

# A YANG Data Model for Optical Performance Monitoring

CCAMP WG, IETF115

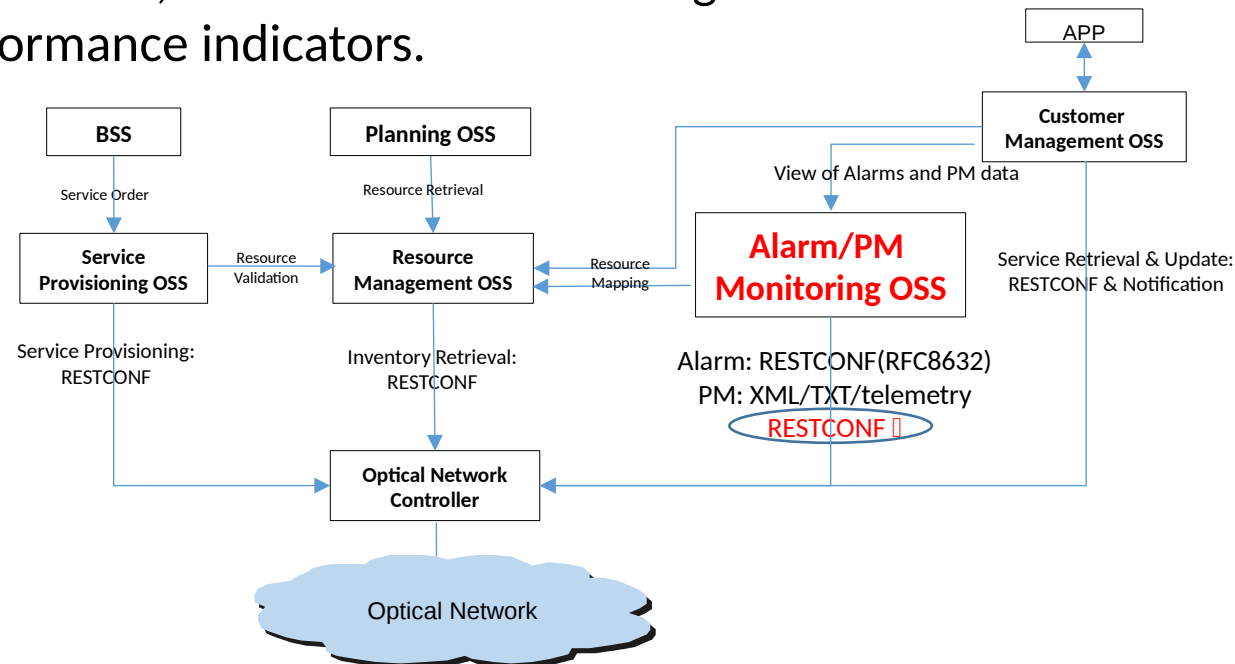
**draft-yu-performance-monitoring-yang-00**

**Author:**

Chaode Yu (yuhaode@huawei.com)

# Motivation of This Model

- Performance monitoring is a basic function of network maintenance. A unified protocol is preferred for PM, e.g. RESTCONF;
- Traditional PM interfaces are operated on network resources such as NE, board, port, channel, etc. Operators prefer to inherit existing PM functions without additional investment.
- The existing PM data models in IETF are either not network resource-based.
- The target of this draft is to specify a data model for performance monitoring based on traditional interfaces, such as TMF MTOSI, and cover all the existing functionalities. And this data model should be generic for various performance indicators.



# Requirements and Interfaces Defined in TMF

## 3.1 Business Requirements

The following business requirements are stated:

R_TMF518_RPM_BR_0001	The Interface shall support the retrieval of current and historical performance measurements for network resources.
Source	TMF518_RPM, Version 1.0
R_TMF518_RPM_BR_0002	The Interface shall support the distribution of Threshold Crossing Alerts (TCAs) to subscribed OSs.
Source	TMF518_RPM, Version 1.0
R_TMF518_RPM_BR_0003	The Interface shall support the control of performance monitoring in the network. This includes PM control, e.g., the enabling and disabling of PM collection and TCA control, e.g., the enabling and disabling of TCA generation.
Source	TMF518_RPM, Version 1.0

Service Interfaces	Operations
<b>PerformanceManagementControl</b>	
	clearPerformanceMonitoringData
	disablePerformanceMonitoringData
	enablePerformanceMonitoringData
<b>PerformanceManagementRetrieval</b>	
	getAllCurrentPerformanceMonitoringData
	getAllPerformanceMonitoringPoints
	getHistoryPerformanceMonitoringData
	getHoldingTime
	getMePerformanceMonitoringCapabilities
	getProfileAssociatedTerminationPoints
	getPerformanceMonitoringDataIterator
	getPerformanceMonitoringPointsIterator
<b>ThresholdCrossingAlertControl</b>	
	createTcaParameterProfile
	deleteTcaParameterProfile
	disableThresholdCrossingAlert
	enableThresholdCrossingAlert
	getAllTcaParameterProfiles
	getTcaParameterProfile
	getTcaParameterProfilesIterator
	getTcaTpParameter
	setTcaParameterProfile
	setTcaTpParameter

—According to [TMF518](#) DDP

The main functionality supported by our data model shall include:

- Control performance monitoring task as per network resource
- Configure TCA attributes and raise TCA notifications
- Retrieve current and historical performance data and performance monitoring capabilities

# Performance Monitoring Task Control

Interfaces defined for PM control in TMF:

- **clearPerformanceMonitoringData:** This operation clears (reset) the PM registers for a list of Measurement Points.
- **disablePerformanceMonitoringData:** This operation disables (turns off) the PM data collection for a list of measurement points.
- **enablePerformanceMonitoringData:** This operation enables (turns on) the PM data collection for a list of measurement points.

```
module: ietf-performance-monitoring
  +--rw monitor-tasks ①
    +--rw monitor-task* [resource]
      +--rw resource ② union
      +--rw resource-type? identityref
      +--rw task-name? string
      +--rw task-status? ③ enumeration
      +--rw task-cfg
        +--rw period? identityref
        +--rw indicators
          +--rw indicator* [indicator-name]
            +--rw indicator-name string
            +--rw tca
```

```
leaf resource {
  description "the identifier of network resource which is monitored.";
  type union {
    type instance-identifier {
      require-instance false;
    }
    type yang:object-identifier; ②
    type string;
    type yang:uuid;
  }
}
```

Data model in our draft

- ① A “PM task” concept is introduced to control performance monitoring. The task is running as per resource which is similar to the measurement point in TMF. If the client clears performance monitoring data, this PM task can then be deleted.
- ② The same resource definition of RFC8632 is reused - Generic for different types of measurement point, for example node, board, port, etc.
- ③ The task-status is used to enable/disable PM data collection.

# Performance Monitoring Data Retrieval

Some interfaces defined for PM data retrieval in TMF:

- **getAllCurrentPerformanceMonitoringData**: Retrieves current PM data for a list of measurement points
- **getHistoryPerformanceMonitoringData**: Retrieves historical PM data for a list of measurement points

```
+---x get-all-current-pm-data
|
| +--ro input
| | +--ro resources*   leafref
| +--ro output
|   +--ro pm-data
|     +--ro pm-data-list* [resource]
|       +--ro resource           leafref
|       +--ro collect-time?      yang:date-and-time
|       +--ro resource-type?     identityref
|       +--ro indicator-data
|         +--ro indicator-data-list* [indicator-name]
|           +--ro indicator-name      string
|           +--ro indicator-value?    string
|           +--ro indicator-value-unit? string
```

```
+---x get-history-pm-data
|
| +--ro input
| | +--ro resources*   leafref
| | +--ro start-time?  yang:date-and-time
| | +--ro end-time?    yang:date-and-time
| +--ro output
|   +--ro pm-data
|     +--ro pm-data-list* [resource]
|       +--ro resource           leafref
|       +--ro collect-time?      yang:date-and-time
|       +--ro resource-type?     identityref
|       +--ro indicator-data
|         +--ro indicator-data-list* [indicator-name]
|           +--ro indicator-name      string
|           +--ro indicator-value?    string
|           +--ro indicator-value-unit? string
```

- There is a great amount of PM data that changes frequently most of the time.
- It is inappropriate to define them as state data in the model, which can result in a lot of useless data saved in the data store. Not easy either for client to retrieve data by RESTCONF.
- Instead, two RPC operations are defined to support these PM data retrieval.

## Next Step

- Investigate how to cover TCA functions based on current data model
- Investigate how to cover the left PM retrieval interfaces based on current data model
- Call for interest & joint contribution

Thank You ☐