A Data Manifest for Contextualized Telemetry Data

draft-claise-opsawg-collected-data-manifest-05

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

IETF 115, OPSAWG

Goal & Problem Statement

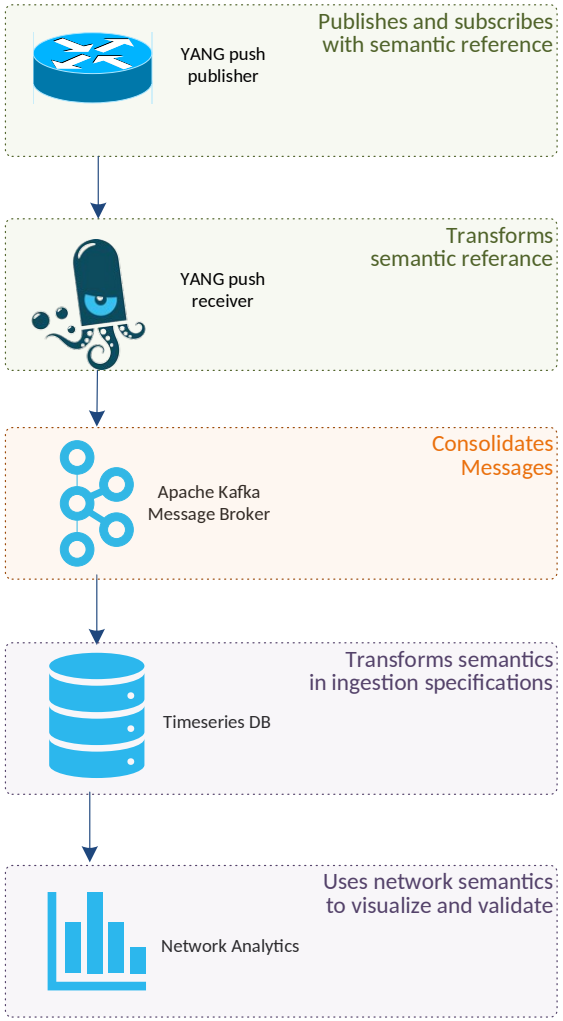
- Disclaimer: did not update the draft as much as we wanted
- Background: Multiple presentations/meetings about keeping the semantic in data collection
  - YANG/CBOR Kafka side meeting on YANG Push onboarding (Swisscom)
  - IEPG
  - Modelling Boundaries: **draft-davis-netmod-modelling-boundaries-00**
- Data without clear semantic (both config and operational data) is not usefull
- End goal: analyze the data, from the data collection system, with the proper context/semantic, for anomaly detection and, in the end, closed loop automation
- Goal is not to expose new information via YANG but rather to define what needs to be kept as metadata (or Data Manifest) to ensure that the data can still be interpreted correctly even:
  - if the source device is not accessible (from the collection system)
  - If the source device has been updated or has a new configuration
- Per-node capability discovery exists
  - YANG Modules describing Capabilities for Systems and Datastore Update Notifications, RFC9196 + YANG Instance Data File Format, RFC9195
  - Per-Node Capabilities for Optimum Operational Data Collection , draft-claise-netconf-metadata-forcollection-03
- But how were data actually metered, under which circumstances?

Data Collection Vantage Point

Counter1 = 42
Status = UP

No updates since t1
- Problem: Telemetry issue?
- Problem: Sender overloaded?
- Problem: bug?
- No problem: long telemetry cadence?
- No problem: “on-change”?
- No problem: “suppress-redundancy”?

End goal: analyze the data, from the data collection system, with the proper context, for anomaly detection and, in the end, closed loop automation



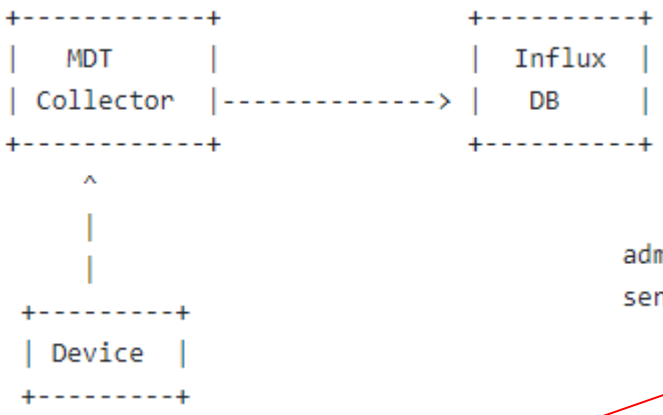
Proposal: Data Manifest

- Data Manifest composed of 2 YANG models for storing the context:
  - Platform Manifest: part of the Data Manifest that completely characterizes the platform producing the data.
  - Data Collection Manifest: part of the Data Manifest that completely characterizes how and when the telemetry was metered.
- "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."

New: renaming

- ietf-collected-data-platform-manifest
  -> ietf-platform-manifest
- Ietf-collected-data-manifest
  -> ietf-data-collection-manifest
- Prefixes unchanged

New: example with InfluxDB



Influx input – no manifest:

```
admin_status,device="PE1",interface="gig1" val=T
sent_bytes,device="PE1",interface="gig1" val=1234
```

Influx input – with manifest:

```
admin_status,device="PE1",interface="gig1" val=T,subId=42,path=<path>
sent_bytes,device="PE1",interface="gig1" val=1234,subId=42,path=<path>
```

Identifies the Platform Manifest

Identifies the Data Collection Manifest

```
platform-manifest,device="PE1" val=<platform-manifest>
collection-data-manifest,device="PE1",subId=42 val=<data-manifest>
```

Possible way of storing manifest in InfluxDB

Next Steps

- Identified Improvements:
  - Use Private Enterprise Number (PEN) from IANA to identify vendors
  - Data integrity
  - Add IOT Controller example to be added (Dean Bogdanovic)
- How to properly specify devices / virtual devices as source?
  - See the inventory discussions at this IETF
- What to include for data manifest?

    https://github.com/JeanQuilbeufHuawei/draft-collected-data-manifest/issues/9

  - MDT subscription is a collection of XPaths
    but NETCONF subscription is a xml filter
    what to use as key for subscription contents?
  - Should we include encoding? (XML, JSON, CBOR, Protobuf)

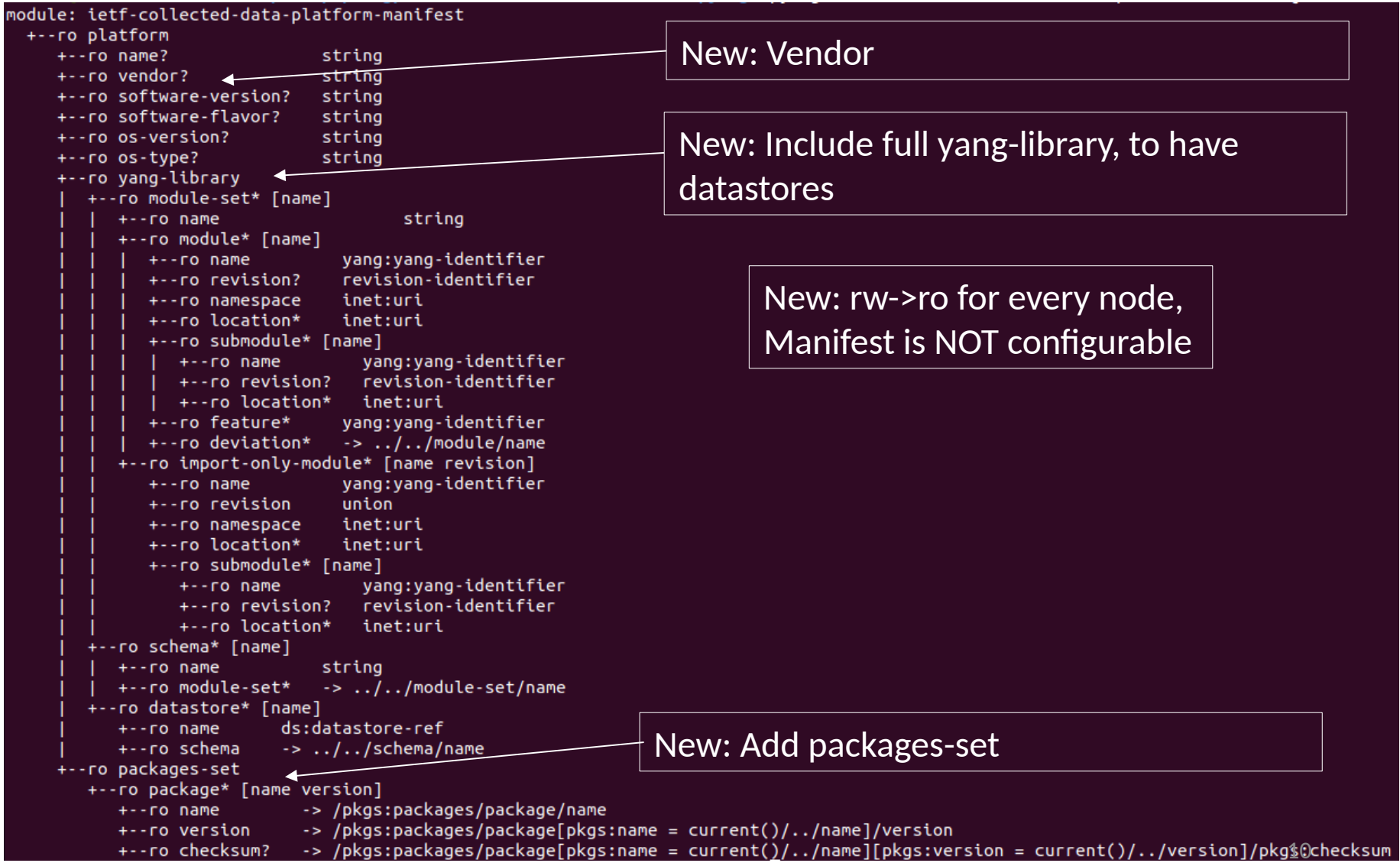
Conclusion

- Multiple presentations/draft versions
- New draft version (Thanks Med & Tianran & Ignacio & Joe for your review)
- We would like to request for WG adoption

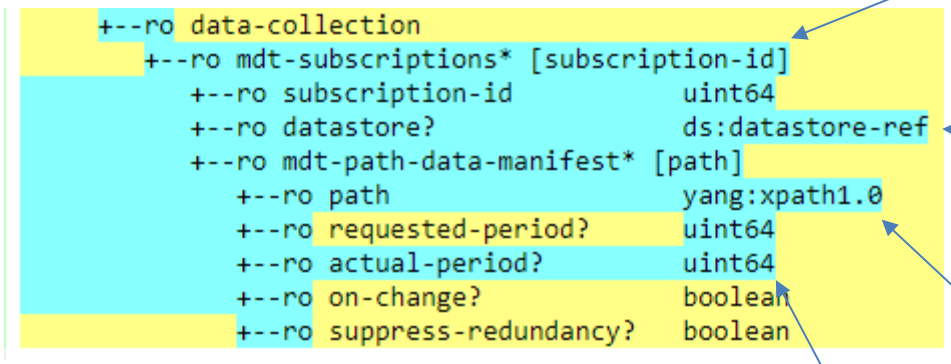
Feedback, suggestions, issues, PRs:
https://github.com/JeanQuilbeufHuawei/draft-collected-data-manifest

BACKUP

Platform Manifest



Data Collection Manifest



New list indexed by subscription ID

Specify datastore per subscription
- monitor diff between config and operational
- monitor candidate

MDT path typed as XPath

Unsigned ints for periods

Update Frequency

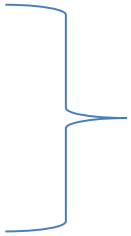
- Platform manifest:
  - **Update when**: platform changes (i.e. at reboot)
- Data collection manifest:
  - **Update when**: collection condition changes:
  - New subscription
  - Collection period is adjusted based on CPU availability

Collecting Manifests: Use Telemetry

- Platform and data collection manifests are about telemetry
  - we can assume a telemetry system is available
- Event-driven Telemetry/onChange well suited to stream the manifests updates only
- Collectors can choose or not to collect the manifests / header approach would force collector to have the manifests
- Collectors know IDs of their subscription -> YANG key for selecting the data-manifest they need

Mapping Data to Data Manifest

- Collected data needs the following metadata:
  - Source device -> for mapping to platform manifest
  - Subscription ID
  - MDT path



For mapping data to data manifest

Specifying how to store metadata for collected data is out-of-scope of this draft.