Bronze use case: Health-Check

In the Bronze release, the following use case will be supported by SMO supporting O1 interface.

Health-Check Use Case (entire use case)

O1 Managed Function (O-CU/O-DU/O-RU) Health-Check (Flow #4) (focus on O1 interface)

For the part "O1 MF health status retrieval" a common **yang module** is required. O-RAN Alliance WG4 developed a yang module for this purpose called "o-ran-fm.yang".

In order to give all the O-RAN Alliance working groups and O-RAN-SC project the flexibility to define the alarms/faults needed for their domain, yang provides mechanisms to add items to an extensible enum - called identities.

Those yang capabilities will be exposed by NetConf-Servers to NetConf-Client(s) on SMO in the NetConf Hello-Message so that SMO can learn the supported alarms by each of the different components.

Instructions how to create such yang are described in https://tools.ietf.org/html/rfc8632 - see "Appendix A. Vendor-Specific Alarm Types Example" and think of "O-RAN-Managed-Function-Specific" instead of "Vendor-Specific".

Related **YANG** data modules in scp/oam/modeling.

Matti Hiltunen Thoralf Czichy user-a934b Salvatore Scarpina Zhimin Yuan : Please comment 😉

AlarmList

Under discussion with O-RAN Alliance - the alarm-list compared to RFC8632

```
...

ro alarm-list
~~| +--ro number-of-alarms? yang:gauge32~~
~~| +--ro last-changed? yang:date-and-time~~
| +--ro alarm* [resource alarm-type-id alarm-type-
qualifier]
| | +--ro resource resource
| | +--ro alarm-type-id alarm-type-id
~~| | +--ro alarm-type-qualifier alarm-type-qualifier~~
| | +--ro alt-resource* resource
| | +--ro related-alarm* [resource alarm-type-id
alarm-type-qualifier] {alarm-correlation}?
| | | +--ro resource -> /alarms/alarm-list/alarm
/resource
| | | +--ro alarm-type-id -> /alarms/alarm-list/alarm
[resource=current()/../resource]/alarm-type-id
| | | +--ro alarm-type-qualifier -> /alarms/alarm-list
/alarm[resource=current()/../resource][alarm-type-
id=current()/../alarm-type-id]/alarm-type-qualifier
| | +--ro impacted-resource* resource {service-impact-
analysis}?
| | +--ro root-cause-resource* resource {root-cause-
analysis}?
| | +--ro time-created yang:date-and-time
~~| | +--ro is-cleared boolean~~
~~| | +--ro last-raised yang:date-and-time~~
~~| | +--ro last-changed yang:date-and-time~~
| | +--ro perceived-severity severity
```

Mapping YANG to VES

VES format:

https://gerrit.onap.org/r/gitweb?p=dcaegen2/collectors/ves.git;a=blob;f=etc/CommonEventFormat_30.0.1.json;h=49c7f9320b7aad6dee22ccb4fc401c81521142bd;hb=7547ad7dbd45fff19c948da675ab09d72775f973

VES	NetConf Alarm
ves:eventSeverity	`severity`
ves:eventSeverity = NORMAL	is-cleared = true
ves:faultFields:alarmCondition	yang-model namespace + ":" + `alarm-type-id`
ves:faultFields: alarmAdditionalInformation	description of `alarm-type-id`

```
| | +--ro alarm-text alarm-text
| | +--ro status-change* [time] {alarm-history}?
| | | +--ro time yang:date-and-time
| | | +--ro perceived-severity severity-with-clear
| | | +--ro alarm-text alarm-text
| | +--ro operator-state-change* [time] {operator-
actions}?
| | | +--ro time yang:date-and-time
| | | +--ro operator string
| | | +--ro state operator-state
| | | +--ro text? string
| | +--x set-operator-state {operator-actions}?
| | | +--w input
| | | +--w state writable-operator-state
| | | +--w text? string
| | +--n operator-action {operator-actions}?
| | +--  time yang:date-and-time
| | +--  operator string
| | +--  state operator-state
| | +--  text? string
| +--x purge-alarms
| | +--w input
| | | +--w alarm-clearance-status enumeration
| | | +--w older-than!
| | | | +--w (age-spec)?
| | | | +--:(seconds)
| | | | | +--w seconds? uint16
| | | | +--:(minutes)
| | | | | +--w minutes? uint16
| | | | +--:(hours)
| | | | | +--w hours? uint16
| | | | +--:(days)
| | | | | +--w days? uint16
| | | | +--:(weeks)
| | | | +--w weeks? uint16
| | | +--w severity!
| | | | +--w (sev-spec)?
| | | | +--:(below)
| | | | | +--w below? severity
| | | | +--:(is)
| | | | | +--w is? severity
| | | | +--:(above)
| | | | +--w above? severity
| | | +--w operator-state-filter! {operator-actions}?
| | | +--w state? operator-state
| | | +--w user? string
| | +--ro output
| | +--ro purged-alarms? uint32
| +--x compress-alarms {alarm-history}?
| +--w input
| | +--w resource? resource-match
| | +--w alarm-type-id? -> /alarms/alarm-list/alarm
/alarm-type-id
| | +--w alarm-type-qualifier? -> /alarms/alarm-list
/alarm/alarm-type-qualifier
| +--ro output
| +--ro compressed-alarms? uint32

...
```