3GPP TSG SA WG5 Meeting 136-e S5-212201

electronic meeting, online, 1 - 9 March 2021

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *CR-Form-v12.1* | | | | | | | | |
| **CHANGE REQUEST** | | | | | | | | |
|  | | | | | | | | |
|  | **28.532** | **CR** | **0168** | **rev** | **-** | **Current version:** | **16.6.0** |  |
|  | | | | | | | | |
| *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:*** | Correct the misalignment information for stage2 Fault Supervision MnS | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Source to WG:*** | S5 | | | | | | | | | |
| ***Source to TSG:*** | Huawei | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Work item code:*** | TEI16 | | | | |  | ***Date:*** | | | 2021-02-07 |
|  |  | | | |  | |  | | |  |
| ***Category:*** | **F** |  | | | | | ***Release:*** | | | Rel-16 |
|  | *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 can be 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-15 (Release 15) Rel-16 (Release 16) Rel-17 (Release 17) Rel-18 (Release 18)* | |
|  |  | | | | | | | | | |
| ***Reason for change:*** | | 1. The title of Clause 11.2.1.1 is ‘Operation and notification of fault supervision data report management service’, and the title of clause 11.2.1.2 ‘Fault supervision data control management service’. However, the currently only generic Fault Supervision MnS is defined in TS 28.532, no definition for ‘fault supervision data report management service’ and ‘Fault supervision data control management service’. 2. Both term ‘MnS producer’ and ‘service provider’ are used, which represent same object. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Summary of change:*** | | 1. Correct the title for clause 11.2.1.1 and clause 11.2.1.2. 2. Replace the term ‘service provider’ with ‘MnS producer’. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Consequences if not approved:*** | | Some misalignment information exist in TS 28.532 for fault supervision MnS | | | | | | | | |
|  | |  | | | | | | | | |
| ***Clauses affected:*** | | 11.2.1.1, 11.2.1.1.1.2,11.2.1.1.3.2,11.2.1.1.3.3,11.2.1.1.6.3.1,11.2.1.1.6.3.2,11.2.1.1.8.2,11.2.1.2,11.2.1.2.3.2,11.2.1.2.7.3.1,11.2.1.2.7.3.2, | | | | | | | | |
|  | |  | | | | | | | | |
|  | | **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:*** | |  | | | | | | | | |

|  |
| --- |
| **1st Change** |

## 1.2 Generic fault supervision management service

### 11.2.1 Operations and notifications

#### 11.2.1.1 Fault supervision data report

##### 11.2.1.1.1 subscribe

###### 11.2.1.1.1.1 Definition

A MnS consumer invokes this operation to establish subscription to receive network events via notifications, under the filter constraint specified in this operation.

###### 11.2.1.1.1.2 Input parameters

| Parameter Name | S | Information Type / Legal Values | Comment |
| --- | --- | --- | --- |
| consumerReference | M | NtfSubscriber.ntfManagerReference | It specifies the reference of the authorized MnS consumer to which notifications shall be sent. |
| timeTick | O | NtfSubscription.ntfTimeTick | It specifies the value of a timer held for the subject management service consumer.  The value is in unit of whole minute.  A special infinite value is assumed when parameter is absent or present but equal to zero. |
| filter | O | This attribute represents the filter of a subscription. | It specifies a filter constraint that MnS producer shall use to filter notification of the alarms.  If this parameter is absent, then no filter constraint shall be applied. |

###### 11.2.1.1.1.3 Output parameters

| Parameter Name | S | Matching Information /  Information Type / Legal Values | Comment |
| --- | --- | --- | --- |
| subscriptionId | M | NtfSubscription.ntfSubscriptionId. | It holds an unambiguous identity of this subscription. |
| status | M | ENUM (OperationSucceeded, OperationFailedExistingSubscription, OperationFailed) | If subscriptionCreated is true, status = OperationSuceeded.  If operation\_failed\_existing\_subscription is true, status = OperationFailedExistingSubscription  If operation\_failed is true, status = OperationFailed. |

###### 11.2.1.1.1.4 Pre-condition

notificationCategoriesNotAllSubscribed OR notificationCategoriesParameterAbsentAndNotAllSubscribed.

|  |  |
| --- | --- |
| Assertion Name | Definition |
| notificationCategoriesNotAllSubscribed | At least one notificationCategory identified in the notificationCategories input parameter is supported by management service 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 management service 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. |

###### 11.2.1.1.1.5 Post-condition

subscriberPossiblyCreated AND subscriptionCreated.

|  |  |
| --- | --- |
| Assertion Name | Definition |
| subscriberPossiblyCreated | An NtfSubscriber with an ntfManagerReference attribute equal to the value of the managerReference input parameter is involved in a subscriptionRegistration relationship. |
| subscriptionCreated | An NtfSubscription has been created according to the following rules:  - ntfSubscriptionState attribute value has been set to "notSuspended";  - ntfTimeTick attribute value has been set to the value of the timeTick input parameter if This value was higher or equal to 15, or set to 15 if this parameter value was between 1 and 15, or set to a special infinite value if the parameter value was lower or equal to 0 or if parameter was absent;  - ntfTimeTickTimer has been reset with the value of timeTick attribute;  - ntfFilter attribute value has been set to the value of the filter input parameter if present;  - NtfSubscription is involved in a subscription relationship with the NtfSubscriber identified by the managerReference input parameter;  - attribute ntfNotificationCategorySet of NtfSubscription contains EITHER the notification categories identified by the notificationCategories input parameter that were not already contained in the ntfNotificationCategorySet attribute of other NtfSubscription of the same NtfSubscriber identified by the managerReference input parameter OR if notificationCategories input parameter is absent, all notification categories supported by management service producer that were not already contained in the ntfNotificationCategorySet attribute of other subscriptions of the same NtfSubscriber identified by the managerReference input parameter. |

###### 11.2.1.1.1.6 Exceptions

|  |  |
| --- | --- |
| Name | Definition |
| operation\_failed\_existing\_subscription | **Condition:** (notificationCategoriesNotAllSubscribed OR notificationCategoriesParameterAbsentAndNotAllSubscribed) not true  **Returned Information:** The output parameter status  **Exit state:** Entry State |
| operation\_failed | **Condition:** Post-condition is false  **Returned Information:** The output parameter status  **Exit state:** Entry State |

##### 11.2.1.1.2 unsubscribe

###### 11.2.1.1.2.1 Definition

A MnS consumer invokes this operation to cancel subscriptions. The MnS consumer can cancel one subscription made with a consumerReference by providing the corresponding subscriptionId or all subscriptions made with the same consumerReference by leaving the subscriptionId parameter absent.

###### 11.2.1.1.2.2 Input parameters

| Parameter Name | S | Information Type / Legal Values | Comment |
| --- | --- | --- | --- |
| consumerReference | M | DN | It specifies the reference of the MnS consumer to which notifications shall be sent. |
| subscriptionId | O | A unique identifier that is SS dependent. | It holds a subscriptionId carried as the output parameter in the subscribe operation. |

###### 11.2.1.1.2.3 Output parameters

| Parameter Name | S | Matching Information /  Information Type / Legal Values | Comment |
| --- | --- | --- | --- |
| status | M | ENUM (OperationSucceeded, OperationFailed) | If (subscriptionDeleted OR allSubscriptionDeleted) is true, status = OperationSucceeded.  If operation\_failed is true, status = OperationFailed. |

###### 11.2.1.1.2.4 Pre-condition

validSubscriptionId&ManagerReference OR SubscriptionIdAbsent&ValidManagerReference.

|  |  |
| --- | --- |
| Assertion Name | Definition |
| validSubscriptionId&ManagerReference | The NtfSubscription identified by subscriptionId input parameter is involved in a subscription relationship with the NtfSubscriber identified by the managerReference input parameter. |
| subscriptionIdAbsent&ValidManagerReference | The subscriptionId input parameter is absent and the NtfSubscriber identified by the managerReference input parameter exists. |

###### 11.2.1.1.2.5 Post-condition

subscriptionDeleted OR allSubscriptionDeleted.

|  |  |
| --- | --- |
| 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. |

###### 11.2.1.1.2.6 Exceptions

|  |  |
| --- | --- |
| Name | Definition |
| operation\_failed | **Condition:** Pre-condition is false or post-condition is false  **Returned Information:** The output parameter status  **Exit state:** Entry State |

##### 11.2.1.1.3 getAlarmList

###### 11.2.1.1.3.1 Definition

A MnS consumer invokes this operation to request the MnS producer to provide either the complete list of AlarmInformation instances in the AlarmList or only a part of this list (partial alarm alignment).

The parameters baseObjectClass and baseObjectInstance are used to identify the part of the alarm list to be returned. If they are absent, then the complete alarm list shall be provided (full alarm alignment). If they identify a particular class instance, then only a) the AlarmInformation instances related to this class instance and b) the AlarmInformation instances related to the subordinate class instances of this class instance shall be provided (partial alarm alignment). An instance-a is said to be subordinate to instance-b if the DN of the latter is part of the DN of the former.

There are two modes of operation. One mode is synchronous. In this mode, the list of AlarmInformation instances in AlarmList is returned synchronously with the operation. The other mode is asynchronous. In this mode, the list of AlarmInformation instances is returned via alarm notifications. In asynchronous mode of operation, the only information returned synchronously is the status of the operation. A method allowing to abort an ongoing alarm alignment process shall be available in the asynchronous mode. The mode of operation to be used is determined by means outside the scope of specification. To use asynchronous mode, the authorized consumer needs to have established a subscription via the subscribe operation.

###### 11.2.1.1.3.2 Input parameters

| Parameter Name | S | Information Type / Legal Values | Comment |
| --- | --- | --- | --- |
| alarmAckState | O | ENUM (all alarms, all active alarms, all active and acknowledged alarms, all active and unacknowledged, all Cleared and unacknowledged alarms, all unacknowledged) | It carries a constraint. The FaultSupervision MnS producer shall apply it on AlarmInformation instances in AlarmList when constructing its output parameter AlarmInformationList. |
| baseObjectClass | O, see note 1 | This parameter is either absent or carries the object class of a certain class. | See how this attribute is used to support full alarm alignment and partial alarm alignment in 11.1.2.3.3.1.  See note 2. |
| baseObjectInstance | O, see note 1 | This parameter is either absent or carries the DN of a certain class instance. | See how this attribute is used to support full alarm alignment and partial alarm alignment in 11.1.2.3.3.1.  See note 2. |
| filter | O | N/A | It carries a filter constraint.  If the filter is present, the MnS producer shall apply it on AlarmInformation instances in AlarmList when constructing its output parameter AlarmInformationList.  If the filter is not present, all of the AlarmInformation instances included by the scope are selected. |
| NOTE 1: If the notification notifyAlarmListRebuilt supports indicating that only a part of the alarm list has been rebuilt then the operation getAlarmList shall support partial alarm alignment.  NOTE 2: The legal values of the parameters baseObjectClass and baseObjectInstance are restricted to those carried by the parameters baseObjectClass and baseObjectInstance in the recent notifyAlarmListRebuilt notifications. The timeline for "recent" is vendor-specific. | | | |

###### 11.2.1.1.3.3 Output parameters

Table 11.2.1.1.3.3-1: Output parameters for the operation getAlarmList

| Parameter Name | S | Matching Information /  Information Type / Legal Values | Comment |
| --- | --- | --- | --- |
| alarmInformationList | M | List of AlarmInformation. | It carries the requested AlarmInformation instances.  Case when synchronous mode of operation is used:  (a) The MnS producer shall apply the constraints expressed in alarmAckState and filter to AlarmInformation instances when constructing this output parameter.  Case when asynchronous mode of operation is used (i.e. this output parameter is conveyed via notifications):  (a) If the filter parameter is present, theMnS producer shall apply the constraint when constructing this output parameter. Furthermore, if the alarmAckState constraint is present, the MnS producerr shall apply that constraint as well. The filter constraint, if any, that is currently active in the notification channel is not used for the construction of this output parameter.  (b) If the filter parameter is absent, the MnS producer shall apply the filter constraint currently active in the notification channel when constructing this output parameter. If the alarmAckState constraint is present, the MnS producer shall apply that constraint as well. |
| status | M | ENUM (OperationSucceeded, OperationFailed) | If all the AlarmInformation are returned, status = OperationSucceeded.  If operation is failed, status = OperationFailed. |

The following table defines an item of alarmInformationList.

Table 11.2.1.1.3.3-2: Definition of an item of alarmInformationList

| **Parameter name** | **S** | **Matching information** | **Comment** |
| --- | --- | --- | --- |
| objectClass,  objectInstance | M | MonitoredEntity.objectClass,  MonitoredEntity.objectInstance | The MonitoredEntity is identified by the relation-AlarmedObject-AlarmInformation. |
| notificationId | M | AlarmInformation.notificationId |  |
| notificationType | M | "notifyNewAlarm"  or  "notifyChangedAlarm"  or  "notifyClearedAlarm" | The parameter carries  - notifyNewAlarm in case the alarm has not yet changed and has not yet been cleared.  - notifyChangedAlarm in case the alarm has changed but has not yet been cleared.  - notifyClearedAlarm in case the alarm has been cleared but not yet acknowledged. |
| eventTime | O | AlarmInformation.alarmRaisedTime or  AlarmInformation.alarmChangedTime or  AlarmInformation.alarmClearedTime | The parameter carries the  - alarmRaisedTime in case notificationType carries notifyNewAlarm  - alarmChangedTime in case notificationType carries notifyChangedAlarm  - alarmClearedTime in case notificationType carries notifyClearedAlarm |
| systemDN | M | -- |  |
| alarmId | M | AlarmInformation.alarmId |  |
| [objectClass],  [objectInstance] | n/a | MonitoredEntity.objectClass,  MonitoredEntity.objectInstance | Parmeter identical to the first parameter in this list, shown here to clarify all elements of AlarmInformation are present |
| [notificationId] | n/a | AlarmInformation.notificationId | Parmeter identical to the second parameter in this list, shown here to clarify all elements of AlarmInformation are present |
| alarmRaisedTime | M | AlarmInformation.alarmRaisedTime |  |
| alarmChangedTime | O | AlarmInformation.alarmChangedTime | not applicable if the severity of related alarm was not changed |
| alarmClearedTime | M | AlarmInformation.alarmClearedTime | not applicable if related alarm was not cleared |
| alarmType | M | AlarmInformation.alarmType |  |
| probableCause | M | AlarmInformation.probableCause |  |
| specificProblem | O | AlarmInformation.specificProblem |  |
| perceivedSeverity | M | AlarmInformation.perceivedSeverity |  |
| backedUpStatus | O | AlarmInformation.backedUpStatus | not applicable if related alarm is a security alarm |
| backUpObject | O | MonitoredEntity.objectInstance | The MonitoredEntity is identified by relation-BackUpObject-AlarmInformation.  Not applicable if related alarm is a security alarm |
| trendIndication | O | AlarmInformation.trendIndication | not applicable if related alarm is a security alarm |
| thresholdInfo | O | AlarmInformation.thresholdInfo | not applicable if related alarm is a security alarm |
| correlatedNotifications | O | The set of CorrelatedNotification instances related to this AlarmInformation. |  |
| stateChangeDefinition | O | AlarmInformation.stateChange | not applicable if related alarm is a security alarm |
| monitoredAttributes | O | AlarmInformation.monitoredAttributes | not applicable if related alarm is a security alarm |
| proposedRepairActions | O | AlarmInformation.proposedRepairActions | not applicable if related alarm is a security alarm |
| additionalText | O | AlarmInformation.additionalText |  |
| additionalInformation | O | AlarmInformation.additionalInformation |  |
| rootCauseIndicator | O | AlarmInformation.rootCauseIndicator |  |
| ackTime | M | AlarmInformation.ackTime | not applicable if related alarm was not acknowledged nor unacknowledged  The availability and accuracy of time carried by the time parameters in individual entries of the list (i.e. eventTime, alarmRaisedTime, alarmClearedTime and ackTime) shall be "best effort".  Reason: A Management System is not required to persistently store these times or other alarm information (as in case of synchronization information may be provided by the NE), while also some NE's do not keep these times (and a later attempt to retrieve the alarm data from the NEs will not deliver these time data). |
| ackUserId | M | AlarmInformation.ackUserId | not applicable if related alarm was not acknowledged nor unacknowledged |
| ackSystemId | O | AlarmInformation.ackSystemId | not applicable if related alarm was not acknowledged nor unacknowledged |
| ackState | M | AlarmInformation.ackState | not applicable if related alarm was not acknowledged nor unacknowledged |
| clearUserId | O | AlarmInformation.clearUserId | not applicable if related alarm was not cleared |
| clearSystemId | O | AlarmInformation.clearSystemId | not applicable if related alarm was not cleared |
| serviceUser | M | AlarmInformation.serviceUser | not applicable if related alarm is not a security alarm |
| serviceProvider | M | AlarmInformation.serviceProvider | not applicable if related alarm is not a security alarm |
| securityAlarmDetector | M | AlarmInformation.securityAlarmDetector | not applicable if related alarm is not a security alarm |
| comments | M | The set of Comment instances related to this AlarmInformation. | Not applicable if the related alarm has no related comments |

###### 11.2.1.1.3.4 Exceptions and constraints

| Exception Name | Definition |
| --- | --- |
| operation\_failed | **Condition:** Operation is failed  **Returned Information:** The output parameter status  **Exit state:** Entry State |

##### 11.2.1.1.4 notifyNewAlarm

###### 11.2.1.1.4.1 Definition

This notification is generated by the MnS producer when a new AlarmInformation is added to the AlarmList. The notification parameters depend on the alarmType and are different for non-security and security alarms.

###### 11.2.1.1.4.2 Input parameters

The notifyNewAlarm notification is defined by Table 11.2.1.1.4.2-1, if the alarmType is equal to "Communications Alarm", "Processing Error Alarm", "Environmental Alarm". "Quality Of Service Alarm" or "Equipment Alarm".

Table 11.2.1.1.4.2-1: Input parameters for notifications related to non-security alarms

| Parameter Name | Qualifier | Matching Information/ Information Type / Legal Values | Comment |
| --- | --- | --- | --- |
| objectClass | M | MonitoredEntity.objectClass | The MonitoredEntity is identified by the relation-AlarmedObject-AlarmInformation of the new AlarmInformation. |
| objectInstance | M | MonitoredEntity.objectInstance | The MonitoredEntity is identified by the relation-AlarmedObject-AlarmInformation of the new AlarmInformation. |
| notificationId | M | -- |  |
| notificationType | M | "notifyNewAlarm" |  |
| eventTime | M | AlarmInformation.alarmRaisedTime |  |
| systemDN | M | -- |  |
| alarmId | M | AlarmInformation.alarmId |  |
| alarmType | M | AlarmInformation.alarmType |  |
| probableCause | M | AlarmInformation.probableCause |  |
| perceivedSeverity | M | AlarmInformation.perceivedSeverity |  |
| specificProblem | O | AlarmInformation.specificProblem |  |
| backedUpStatus | O | AlarmInformation.backedUpStatus |  |
| backUpObject | O | MonitoredEntity.objectInstance  It carries the DN of the back up object. | The object is identified by relation-BackUpObject-AlarmInformation of the new AlarmInformation. |
| trendIndication | O | AlarmInformation.trendIndication |  |
| thresholdInfo | O | AlarmInformation.thresholdInfo |  |
| correlatedNotifications | O | The CorrelatedNotification instances related to this AlarmInformation. |  |
| stateChangeDefinition | O | AlarmInformation.stateChangeDefinition |  |
| monitoredAttributes | O | AlarmInformation.monitoredAttributes |  |
| proposedRepairActions | O | AlarmInformaton.proposedRepairActions |  |
| additionalText | O | AlarmInformation.additionalText |  |
| additionalInformation | O | AlarmInformation.additionalInformation |  |
| rootCauseIndicator | O | AlarmInformation.rootCauseIndicator |  |

###### 11.2.1.1.4.2a Input parameters for notifications related to non-security alarms

The notifyNewAlarm notification is defined by Table 11.2.1.1.4.2a-1, if the alarmType is equal to "Integrity Violation", "Operational Violation", "Physical Violation", "Security Service or Mechanism Violation" or "Time Domain Violation".

Table 11.2.1.1.4.2a-1: Input parameters for notifications related to security alarms

| Parameter Name | Qualifier | Matching Information/ Information Type / Legal Values | Comment |
| --- | --- | --- | --- |
| objectClass | M | MonitoredEntity.objectClass | The MonitoredEntity is identified by the relation-AlarmedObject-AlarmInformation of the new AlarmInformation. |
| objectInstance | M | MonitoredEntity.objectInstance | The MonitoredEntity is identified by the relation-AlarmedObject-AlarmInformation of the new AlarmInformation. |
| notificationId | M | -- |  |
| notificationType | M | "notifyNewAlarm" |  |
| eventTime | M | AlarmInformation.alarmRaisedTime |  |
| systemDN | M | -- |  |
| alarmId | M | AlarmInformation.alarmId |  |
| alarmType | M | AlarmInformation.alarmType |  |
| probableCause | M | AlarmInformation.probableCause |  |
| perceivedSeverity | M | AlarmInformation.perceivedSeverity |  |
| correlatedNotifications | O | The set of CorrelatedNotification related to this AlarmInformation. |  |
| additionalText | O | AlarmInformation.additionalText |  |
| additionalInformation | O | AlarmInformation.additionalInformation |  |
| rootCauseIndicator | O | AlarmIngormation.rootCauseIndicator |  |
| serviceUser | M | AlarmInformation.securityServiceUser | This may contain no information if the identify of the service-user (requesting the service) is not known. |
| serviceProvider | M | AlarmInformation.securityServiceProvider | This shall always identify the service-provider receiving a service request, from serviceUser, that provokes the security alarm. |
| securityAlarmDetector | M | AlarmInformation.securityAlarmDetector | This may contain no information if the detector of the security alarm is the serviceProvider. |

###### 11.2.1.1.4.3 Triggering event

11.2.1.1.4.3.1 From-state

noMatchedAlarm.

|  |  |
| --- | --- |
| Assertion Name | Definition |
| noMatchedAlarm | AlarmList does not contain an AlarmInformation that has the following properties:  Its matching-criteria-attributes values are identical to that of the newly generated network alarm and it is involved in relation-AlarmObject-AlarmInformation with the same MonitoredEntity as the one identified by the newly generated network alarm. |

11.2.1.1.4.3.2 To-state

newAlarmInAlarmList.

|  |  |
| --- | --- |
| Assertion Name | Definition |
| newAlarmInAlarmList | AlarmList contains an AlarmInformation holding information conveyed by the newly generated network alarm. This AlarmInformation is involved in relation-AlarmObject-AlarmInformation with the same MonitoredEntity as the one identified by the newly generated network alarm.  The following attributes of the AlarmInformation shall be populated with information in the newly generated alarm:  notificationId, alarmRaisedTime, alarmId, alarmType, , probableCause, perceivedSeverity.  The following attributes of the same AlarmInformation shall be populated with information of the newly generated alarm if the information is present (in the newly generated alarm) and if the attribute is supported:  specificProblem, backedUpStatus, trendIndication, thresholdInfo, stateChangeDefinition, monitoredAttributes, proposedRepairActions, additionalText, additionalInformation. |

##### 11.2.1.1.5 notifyChangedAlarm

###### 11.2.1.1.5.1 Definition

This notification is generated by the MnS producer when the perceivedSeverity of an existing AlarmInformation changes (except to the value "CLEARED").

11.2.1.1.5.2 Input parameters

| Parameter Name | S | Matching Information/ Information Type / Legal Values | Comment |
| --- | --- | --- | --- |
| objectClass | M | MonitoredEntity.objectClass | The MonitoredEntity is identified by the relation-AlarmedObject-AlarmInformation. |
| objectInstance | M | MonitoredEntity.objectInstance | The MonitoredEntity is identified by the relation-AlarmedObject-AlarmInformation. |
| notificationId | M | -- |  |
| notificationType | M | "notifyChangedAlarm" |  |
| eventTime | M | AlarmInformation.alarmChangedTime |  |
| systemDN | M | -- |  |
| alarmId | M | AlarmInformation.alarmId |  |
| alarmType | M | AlarmInformation.alarmType |  |
| probableCause | M | AlarmInformation.probableCause |  |
| perceivedSeverity | M | AlarmInformation.perceivedSeverity |  |

###### 11.2.1.1.5.3 Triggering event

11.2.1.1.5.3.1 From-state

alarmMatched AND alarmNotCleared AND alarmChanged.

|  |  |
| --- | --- |
| 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. |
| alarmNotCleared | The perceivedSeverity of the newly generated network alarm is not Cleared. |
| alarmChanged | The perceivedSeverity of the newly generated network alarm and of the matched AlarmInformation are different. |

11.2.1.1.5.3.2 To-state

informationUpdate.

|  |  |
| --- | --- |
| Assertion Name | Definition |
| informationUpdate | The AlarmInformation identified in alarmMatched in from-state has been updated according to the following rules:  - notificationId is updated;  - alarmChangedTime is updated;  - perceivedSeverity is updated;  - ackTime, ackUserId and ackSystemId are updated to contain no information;  - ackState is updated to "unacknowledged"; |

##### 11.2.1.1.6 notifyAlarmListRebuilt

###### 11.2.1.1.6.1 Definition

This notification is generated by the MnS producer when the AlarmList has been completely or partially rebuilt.

###### 11.2.1.1.6.2 Input parameters

| Parameter Name | S | Legal type | Comment |
| --- | --- | --- | --- |
| objectClass | M | -- | Identifies, together with the objectInstance parameter, the part of the alarm list that has been rebuilt.  If this paramter specifies the class of the instance carried in systemDN, then all AlarmInformation instances in the AlarmList may have been rebuilt.  If this parameter specifies some class represented by MonitoredEntity, then a subset of the AlarmInformation instances in the AlarmList may have been rebuilt. |
| objectInstance | M | -- | Identifies, together with the objetClass parameter, the part of the alarm list that has been rebuilt.  If this parameter is equal to the instance carried in systemDN, then all AlarmInformation instances in the AlarmList may have been rebuilt.  If this parameter is equal to some instance represented by MonitoredEntity, then only AlarmInformation related to this instance and its descendants may have been rebuilt.. |
| notificationId | M | -- | -- |
| notificationType | M | "notifyAlarmListRebuilt" |  |
| eventTime | M | -- | The time when the alarm list has been rebuilt. |
| systemDN | M | -- | It identifies the DN of sMnS producer. |
| reason | M | "System-NE communication error", "System restarts", "indeterminate". Other values can be added. | The reason why the system has rebuilt the AlarmList. This may carry different reasons than that carried by the immediate previous notifyPotentialFaultyAlarmList. |
| alarmListAlignmentRequirement | O | "alignmentRequired", "alignmentNotRequired". | It carries an enumeration of "alignmentRequired" and "alignmentNotRequired". |

###### 11.2.1.1.6.3 Triggering event

11.2.1.1.6.3.1 From-state

alarmListRebuilt\_0 OR alarmListRebuilt\_1.

|  |  |
| --- | --- |
| Assertion Name | Definition |
| alarmListRebuilt\_0 | MnS producer has cold-started, initialized, re-initialized or rebooted and it has initiated procedure to rebuild its AlarmList. |
| alarmListRebuilt\_1 | MnS producer loses confidence in part or whole of its AlarmList. MnS producer has initiated procedure to repair its AlarmList. |

11.2.1.1.6.3.2 To-state

alarmListRebuilt\_2.

|  |  |
| --- | --- |
| Assertion Name | Definition |
| alarmListRebuilt\_2 | MnS producer rebuilds the whole or part of AlarmList. |

##### 11.2.1.1.7 notifyCorrelatedNotificationChanged

###### 11.2.1.1.7.1 Definition

This notification is generated by the MnS producer when the set of CorrelatedNotification is created, updated or deleted.

###### 11.2.1.1.7.2 Input parameters

| Parameter Name | S | Matching Information/ Information Type / Legal Values | Comment |
| --- | --- | --- | --- |
| objectClass | M | MonitoredEntity.objectClass | The MonitoredEntity is identified by the relation-AlarmedObject-AlarmInformation. |
| objectInstance | M | MonitoredEntity.objectInstance | The MonitoredEntity is identified by the relation-AlarmedObject-AlarmInformation. |
| notificationId | M | -- |  |
| notificationType | M | "notifyCorrelatedNotificationChanged" |  |
| eventTime | M | It carries the time when the CorrelatedNotification is created, updated or deleted. |  |
| systemDN | M | -- |  |
| alarmId | M | AlarmInformation.alarmId |  |
| correlatedNotifications | M | The CorrelatedNotification instances related to this AlarmInformation. |  |
| rootCauseIndicator | O | AlarmInformation.rootCauseIndicator |  |

###### 11.2.1.1.7.3 Triggering event

11.2.1.1.7.3.1 From-state

newAlarmCorrelationInfoIsAvailable AND alarmInformationExists.

|  |  |
| --- | --- |
| Assertion Name | Definition |
| newAlarmCorrelationInfoIsAvailable | New alarm correlation information is available but not yet conveyed to any consumer. |
| alarmInformationExists | The AlarmInformation is in AlarmList. |

11.2.1.1.7.3.2 To-state

alarmCorrelatedInfoUpdated.

|  |  |
| --- | --- |
| Assertion Name | Definition |
| alarmCorrelatedInfoUpdated | The set of CorrelatedNotification network slice instances is created, updated or deleted. |

##### 11.2.1.1.8 getAlarmCount

###### 11.2.1.1.8.1 Definition

A MnS consumer invokes this operation to get the number of alarms in the alarm list. The alarms are counted separately for each perceived severity level. An input parameter allows to control which alarms are counted.

###### 11.2.1.1.8.2 Input parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Name | S | Information Type | Comment |
| filter | O | N/A | It carries a filter constraint. The operation shall apply it when counting the AlarmInformation instances in AlarmList.  Case when synchronous mode of operation is used for getAlarmList:  (a) If this parameter is present, the operation shall count the AlarmInformation instances which satisfy both (a) this filter constraint and (b) the condition set by input parameter alarmAckState.  (b) If this parameter is absent, the operation shall count all AlarmInformation instances that satisfy the condition set by input parameter alarmAckState.  Case when asynchronous mode of operation is used for getAlarmList:  (a) If this parameter is present, the operation shall count all AlarmInformation instances that satisfy this filter constraint and the condition set by input parameter alarmAckState.  (b) If this parameter is absent, the operation shall count AlarmInformation instances that satisfy (a) the filter constraint currently active in the notification channel established between the authorized MnS consumer and the MnS producer and (b) the condition set by input parameter alarmAckState. |
| alarmAckState | O | ENUM (all alarms, all active alarms, all active and acknowledged alarms, all active and unacknowledged, all cleared and unacknowledged alarms, all unacknowledged) | It carries a constraint. The operation shall apply it on AlarmInformation instances in AlarmList when counting. |

###### 11.2.1.1.8.3 Output parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Name | S | Matching Information | Comment |
| criticalCount, majorCount, minorCount, warningCount, indeterminateCount, clearedCount | M | N/A | They carry the number of AlarmInformation in AlarmList that has the following properties.  Case when synchronous mode of operation is used:  (a) The operation shall apply the constraints expressed in alarmAckState and filter to AlarmInformation instances when counting.  Case when asynchronous mode of operation is used (i.e. this output parameter is conveyed via notifications):  (a) If the filter parameter is present, the operation shall apply the constraint when counting. Furthermore, if the alarmAckState constraint is present, the operation shall apply that constraint as well. The filter constraint, if any, that is currently active in the notification channel is not used for the counting.  (b) If the filter parameter is absent, the operation shall apply the filter constraint currently active in the notification channel when counting. If the alarmAckState constraint is present, the operation shall apply that constraint as well. |
| status | M | ENUM (OperationSucceeded, OperationFailed) | If allAlarmInformationCounted is true, status = OperationSucceeded.  If operation\_failed is true, status = OperationFailed. |

###### 11.2.1.1.8.4 Pre-condition

There are no pre-conditions.

###### 11.2.1.1.8.5 Post-condition

allAlarmInformationCounted.

|  |  |
| --- | --- |
| Assertion Name | Definition |
| allAlarmInformationCounted | All AlarmInformation that satisfy the constraints expressed in input parameters filter and alarmAckState and are present in the AlarmList at the moment of this operation invocation are counted and the result returned.  All AlarmInformation in AlarmList remains unchanged as the result of this operation. |

###### 11.2.1.1.8.6 Exceptions

|  |  |
| --- | --- |
| Name | Definition |
| operation\_failed | **Condition:** the pre-condition is false or the post-condition is true.  **Returned Information:** The output parameter status.  **Exit state:** Entry state. |
| filter\_complexity\_limit | **Condition:** Operation not performed because the filter parameter is too complex.  **Returned Information**: The output parameter status.  **Exit state:** Entry state. |

##### 11.2.1.1.9 setComment

###### 11.2.1.1.9.1 Definition

A MnS consumer invokes this operation to set a comment in one or more AlarmInformation instances in AlarmList.

###### 11.2.1.1.9.2 Input parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Name | S | Information Type | Comment |
| alarmInformation ReferenceList | M | List of AlarmInformation.alarmId | It carries one or more identifiers identifying AlarmInformation instances in the AlarmList. |
| commentUserId | M | Comment.commentUserId | The Comment is identified by the relation-AlarmInformation-Comment. |
| commentSystemId | O | Comment.commentSystemId | The Comment is identified by the relation-AlarmInformation-Comment. |
| commentText | M | Comment.commentText | The Comment is identified by the relation-AlarmInformation-Comment. |

###### 11.2.1.1.9.3 Output Parameters

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Qualifier | Matching Information | Comment |
| badAlarm Information ReferenceList | M | List of pair of AlarmInformation.alarmId and the failure reason. | If allUpdated is true, it contains no information.  If someUpdated is true, then it contains identifications of AlarmInformation that are not present in AlarmList or that they are present, but AlarmInformation.comments has not changed, in contrast to authorized consumer's request. |
| status | M | ENUM( Operation succeeded, Operation failed, Operation partially failed) | If allUpdated is true, then status = OperationSucceeded.  If someUpdated is true, then status = OperationPartiallyFailed.  If exception operationFailed is raised, then status = OperationFailed. |

#### 11.2.1.2 Fault supervision data control

##### 11.2.1.2.1 acknowledgeAlarms

###### 11.2.1.2.1.1 Definition

The MnS consumer invokes this operation to acknowledge one or more alarms.

When this operation is not supported, the MnS producer shall support acknowledging alarms.

###### 11.2.1.2.1.2 Input parameters

| Parameter Name | S | Information Type / Legal Values | Comment |
| --- | --- | --- | --- |
| alarmInformationReferenceList | M | SET OF SEQUENCE {  AlarmInformation.alarmId (M)  AlarmInformation.perceivedSeverity (O)  } | It identifies the alarms to be acknowledged. If an alarm id is qualified with an optional perceived severity, the alarm shall be acknowledged only when the perceived severity in the alarm list matches the perceived severity provided in the operation request. |
| ackUserId | M | AlarmInformation.ackUserId | The identifier of the user acknowledgeding the alarm. |
| ackSystemId | O | AlarmInformation.ackSystemId | The identifier of the system where the acknowledgement request was originated. |

###### 11.2.1.2.1.3 Output parameters

| Parameter Name | S | Matching Information /  Information Type / Legal Values | Comment |
| --- | --- | --- | --- |
| badAlarm Information ReferenceList | M | SET OF SEQUENCE {  AlarmInformation.alarmId (M)  errorReason (M)  }  errorReason ::= ENUM {  UnknownAlarmId,  AcknowledgmentFailed,  WrongPerceivedSeverity  } | If all alarms are acknowledged, it contains no information.  If only some alarms are acknowledged, then it contains identifications of AlarmInformation that are  (a) present in input parameter AlarmInformationReferenceList but absent in the AlarmList (errorReason = UnknownAlarmId; or  (b) present in input parameter AlarmInformationReferenceList and present in the AlarmList but the Acknowledgement Information (see note below table) has not changed despite the consumer's request (errorReason = AcknowledgmentFailed); or  (c) present in input parameter AlarmInformationReferenceList and present in the AlarmList but the perceivedSeverity to be acknowledged has changed and/or is different in the Alarm List (ErrorReason = WrongPerceivedSeverity), applicable only if perceivedSeverity is provided. |
| status | M | ENUM {  OperationSucceeded,  OperationPartiallySucceeded,  OperationFailed  } | If all alarms acknowledged, then  status = OperationSucceeded.  If some alarms are acknowledged, then status = OperationPartiallySuceeded.  If operation failed is true, then  status = OperationFailed. |

NOTE: Acknowledgement Information is defined as the information contained in AlarmInformation.ackTime, AlarmInformation.ackUserId, AlarmInformaton.ackSystemId, AlarmInformation.ackState.

###### 11.2.1.2.1.4 Exceptions and constraints

| Exception Name | Definition |
| --- | --- |
| operation\_failed | **Condition:** Operation is failed  **Returned Information:** The output parameter status  **Exit state:** Entry State |

##### 11.2.1.2.2 unacknowledgeAlarms

###### 11.2.1.2.2.1 Definition

The MnS consumer invokes this operation to remove acknowledgement information kept in one or more AlarmInformation instances.

###### 11.2.1.2.2.2 Input parameters

| Parameter Name | S | Information Type / Legal Values | Comment |
| --- | --- | --- | --- |
| alarmInformationReferenceList | M | List of AlarmInformation.alarmId | It carries one or more identifiers identifying AlarmInformation in AlarmList. |
| ackUserId | M | AlarmInformation.ackUserId | The identifier of the user unacknowledgeding the alarm. |
| ackSystemId | O | AlarmInformation.ackSystemId | The identifier of the system where the acknowledgement request was originated. |

###### 11.2.1.2.2.3 Output parameters

| Parameter Name | Support Qualifier | Matching Information /  Information Type / Legal Values | Comment |
| --- | --- | --- | --- |
| badAlarmInformation  ReferenceList | M | SET OF SEQUENCE {  AlarmInformation.alarmId (M)  errorReason (M)  }  errorReason ::= ENUM {  UnknownAlarmId,  AcknowledgmentFailed,  WrongPerceivedSeverity,  } | If all alarms are acknowledged, it contains no information.  If only some alarms are acknowledged, then it contains identifications of AlarmInformation that are  (a) present in input parameter AlarmInformationReferenceList but absent in the AlarmList (errorReason = UnknownAlarmId; or  (b) present in input parameter AlarmInformationReferenceList and present in the AlarmList but the Acknowledgement Information (see note below table) has not changed despite the consumer's request (errorReason = AcknowledgmentFailed); or  (c) present in input parameter AlarmInformationReferenceList and present in the AlarmList but the perceivedSeverity to be acknowledged has changed and/or is different in the Alarm List (ErrorReason = WrongPerceivedSeverity), applicable only if perceivedSeverity is provided. |
| status | M | ENUM {  OperationSucceeded,  OperationPartiallySucceeded,  OperationFailed | If all alarms acknowledged, then  status = OperationSucceeded.  If some alarms are acknowledged, then  status = OperationPartiallySuceeded.  If operation failed is true, then  status = OperationFailed. |

NOTE: Acknowledgement Information is defined as the information contained in AlarmInformation.ackTime, AlarmInformation.ackUserId, AlarmInformaton.ackSystemId, AlarmInformation.ackState.

###### 11.2.1.2.2.4 Exceptions and constraints

| Exception Name | Definition |
| --- | --- |
| Operation\_failed | **Condition:** Operation is failed  **Returned Information:** The output parameter status  **Exit state:** Entry State |

##### 11.2.1.2.3 clearAlarms

###### 11.2.1.2.3.1 Definition

The authorized consumer invokes this operation to clear one or more AlarmInformation instances in AlarmList. For example, this operation can be used to support the manual clearing of the ADMC (automatic detection and manual clearing, see also 3GPP TS 32.111-1 [3]) alarms.

###### 11.2.1.2.3.2 Input parameters

| Parameter Name | S | Information Type / Legal Values | Comment |
| --- | --- | --- | --- |
| alarmInformation ReferenceList | M | List of AlarmInformation.alarmId | It carries one or more identifiers identifying AlarmInformation instances in the AlarmList. |
| clearUserId | M | AlarmInformation.clearUserId | It identities the user clearing the alarm. |
| clearSystemId | O | AlarmInformation.clearSystemId | It identifies the authorized consumer. It may be absent implying that consumer does not wish this information be known to the MnS producer. |

###### 11.2.1.2.3.3 Output parameters

| Parameter Name | S | Matching Information /  Information Type / Legal Values | Comment |
| --- | --- | --- | --- |
| badAlarmInformation ReferenceList | M | List of pair of AlarmInformation.alarmId and the failure reason. | If all alarms are cleared, it contains no information.  If some alarms are cleared, then it contains identifications of AlarmInformation that are not present in AlarmList or that are present in AlarmList but remain unchanged, in contrast to consumer's request. |
| status | M | ENUM( OperationSucceeded, OperationFailed, OperationPartiallySucceeded) | If all alarms are cleared, then status = OperationSucceeded.  If some alarms are cleared, then status = OperationPartiallySucceeded.  If operation is failed, then status = OperationFailed. |

###### 11.2.1.2.3.4 Exceptions and constraints

| Exception Name | Definition |
| --- | --- |
| operation\_failed | **Condition:** Operation is failed  **Returned Information:** The output parameter status  **Exit state:** Entry State |

##### 11.2.1.2.4 notifyClearedAlarm

###### 11.2.1.2.4.1 Definition

This notification is generated by the MnS producer when the perceivedSeverity of an existing AlarmInformation changes to "CLEARED".

###### 11.2.1.2.4.2 Input parameters

| Parameter Name | S | Matching Information/ Information Type / Legal Values | Comment |
| --- | --- | --- | --- |
| objectClass | M | MonitoredEntity.objectClass |  |
| objectInstance | M | MonitoredEntity.objectInstance |  |
| notificationId | M | -- |  |
| notificationType | M | "notifyClearedAlarm" |  |
| eventTime | M | AlarmInformation.alarmClearedTime |  |
| systemDN | M | -- |  |
| alarmId | M | AlarmInformation.alarmId |  |
| alarmType | M | AlarmInformation.alarmType |  |
| probableCause | M | AlarmInformation.probablaCause |  |
| perceivedSeverity | M | AlarmInformation.perceivedSeverity | Value shall be "CLEARED" |
| correlatedNotifications | O | The CorrelatedNotification instances related to this AlarmInformation. | This parameter contains references to other AlarmInformation instances whose perceivedSeverity levels are cleared as well. In this way, the perceivedSeverity level of multiple AlarmInformation instances can be cleared by one notification. |
| clearUserId | O | AlarmInformation.clearUserId | This parameter shall be present and contain valid information if the AlarmInformation is cleared by a clearAlarms operation request. |
| clearSystemId | O | AlarmInformation.clearSystemId | This parameter is present if clearUserId is present and if AlarmInformation.clearSystemId contains valid information. |

###### 11.2.1.2.4.3 Triggering event

11.2.1.2.4.3.1 From-state

alarmMatchedAndCleared OR clearedByProvider.

|  |  |
| --- | --- |
| Assertion Name | Definition |
| alarmMatchedAndCleared | The matching-criteria-attributes of the newly generated network alarm have values that are identical (matched) with ones in one AlarmInformation in AlarmList and the perceivedSeverity of the matched AlarmInformation is not Cleared  AND  The perceivedSeverity of the newly generated network alarm is cleared. |
| clearedByProvider | Reception of a valid clearAlarms operation that identifies the subject AlarmInformation instances. This triggering event shall occur regardless of the perceivedSeverity state of the identified AlarmInformation instances. |

11.2.1.2.4.3.2 To-state

alarmInformationCleared\_1 OR alarmInformationCleared\_2.

|  |  |
| --- | --- |
| Assertion Name | Definition |
| alarmInformationCleared\_1 | Case if From-state is alarmMatchedAndCleared:  The following attributes of the subject AlarmInformation are updated:  notificationId, perceivedSeverity (updated to Cleared), alarmClearedTime. |
| alarmInformationCleared\_2 | Case if From-state is clearedByProvider:  The following attributes of the subject AlarmInformation are updated:  notificationId, alarmClearedTime, perceivedSeverity (updated to CLEARED), alarmClearedUserId, alarmClearedSystemId. |

##### 11.2.1.2.5 notifyAckStateChanged

###### 11.2.1.2.5.1 Definition

This notification is generated by the MnS producer when a the acknowledgement state of an alarm changes from "UNACKNOWLEDGED" to "ACKNOWLEDGED" or back from "ACKNOWLEDGED" to "UNACKNOWLEDGED".

###### 11.2.1.2.5.2 Input parameters

| Parameter Name | S | Matching Information/ Information Type / Legal Values | Comment |
| --- | --- | --- | --- |
| objectClass | M | MonitoredEntity.objectClass |  |
| objectInstance | M | MonitoredEntity.objectInstance |  |
| notificationId | M | -- |  |
| notificationType | M | "notifyAckStateChanged" |  |
| eventTime | M | AlarmInformation.ackTime |  |
| systemDN | M | -- |  |
| alarmId | M | AlarmInformation.alarmId |  |
| alarmType | M | AlarmInformation.alarmType |  |
| probableCause | M | AlarmInformation.probableCause |  |
| perceivedSeverity | M | AlarmInformation.perceivedSeverity |  |
| ackState | M | AlarmInformation.ackState |  |
| ackUserId | M | AlarmInformation.ackUserId | The identifier of the user who acknowledged or unacknowledged the alarm. |
| ackSystemId | O | AlarmInformation.ackSystemId | The identifier of the system where the acknowledgement or unacknowledgement request was originated. |

###### 11.2.1.2.5.3 Triggering event

11.2.1.2.5.3.1 From-state

ackedByConsumer OR ackedByProvider AND alarmInformationExists.

|  |  |
| --- | --- |
| Assertion Name | Definition |
| ackedByConsumer | Reception of an acknowledgeAlarms operation and a subsequent operation success return. |
| ackedByProvider | Reception of a local (non-standard) acknowlegeAlarms equivalent operation and a subsequent operation success return. |
| alarmInformationExists | The AlarmInformation exists in AlarmList. |

11.2.1.2.5.3.2 To-state

alarmAckStateHasChanged.

|  |  |
| --- | --- |
| Assertion Name | Definition |
| alarmAckStateHasChanged | The AlarmInformation.ackState of the AlarmInformation identified by from-state assertion alarmInformationExists have been updated. Specifically, the following attributes of the subject AlarmInformation are updated:  -- notificationId, ackTime, ackUserId, ackState, ackSystemId. |

##### 11.2.1.2.6 notifyComments

###### 11.2.1.2.6.1 Definition

This notification is generated by the MnS producer when a Comment instance is added to an AlarmInformation instance in the AlarmList.

A MnS producer shall support this notification if it supports the operation setComment.

###### 11.2.1.2.6.2 Input parameters

| Parameter Name | S | Matching Information/ Information Type / Legal Values | Comment |
| --- | --- | --- | --- |
| objectClass | M | MonitoredEntity.objectClass |  |
| objectInstance | M | MonitoredEntity.objectInstance |  |
| notificationId | M | -- |  |
| notificationType | M | "notifyComments" |  |
| eventTime | M | Comment.commentTime |  |
| systemDN | M | -- |  |
| alarmId | M | AlarmInformation.alarmId |  |
| alarmType | M | AlarmInformation.alarmType |  |
| probableCause | M | AlarmInformation.probableCause |  |
| perceived Severity | M | AlarmInformation.perceivedSeverity |  |
| comments | M | The Comment instances related to this AlarmInformation. |  |

###### 11.2.1.2.6.3 Trigger event

11.2.1.2.6.3.1 From-state

commentedByServiceprovider OR commentedByServiceprovider AND alarmInformationExists.

|  |  |
| --- | --- |
| Assertion Name | Definition |
| commentedByServiceprovider | Reception of a setComment operation and a subsequent operation success return. |
| commentedByServiceprovider | Reception of a local (non-standard) setComment equivalent operation and a subsequent operation success return. |
| alarmInformationExists | The AlarmInformation is in AlarmList. |

11.2.1.2.6.3.2 To-state

commentInserted.

|  |  |
| --- | --- |
| Assertion Name | Definition |
| commentInserted | One Comment has been created and it is involved in a relationship with the AlarmInformation identified by from-state assertion alarmInformationExists. The following attributes of the newly created Comment instance shall be populated:  commentTime, commentText, commentUserId and commentSystemId. |

##### 11.2.1.2.7 notifyPotentialFaultyAlarmList

###### 11.2.1.2.7.1 Definition

This notification is generated by the MnS producer when the MnS producer looses confidence in the integrity of its alarm list.

The MnS producer may then rebuilt the faulty alarm list. When the alarm List is rebuilt or confidence in the existing alarm list is re-established the MnS producer may generate a notifyAlarmListRebuilt notification.

The parameters objectClass and objectInstance are used to specify if the complete alarm list is unreliable or only parts thereof.

The MnS consumer behaviour, on reception of this notifyPotentialFaultyAlarmList notification, is not specified. The authorized consumer behaviour is considered not essential for the specification of the interface itself. However, the following are recommended actions the uthorized consumer should take, in case it receives this notification.

1) The uthorized consumer should not perform any task requiring the integrity of the AlarmInformation identified as faulty or unreliable by the subject notification.

2) The uthorized consumer should not invoke operations that require integrity of the AlarmList such as getAlarmList., acknolwedgeAlarms operations.

###### 11.2.1.2.7.2 Input parameters

| Parameter Name | S | Matching Information/ Information Type / Legal Values | Comment |
| --- | --- | --- | --- |
| objectClass | M | It identifies  the class of the instance identified by systemDN or  the class of MonitoredEntity. | Identifies, together with the objectInstance parameter, the part of the alarm list that is not reliable.  If this paramter specifies the class of the instance carried in systemDN, then all AlarmInformation instances in the AlarmList may not be reliable.  If this parameter specifies some class represented by MonitoredEntity, then a subset of the AlarmInformation instances in the AlarmList is not reliable. |
| objectInstance | M | It identifies  the instance identified by systemDN or  an instance of MonitoredEntity. | Identifies, together with the objetClass parameter, the part of the alarm list that may not be reliable.  If this parameter is equal to the instance carried in systemDN, then all AlarmInformation instances in the AlarmList may not be reliable.  If this parameter is equal to some instance represented by MonitoredEntity, then only AlarmInformation related to this instance and its descendants may not be reliable. |
| notificationId | M | -- |  |
| notificationType | M | "notifyPotentialFaultyAlarmList" |  |
| eventTime | M | -- | Time when the MnS producer lost confidence in the integrity of the alarm list |
| systemDN | M | -- |  |
| reason | M | "serviceprovider-NE communication error", " serviceprovider restarts", "indeterminate". Other values can be added. | Reason why the MnS producer has to rebuild its AlarmList. |

###### 11.2.1.2.7.3 Trigger event

11.2.1.2.7.3.1 From-state

faultyAlarmListDetected.

|  |  |
| --- | --- |
| Assertion Name | Definition |
| faultyAlarmListDetected | MnS producer detects faults in part or whole of its AlarmList. |

11.2.1.2.7.3.2 To-state

faultyAlarmList

|  |  |
| --- | --- |
| Assertion Name | Definition |
| faultyAlarmList | MnS producer initiates the AlarmList rebuild process. |

##### 11.2.1.2.8 notifyChangedAlarmGeneral

###### 11.2.1.2.8.1 Definition

This notification is generated by the MnS producer when one or more of the following attributes of an AlarmInformation instance in the AlarmList changes its value: perceivedSeverity, backedUpStatus, backUpObject, trendIndication, thresholdInfo, stateChangeDefinition, monitoredAttributes, proposedRepairActions, additionalText, additionalInformation, serviceUser, serviceProvider or securityAlarmDetector. From the attributes listed above, only those that changed value shall be included in the notification.

The notification parameters depend on the alarmType and are different for non-security and security alarms.

###### 11.2.1.2.8.2 Input parameters for notifications related to non-security alarms

The notifyChangedAlarmGeneral notification is defined by Table 11.2.1.2.8.2-1, if the alarmType is equal to "Communications Alarm", "Processing Error Alarm", "Environmental Alarm",, "Quality Of Service Alarm" or "Equipment Alarm".

Table 11.2.1.2.8.2-1: Input parameters for notifications related to non-security alarms

| Parameter Name | S | Matching Information/ Information Type / Legal Values | Comment |
| --- | --- | --- | --- |
| objectClass | M | MonitoredEntity.objectClass |  |
| objectInstance | M | MonitoredEntity.objectInstance |  |
| notificationId | M | -- |  |
| notificationType | M | "notifyChangedAlarmGeneral" |  |
| eventTime | M | AlarmInformation.alarmChangedTime |  |
| systemDN | M | -- |  |
| alarmId | M | AlarmInformation.alarmId |  |
| alarmType | M | AlarmInformation.alarmType |  |
| probableCause | O | AlarmInformation.probableCause |  |
| specificProblem | O | AlarmInformation.specificProblem |  |
| perceivedSeverity | O | AlarmInformation.perceivedSeverity |  |
| backedUpStatus | O | AlarmInformation.backedUpStatus |  |
| backUpObject | O | MonitoredEntity.objectInstance | The DN of the back up object. The object is identified by relation-BackUpObject-AlarmInformation of the new AlarmInformation. |
| trendIndication | O | AlarmInformation.trendIndication |  |
| thresholdInfo | O | AlarmInformation.thresholdInfo |  |
| correlatedNotifications | O | Set of CorrelatedNotification related to this AlarmInformation. |  |
| stateChangeDefinition | O | AlarmInformation.stateChange |  |
| monitoredAttributes | O | AlarmInformation.monitoredAttributes |  |
| proposedRepairActions | O | AlarmInformaton.proposedRepairActions |  |
| additionalText | O | AlarmInformation.additionalText |  |
| additionalInformation | O | AlarmInformation.additionalInformation |  |
| rootCauseIndicator | O | alarmInformation.rootCauseIndicator |  |
| changedAlarmAttributes | O | LIST OF SEQUENCE <AttributeName, OldAttributeValue> | The changed alarm attributes (name/value pairs) (with old values). |

###### 11.2.1.2.8.3 Input parameters for notifications related to security alarm

The notifyChangedAlarmGeneral notification is defined by Table 11.2.1.1.4.2a-1, if the alarmType is equal to "Integrity Violation", "Operational Violation", "Physical Violation", "Security Service or Mechanism Violation" or "Time Domain Violation".

Table 11.2.1.2.8.3-1: Input parameters for notifications related to security alarms

| Parameter Name | S | Matching Information/ Information Type / Legal Values | Comment |
| --- | --- | --- | --- |
| objectClass | M | MonitoredEntity.objectClass |  |
| objectInstance | M | MonitoredEntity.objectInstance |  |
| notificationId | M | -- |  |
| notificationType | M | "notifyChangedAlarmGeneral". |  |
| eventTime | M | AlarmInformation.alarmChangedTime |  |
| systemDN | M | -- |  |
| alarmId | M | AlarmInformation.alarmId |  |
| alarmType | M | AlarmInformation.alarmType |  |
| probableCause | O | AlarmInformation.probableCause |  |
| perceivedSeverity | O | AlarmInformation.perceivedSeverity |  |
| correlatedNotifications | O | Set of CorrelatedNotification related to this AlarmInformation. |  |
| additionalText | O | AlarmInformation.additionalText |  |
| additionalInformation | O | AlarmInformation.additionalInformation |  |
| rootCauseIndicator | O | alarmInformation.rootCauseIndicator |  |
| serviceUser | M | AlarmInformation.serviceUser | This may contain no information if the identify of the service-user (requesting the service) is not known. |
| serviceProvider | M | AlarmInformation.serviceProvider | This shall always identify the service-provider receiving a service request, from serviceUser, that provokes the security alarm. |
| securityAlarmDetector | M | AlarmInformation.securityAlarmDetector | This may contain no information if the detector of the security alarm is the serviceProvider. |
| changedAlarmAttributes | O | LIST OF SEQUENCE <AttributeName, OldAttributeValue> | The changed alarm attributes (name/value pairs) (with old values). |

###### 11.2.1.2.8.4 Trigger event

11.2.1.2.8.4.1 From-state

alarmMatched AND alarmNotCleared AND alarmChanged.

|  |  |
| --- | --- |
| 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. |
| alarmChanged | One or more of perceivedSeverity, backedUpStatus, backUpObject, trendIndication, thresholdInfo, stateChangeDefinition, monitoredAttributes, proposedRepairActions, additionalText, additionalInformation, serviceUser, serviceProvider or securityAlarmDetector of the newly generated network alarm and of the matched AlarmInformation are different. |

11.2.1.2.8.4.2 To-state

informationUpdate.

|  |  |
| --- | --- |
| Assertion Name | Definition |
| informationUpdate | The AlarmInformation identified in alarmMatched in from-state has been updated according to the following rules: perceivedSeverity, backedUpStatus, backUpObject, trendIndication, thresholdInfo, stateChangeDefinition, monitoredAttributes, proposedRepairActions, additionalText, additionalInformation, serviceUser, serviceProvider or securityAlarmDetector is updated;  notificationId is updated;  alarmChangedTime is updated;  ackTime, ackUserId and ackSystemId are updated to contain no information;  ackState is updated to "unacknowledged"; |

|  |
| --- |
| **End of Change** |