3G core specifications	[ ]	→ List of CRs:
Other GSM core specifications	[ ]	→ List of CRs:
MS test specifications	[ ]	→ List of CRs:
BSS test specifications	[ ]	→ List of CRs:
O&M specifications	[ ]	→ List of CRs:

Other comments:

Table 6: Mapping from IS `getNotificationIRPVersion` parameters to SS equivalents

IS Operation parameter	SS Method parameter	Qualifier
versionNumber List	Return value of type CommonIRPConstDefs::VersionNumberSet	M
status	GetNotificationIRPVersion~~,InvalidParameter~~	M

========== Page 19 ==========

```
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 Attach { string reason; };
  exception DetachException { string reason; };

  exception GetSubscriptionStatus { string reason; };
  exception GetSubscriptionIds { string reason; };
  exception ChangeSubscriptionFilter { string reason; };
  exception GetNotificationCategories { string reason; };
  exception GetNotificationIRPVersion { string reason; };

  exception ParameterNotSupported { string parameter; };
    // name of the unsupported parameter as defined in IDL
  exception InvalidParameter { string parameter; };
    // name of the parameter as defined in IDL
  exception OperationNotSupported {};
  exception AlreadySubscribed {};
  exception AtLeastOneNotificationCategoryNotSupported {};
```

========== Page 20 ==========

```
    /* ## Operation: get_notification_IRP_version
    */
    CommonIRPConstDefs::VersionNumberSet get_notification_IRP_version ()
    raises (GetNotificationIRPVersion);
    ;
```