TSGS#28(05)0300

Source: **SA5 (Telecom Management)**

Title: **CR 32672-3 State Management IRP CORBA SS**

Document for: **Approval**

Agenda Item: **7.5.3**

Doc-1st-Level	Spec__#_	CR__#	R	Phase	Subject	Cat	Ver-Cur	Doc-2nd-Level	Workitem
SP-050300	32.673	0006	-	Rel-5	Align AlarmStatus with the IS in TS 32.672	F	5.2.0	S5-058416	OAM-NIM
SP-050300	32.673	0007	-	Rel-6	Align AlarmStatus with the IS in TS 32.672	A	6.2.0	S5-058417	OAM-NIM

CR-Form-v7.1

CHANGE REQUEST

⌘ **32.673** **CR** **0006** ⌘**rev** **-** ⌘ Current version: **5.2.0** ⌘

*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 **X** Core Network **X**

Title:	⌘	Align AlarmStatus with the IS in TS 32.672
Source:	⌘	SA5 Lucent Technologies (islip@lucent.com)
Work item code:	⌘	OAM-NIM
Date:	⌘	13/05/2005
Category:	⌘	**F**
Release:	⌘	Rel-5

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:
Ph2 *(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)*
Rel-7 *(Release 7)*

Reason for change:	⌘	alarm Status is defined as set valued which does not align with the IS nor M.3100.
Summary of change:	⌘	Amend alarmStatus so as to be a single value, not set valued. Align enumerated value with CORBA style guide.
Consequences if not approved:	⌘	AlarmStatus definition is current set valued and does not align with the IS.

Clauses affected:	⌘	Annex A1

		Y	N		
Other specs affected:	⌘		**N**	Other core specifications	⌘
			N	Test specifications	
			N	O&M Specifications	

Other comments: ⌘

Change in Clause Annex A.1

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

```
#ifndef StateManagementIRPConstDefs_idl
#define StateManagementIRPConstDefs_idl

#include "CosNotification.idl"
#include "ManagedGenericIRPConstDefs.idl"

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

/* ## Module: StateManagementIRPConstDefs
This module contains commonly used definitions for State Management IRP
=========================================================================
*/
module StateManagementIRPConstDefs
{

   /*
   Definition of Operational State based on X.721 [3], if mandatory.
   */
   enum OperationalState
   {
      Disabled, Enabled
   };

   /*
   Definition of Usage State based on X.721 [3], if mandatory.
   */
   enum UsageState
   {
      Idle, Active, Busy
   };

   /*
   Definition of Administrative State based on X.721 [3], if mandatory.
   */
   enum AdministrativeState
   {
      Locked, Unlocked, ShuttingDown
   };

   /*
   Definition of Alarm Status based on M.3100 [4], if mandatory.
   */
   enum AlarmStatus~~Values~~
   {
      CLEARED, INDETERMINATE, WARNING, MINOR, MAJOR, ~~Critical~~CRITICAL
   };
   ~~typedef sequence <AlarmStatusValues,5> AlarmStatus;~~

   /*
   Definition of Procedural Status based on X.721 [3], if mandatory.
   */
   enum ProceduralStatusValues
   {
      InitializationRequired, NotInitialized, Initializing, Reporting,
      Terminating
   };
   typedef sequence <ProceduralStatusValues,5> ProceduralStatus;

   /*
   Definition of Availability Status based on X.721 [3], if mandatory.
   */
   enum AvailabilityStatusValues
   {
      InTest, Failed, PowerOff, OffLine, OffDuty, Dependency, Degraded,
      NotInstalled, LogFull
   };
   typedef sequence <AvailabilityStatusValues,9> AvailabilityStatus;
```

```
    /*
    Definition of Control Status based on X.721 [3], if mandatory.
    */
    enum ControlStatusValues
    {
       SubjectToTest, PartOfServicesLocked, ReservedForTest, Suspended
    };
    typedef sequence <ControlStatusValues,4> ControlStatus;

    /*
    Definition of Standby Status based on X.721 [3], if mandatory.
    */
    enum StandbyStatus
    {
       HotStandby, ColdStandby, ProvidingService
    };

    /*
    Definition of Unknown Status based on X.721 [3], if mandatory
    (if switch is TRUE then value equal to TRUE implies "unknown status").
    */
    union UnknownStatus switch(boolean)
    {
       case TRUE: boolean value;
    };

};
#endif
```

End of Change in Annex A.1
End of Document

Annex B (informative): Change history

Change history							
Date	**TSG #**	**TSG Doc.**	**CR**	**Rev**	**Subject/Comment**	**Old**	**New**
Sep 2002	S_17	SP-020470	--	--	Submitted to TSG SA #17 for Approval	1.0.0	5.0.0
Mar 2003	S_19	SP-030143	001	--	CORBA IDL Compiler Errors, Invalid CORBA IDL Include Reference	5.0.0	5.1.0
Sep 2004	S_25	SP-040588	002	--	Correction of the alarmStatus mapping – Align with 32.672 CM; State Management IRP Information Service	5.1.0	5.2.0

CR-Form-v7.1

CHANGE REQUEST

⌘ **32.673** CR **0007** ⌘rev **-** ⌘ Current version: **6.2.0** ⌘

*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 **X** Core Network **X**

Title:	⌘	Align AlarmStatus with the IS in TS 32.672
Source:	⌘	SA5 Lucent Technologies (islip@lucent.com)
Work item code:	⌘	OAM-NIM
Date:	⌘	13/05/2005
Category:	⌘	**A**
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:
Ph2 (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)
Rel-7 (Release 7)

Reason for change:	⌘	alarm Status is defined as set valued which does not align with the IS nor M.3100.
Summary of change:	⌘	Amend alarmStatus so as to be a single value, not set valued. Align enumerated value with CORBA style guide.
Consequences if not approved:	⌘	AlarmStatus definition is current set valued and does not align with the IS.

Clauses affected:	⌘	Annex A1

Other specs affected: ⌘	Y	N	
		X	Other core specifications
		X	Test specifications
		X	O&M Specifications

⌘

Other comments: ⌘

Change in Clause Annex A.1

Annex A (normative): IDL specifications

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

```
#ifndef StateManagementIRPConstDefs_idl
#define StateManagementIRPConstDefs_idl

#include "CosNotification.idl"
#include "ManagedGenericIRPConstDefs.idl"
#include "StateManagementIRPOptConstDefs.idl"

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

/* ## Module: StateManagementIRPConstDefs
This module contains commonly used definitions for State Management IRP
=========================================================================
*/
module StateManagementIRPConstDefs
{
/*
  Constant definitions for state management notifications uses when populating the
  Cos::Structured event.
  The "name" party of the structured event carries the following constant definitions
  appropriate to the state being notified.
  Refer to TS 32.663 regarding how to populate the structured event
*/
   interface AttributeNameValue {
      const string OPERATIONAL_STATE    = "operationalState";
      const string USAGE_STATE          = "usageState";
      const string ADMINISTRATIVE_STATE = "administrativeState";
      const string ALARM_STATUS         = "alarmStatus";
      const string PROCEDURAL_STATUS    = "proceduralStatus";
      const string AVAILABILITY_STATUS  = "availabilityStatus";
      const string CONTROL_STATUS       = "controlStatus";
      const string STANDBY_STATUS       = "standbyStatus";
      const string UNKNOWN_STATUS       = "unknownStatus";
   };


/*
   The following structures provide the new state value,
   and the optional old state value
   The structures are passed in the value part of the cos structured event
*/
   Struct OperationalStateOldNewValue{
      OperationalState new;
      StateManagementIRPOptConstDefs::OperationalStateTypeOpt old;
   };


   Struct UsageStateOldNewValue{
      UsageState new;
      StateManagementIRPOptConstDefs:: UsageStateTypeOpt old;
   };


   Struct AdministrativeStateOldNewValue{
      AdministrativeState new;
      StateManagementIRPOptConstDefs:: AdministrativeStateTypeOpt old;
   };


   Struct AlarmStatusOldNewValue{
      AlarmStatus~~Values~~ new;
```

```
      StateManagementIRPOptConstDefs:: AlarmStatusTypeOptAdministrativeStateTypeOpt old;
   };


   Struct ProceduralStatusOldNewValue{
      ProceduralStatusValues new;
      StateManagementIRPOptConstDefs:: ProceduralStatusTypeOpt old;
   };


   Struct AvailabilityStatusOldNewValue{
      AvailabilityStatusValues new;
      StateManagementIRPOptConstDefs:: AvailabilityStatusTypeOpt old;
   };


   Struct ControlStatusOldNewValue{
      ControlStatusValues new;
      StateManagementIRPOptConstDefs:: ControlStatusTypeOpt old;
   };


   Struct StandbyStatusOldNewValue{
      StandbyStatus new;
      StateManagementIRPOptConstDefs:: StandbyStatusTypeOpt old;
   };


   Struct UnknownStatusOldNewValue{
      UnknownStatus new;
      StateManagementIRPOptConstDefs:: UnknownStatusTypeOpt old;
   };


   /*
   Definition of Operational State based on X.721 [3], if mandatory.
   */
   enum OperationalState
   {
      Disabled, Enabled
   };

   /*
   Definition of Usage State based on X.721 [3], if mandatory.
   */
   enum UsageState
   {
      Idle, Active, Busy
   };

   /*
   Definition of Administrative State based on X.721 [3], if mandatory.
   */
   enum AdministrativeState
   {
      Locked, Unlocked, ShuttingDown
   };

   /*
   Definition of Alarm Status based on M.3100 [4], if mandatory.
   */
   enum AlarmStatusValues
   {
      CLEARED, INDETERMINATE, WARNING, MINOR, MAJOR, CRITICAL
   };
   typedef sequence <AlarmStatusValues,5> AlarmStatus;

   /*
   Definition of Procedural Status based on X.721 [3], if mandatory.
   */
   enum ProceduralStatusValues
   {
      InitializationRequired, NotInitialized, Initializing, Reporting,
      Terminating
   };
   typedef sequence <ProceduralStatusValues,5> ProceduralStatus;

   /*
```

```
   Definition of Availability Status based on X.721 [3], if mandatory.
   */
   enum AvailabilityStatusValues
   {
      InTest, Failed, PowerOff, OffLine, OffDuty, Dependency, Degraded,
      NotInstalled, LogFull
   };
   typedef sequence <AvailabilityStatusValues,9> AvailabilityStatus;

   /*
   Definition of Control Status based on X.721 [3], if mandatory.
   */
   enum ControlStatusValues
   {
      SubjectToTest, PartOfServicesLocked, ReservedForTest, Suspended
   };
   typedef sequence <ControlStatusValues,4> ControlStatus;

   /*
   Definition of Standby Status based on X.721 [3], if mandatory.
   */
   enum StandbyStatus
   {
      HotStandby, ColdStandby, ProvidingService
   };

   /*
   Definition of Unknown Status based on X.721 [3], if mandatory
   (if switch is TRUE then value equal to TRUE implies "unknown status").
   */
   union UnknownStatus switch(boolean)
   {
      case TRUE: boolean value;
   };

};
#endif
```

End of Change in Annex A.1
End of Document

Annex B (informative): Change history

Change history							
Date	**TSG #**	**TSG Doc.**	**CR**	**Rev**	**Subject/Comment**	**Old**	**New**
Sep 2002	S_17	SP-020470	--	--	Submitted to TSG SA #17 for Approval	1.0.0	5.0.0
Mar 2003	S_19	SP-030143	001	--	CORBA IDL Compiler Errors, Invalid CORBA IDL Include Reference	5.0.0	5.1.0
Mar 2004	S_23	SP-040105	--	--	Automatic upgrade to Rel-6 (no CR)	5.1.0	6.0.0
Sep 2004	S_25	SP-040588	003	--	Correction of the alarmStatus mapping – Align with 32.672 CM; State Management IRP Information Service	6.0.0	6.1.0
Sep 2004	S_25	SP-040569	004	--	Provide constant definitions to support state change events	6.0.0	6.1.0
Mar 2005	S_27	SP-050051	005	--	Apply the Generic System Context, update of reference to IS specification – Align with TS 32.672	6.1.0	6.2.0