**3GPP TSG- Meeting # *57***

**, , - revision of S5-241130**

|  |
| --- |
| *CR-Form-v12.3* |
| **CHANGE REQUEST** |
|  |
|  |  | **CR** |  | **rev** | **1** | **Current version:** |  |  |
|  |
| *For* [***HE******LP***](http://www.3gpp.org/3G_Specs/CRs.htm#_blank)*on using this form: comprehensive instructions can be found at* [*http://www.3gpp.org/Change-Requests*](http://www.3gpp.org/Change-Requests)*.* |
|  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***Proposed change affects:*** | UICC apps |  | ME |  | Radio Access Network | **X** | Core Network | **X** |

|  |
| --- |
|  |
| ***Title:***  |  |
|  |  |
| ***Source to WG:*** |  |
| ***Source to TSG:*** | S5 |
|  |  |
| ***Work item code:*** | 9 |  | ***Date:*** |  |
|  |  |  |  |  |
| ***Category:*** |  |  | ***Release:*** | Rel-19 |
|  | *Use one of the following categories:****F*** *(correction)****A*** *(mirror corresponding to a change in an earlier release)****B*** *(addition of feature),* ***C*** *(functional modification of feature)****D*** *(editorial modification)*Detailed explanations of the above categories canbe found in 3GPP [TR 21.900](http://www.3gpp.org/ftp/Specs/html-info/21900.htm). | *Use one of the following releases:Rel-8 (Release 8)Rel-9 (Release 9)Rel-10 (Release 10)Rel-11 (Release 11)…Rel-17 (Release 17)Rel-18 (Release 18)Rel-19 (Release 19) Rel-20 (Release 20)* |
|  |  |
| ***Reason for change:*** | For operations and notifications the multiplicity of input and output parameters is not indicated. While the support qualifier is part of the parameter tables, support for a parameter only means that the producer is capable of handling that parameter. It does not mean that the parameter will actually be present in the notification/operation. Thus the consumer cannot depend on the presence of a parameter even if it is mandatory-to-support.Just as in the NRM attribute definitions we need to separately include information about- optional/mandatory to support : the supportQualifier- optional/mandatory to use : the multiplicityA goal is to make the description of NRM attributes and notification/operation parameters similar. |
|  |  |
| ***Summary of change:*** | - Introduce multiplicity information for operation and notification parameters.- Remove filtering qualifier as it is not defined anywhere.- Change GeneralizedTime to dateTime as this has already been changed in all other specifications e.g. TS 32.156 |
|  |  |
| ***Consequences if not approved:*** | Today the consumers does not know what to expect from the producer, whether individual parameters will be or will not be part of an operation/notification. This might lead to interoperability problems.  |
|  |  |
| ***Clauses affected:*** | 5.3 |
|  |  |
|  | **Y** | **N** |  |  |
| ***Other specs*** |  | **X** |  Other core specifications  | TS/TR ... CR ...  |
| ***affected:*** |  | **X** |  Test specifications | TS/TR ... CR ...  |
| ***(show related CRs)*** |  | **X** |  O&M Specifications | TS/TR ... CR ...  |
|  |  |
| ***Other comments:*** |  |
|  |  |
| ***This CR's revision history:*** |  |

**First change**

##  5.3 Template for Management service operations and notifications

Y4 Overview

Yb Management service name

*Management service name should be replaced with the name of the Management Service (MnS).*

*"b" represents a number, starting at 1 and increasing by 1 with each new definition of a Management Service.*

Yb.1 Operations and notifications

Yb.1.a Operation OperationName

*OperationName is the name of the operation followed by a qualifier indicating whether the operation is Mandatory (M), Optional (O), Conditional-Mandatory (CM), Conditional-Optional (CO), or SS-Conditional (C).*

*"a" represents a number, starting at 1 and increasing by 1 with each new definition of an operation.*

Yb.1.a.1 Definition

Yb.1.a.1.1 Description

*This subclause shall be written in natural language.*

*Operations have a lifecycleStatus property as defined by [3] clause 5.2.A. If and only if the lifecycleStatus is not current (its default value), that shall be indicated in this subclause.*

*Information on traceability back to one or more requirements supported by this operation should also be defined here, in the following form:*

|  |  |  |
| --- | --- | --- |
| **Referenced TS** | **Requirement label** | **Comment** |
| 3GPP TS 32.xyz [xy] | REQ-SM-CON-23 | *Optional clarification* |
| 3GPP TS 32.xyz [xy] | REQ-SM-FUN-11 | *Optional clarification* |

Yb.1.a.1.2 Pre-condition

*A pre-condition is a collection of assertions joined by AND, OR, and NOT logical operators. The pre-condition shall be true before the operation is invoked. An example is given here below:*

*notificationCategoriesNotAllSubscribed OR notificationCategoriesParameterAbsentAndNotAllSubscribed*

*Each assertion is defined by a pair (propertyName, propertyDefinition). All assertions constituting the pre-condition are provided in a table. An example of such a table is given here below:*

|  |  |
| --- | --- |
| **Assertion Name** | **Definition** |
| notificationCategoriesNotAllSubscribed | At least one notificationCategory identified in the notificationCategories input parameter is supported by an MnS producer and is not a member of the ntfNotificationCategorySet attribute of an NtfSubscription which is involved in a subscription relationship with the NtfSubscriber identified by the managerReference input parameter. |
| notificationCategoriesParameterAbsentAndNotAllSubscribed | The notificationCategories input parameter is absent and at least one notificationCategory supported by MnS producer is not a member of the ntfNotificationCategorySet attribute of an ntfSsubscription which is involved in a subscription relationship with the NtfSubscriber identified by the managerReference input parameter. |

Yb.1.a.1.3 Post-condition

*A post-condition is a collection of assertions joined by AND, OR, and NOT logical operators. The post-condition shall be true after the completion of the operation. When nothing is said in a post-condition regarding an information entity, the assumption is that this information entity has not changed compared to what is stated in the
pre-condition. An example is given here below:*

*subscriptionDeleted OR allSubscriptionDeleted*

*Each assertion is defined by a pair (propertyName, propertyDefinition). All assertions constituting the post-condition shall be provided in a table. An example of such a table is given here below:*

|  |  |
| --- | --- |
| **Assertion Name** | **Definition** |
| subscriptionDeleted | The ntfSubscription identified by subscriptionId input parameter is no more involved in a subscription relationship with the ntfSubscriber identified by the managerReference input parameter and has been deleted. If this ntfSubscriber has no more ntfSubscription, it is deleted as well. |
| allSubscriptionDeleted | In the case subscriptionId input parameter was absent, the ntfSubscriber identified by the managerReference input parameter is no more involved in any subscription relationship and is deleted, the corresponding ntfSubscription have been deleted as well. |

Yb.1.a.1.4 Exceptions

*List of exceptions that can be raised by the operation. Each element shall be a tuple (exceptionName, condition, ReturnedInformation, exitState).*

Yb.1.a.1.4.c exceptionName

*ExceptionName is the name of an exception.*

*"c" represents a number, starting at 1 and increasing by 1 with each new definition of an exception.*

*This information shall be provided in a table. An example of such a table is given here below:*

| **Exception Name** | **Definition** |
| --- | --- |
| ope\_failed\_existing\_subscription | **Condition:** (notificationCategoriesNotAllSubscribed OR notificationCategoriesParameterAbsentAndNotAllSubscribed) not verified.**Returned information:** output parameter status is set to OperationFailedExistingSubscription.**Exit state:** Entry State. |

NOTE: An example of an exception can be a situation where an operation is raised and the required information between a consumer and producer cannot be conveyed via the input and output parameters.

Yb.1.a.2 Input parameters

*List of input parameters of the operation. Each element contains the Parameter Name, Support Qualifier, Documentation and Allowed Values and Properties. Legal Values for the Support Qualifier are: Mandatory (M), Optional (O), Conditional-Mandatory (CM), Conditional-Optional (CO), or SS-Conditional (C).*

*Properties shall include type and multiplicity. If multiplicity allows multiple values the properties isOrdered and isUnique shall also be included, if multiplicity is not greater than 1 isOrdered and isUnique shall be absent. The individual properties shall follow the same rules as attribute properties, see clause 5.2.*

*This information shall be provided in a table. An example of such a table is given here below:*

| **Parameter Name** | **S** | **Documentation and Allowed Values** | **Properties** |
| --- | --- | --- | --- |
| eventIdList | M | One or more event identifiers  | TypemultiplicityisOrderedisUnique |

NOTE: In the case where the Allowed Values can be enumerated, each element is a pair (Allowed Value Name, Allowed Value Semantics), unless an Allowed Value Semantics applies to several values in which case the definition can be provided only once.

Yb.1.a.3 Output parameters

*List of output parameters of the operation. Each element contains the Parameter Name, Support Qualifier, Documentation and Allowed Values and Properties. Legal Values for the Support Qualifier are: Mandatory (M), Optional (O), Conditional-Mandatory (CM), Conditional-Optional (CO), or SS-Conditional (C).*

*Properties shall include type and multiplicity. If multiplicity allows multiple values the properties isOrdered and isUnique shall also be included, if multiplicity is not greater than 1 isOrdered and isUnique shall be absent. The individual properties shall follow the same rules as attribute properties, see clause 5.2.*

*This information shall be provided in a table. An example of such a table is given here below:*

| **Parameter Name** | **S** | **Documentation and Allowed Values** | **Properties** |
| --- | --- | --- | --- |
| eventTime | M | The parameter carries the* alarmRaisedTime in case notificationType carries notifyNewAlarm,
* alarmChangedTime in case notificationType carries notifyChangedAlarm,

alarmClearedTime in case notificationType carries notifyClearedAlarm.E.g. AlarmRecord.alarmRaisedTime  | TypemultiplicityisOrderedisUnique |

NOTE: Information Type qualifies the parameter of Parameter Name. In the case where the Legal Values can be enumerated, each element is a pair (Legal Value Name, Legal Value Semantics), unless a Legal Value Semantics applies to several values in which case the definition can be provided only once. When the Legal Values cannot be enumerated, the list of Legal Values is defined by a single definition.

*This table shall also include a special parameter ’status’ to indicate the completion status of the operation (success, partial success, failure reason etc.).*

Yb.1.a.4 Result

Yb.1.a.4,1 Error messages

*This subclause presents error messages in case the operation is not successful.*

*This subclause does not need to be present when there are no error messages to define.*

Yb.1.a.4,2 Constraints

*This subclause presents constraints for the operation or its parameters.*

*This subclause does not need to be present when there are no constraints to define.*

Yb.1.a Notification NotificationName

*NotificationName shall be the name of the notification followed by a qualifier indicating whether the notification is Mandatory (M), Optional (O), Conditional-Mandatory (CM), Conditional-Optional (CO) or SS-Conditional (C).*

*"a" represents a number, starting at 1 and increasing by 1 with each new definition of a notification.*

Yb.1.a.1 Definition

*This subclause shall be written in natural language.*

*Notifications have a lifecycleStatus property as defined by [3] clause 5.2.A. If and only if the lifecycleStatus is not current (its default value), that shall be indicated in this subclause.*

*Information on traceability back to one or more requirements supported by this notification should also be defined here, in the following form:*

|  |  |  |
| --- | --- | --- |
| **Referenced TS** | **Requirement label** | **Comment** |
| 3GPP TS 32.xyz [xy] | REQ-SM-CON-23 | *Optional clarification* |
| 3GPP TS 32.xyz [xy] | REQ-SM-FUN-11 | *Optional clarification* |

Yb.1.a.2 Input parameters

*List of input parameters of the notification. Each element contains the Parameter Name, Support Qualifier, Documentation and Allowed Values and Properties. The Support Qualifier indicates whether the attribute is Mandatory (M), Optional (O), Conditional-Mandatory (CM), Conditional-Optional (CO), or SS-Conditional (C).*

*Properties shall include type and multiplicity. If multiplicity allows multiple values the properties isOrdered and isUnique shall also be included, if multiplicity is not greater than 1 isOrdered and isUnique shall be absent. The individual properties shall follow the same rules as attribute properties, see clause 5.2.*

*This information shall be provided in a table. An example of such a table is given here below:*

| **Parameter Name** | **S** | **Documentation and Allowed Values** | **Properties** |
| --- | --- | --- | --- |
| managerReference | M | It specifies the reference of the consumer to which notifications shall be sent.E.g. ntfSubscriber.ntfManagerReference  | Type: StringMultiplicity: 0..\*isOrdered: FalseisUnique: True |
| alarmType | M | AlarmInformation.eventType "Communications Alarm": a communication error alarm."Processing Error Alarm": a processing error alarm."Environmental Alarm": an environmental violation alarm. "Quality Of Service Alarm": a quality of service violation alarm."Equipment Alarm": an alarm related to equipment malfunction. | Type: ENUMMultiplicity: 0..1 |

NOTE: In the case where the Allowed Values can be enumerated, each element is a pair (Allowed Value Name, Allowed Value Semantics), unless an Allowed Value Semantics applies to several values in which case the definition can be provided only once.

Yb.1.a.3 Triggering event

*The triggering event for the notification to be sent is the transition from the information state defined by the "from state" subclause to the information state defined by the "to state" subclause.*

Yb.1.a.3.1 From state

*This subclause is a collection of assertions joined by AND, OR, and NOT logical operators. An example is given here below:*

*alarmMatched AND alarmInformationNotCleared*

*Each assertion is defined by a pair (propertyName, propertyDefinition). All assertions constituting the state "from state" are provided in a table. An example of such a table is given here below:*

|  |  |
| --- | --- |
| **Assertion Name** | **Definition** |
| alarmMatched | The matching-criteria-attributes of the newly generated network alarm has values that are identical (matches) with ones in one AlarmInformation in AlarmList.  |
| alarmInformationNotCleared | The perceivedSeverity of the newly generated network alarm is not Cleared. |

Yb.1.a.3.2 To state

*This subclause contains a collection of assertions joined by AND, OR and NOT logical operators. When nothing is said in a to-state regarding an information entity, the assumption is that this information entity has not changed compared to what is stated in the from-state. An example is given here below:*

*resetAcknowledgementInformation AND perceivedSeverityUpdated*

*Each assertion is defined by a pair (propertyName, propertyDefinition). All assertions constituting the state "to state" are provided in a table. An example of such a table is given here below:*

|  |  |
| --- | --- |
| **Assertion Name** | **Definition** |
| resetAcknowledgementInformation | The matched AlarmInformation identified in inv\_alarmMatched in pre-condition has been updated according to the following rule:ackTime, ackUserId and ackSystemId are updated to contain no information; ackState is updated to "unacknowledged". |
| perceivedSeverityUpdated | The perceivedSeverity attribute of matched AlarmInformation identified in inv\_alarmMatched in pre-condition has been updated. |

Yb.2 Managed information

**End of changes**