

Data Manifest for Streaming Telemetry

[draft-claise-opsawg-collected-data-manifest-00](#)

B. Claise (Huawei), J. Quilbeuf (Huawei), D.
Lopez (Telefonica), T. Graf (Swisscom)

IETF 112, OPSAWG

Issues

- Per-node capability discovery
 - YANG Modules describing Capabilities for Systems and Datastore Update Notifications, [draft-ietf-netconf-notification-capabilities](#) (RFC editor queue)
 - YANG Instance Data File Format, [draft-ietf-netmod-yang-instance-file-format-08](#) (RFC editor queue)
 - Per-Node Capabilities for Optimum Operational Data Collection , [draft-claise-netconf-metadata-for-collection-01](#)
- But how were data actually metered, under which circumstances?

Issues

- Value “42” is in the data collection system, what does it mean?
 - Data collected via telemetry is stored into xml or json object, sometimes as a single value
 - The bare minimum is to link the corresponding YANG object
 - Is this sufficient?
- A posteriori, it’s hard to reuse the values without context:
 - Collection period used?
 - Polling on on-change?
 - Did we have the old/new router OS, with/without the bug?

Proposal

- 2 YANG models for storing the context:
 - Platform manifest
 - Data manifest
- MUST be streamed all with the data and stored along with the collected data.
- In case the data are moved to different place (typically a database), the data manifest MUST follow the collected data.
- The data manifest MUST be encoded with the YANG instance data file format

Platform Manifest

- Elements needed to fully identify the platform:
 - Hardware model
 - OS type and version
 - Set of available YANG models and deviations: reuse YANG library module (RFC8525)
- Aligned with YANG catalog draft
I-D.clacla-netmod-model-catalog

Update when: platform changes (i.e. at reboot)

Platform Manifest

```
module: ietf-collected-data-platform-manifest
+--rw platform
  +--rw platform?          string
  +--rw software-version?  string
  +--rw software-flavor?  string
  +--rw os-version?       string
  +--rw os-type?         string
  +--rw module-set* [name]
    +--rw name              string
    +--rw module* [name]
      | +--rw name          yang:yang-identifier
      | +--rw revision?    revision-identifier
      | +--rw namespace    inet:uri
      | +--rw location*    inet:uri
      | +--rw submodule* [name]
      | | +--rw name        yang:yang-identifier
      | | +--rw revision?  revision-identifier
      | | +--rw location*  inet:uri
      | +--rw feature*     yang:yang-identifier
      | +--rw deviation*   -> ../../module/name
  +--rw import-only-module* [name revision]
    +--rw name              yang:yang-identifier
    +--rw revision          union
    +--rw namespace         inet:uri
    +--rw location*         inet:uri
    +--rw submodule* [name]
      +--rw name            yang:yang-identifier
      +--rw revision?      revision-identifier
      +--rw location*      inet:uri
```

Aligned with YANG catalog draft
I-D.clacla-netmod-model-catalog

Reusing YANG library module
(RFC8525)

Data Manifest

- Elements needed to properly interpret the received data
- Key is the sensor path
- Associated information includes:
 - Whether polling or on-change (event-driven) mode is used
 - Whether suppress-redundancy is used
 - The requested collection period (valid only for polling)
 - The current collection period (valid only for polling)

Update when: collection condition changes:

- New subscription
- Collection period is adjusted based on CPU availability

Data Manifest

```
module: ietf-collected-data-manifest
  +--rw data-collection
    +--rw mdt-collection-item* [sensor-path]
      +--rw sensor-path          string
      +--rw requested-period?    int64
      +--rw current-period?     int64
      +--rw on-change?          boolean
      +--rw suppress-redundancy? boolean
```


Improvements Identified

- Clarify how the data manifest should be collected (telemetry makes sense)
 - Acknowledgement
 - Update frequency
 - Per subscription ID instead of sensor-path -> update YANG model
- Include the source of data somewhere and notably some self-assertiveness

Open Questions

- How to map the data manifest to the collected data?
- If I stream from a line card, do we need the line card information?
- Do we want to handle data manifest for non-telemetry values ?

Conclusion

- We propose to normalize the way contextual information about the data collection
- Should be saved with the data
- Enable later reuse
- Feedback, suggestions, issues, PRs:
 - <https://github.com/JeanQuilbeufHuawei/draft-collected-data-manifest> (updated in « Additional Resources » in datatracker)
 - Mailing list