

OPSAWG
Internet-Draft
Intended status: Standards Track
Expires: April 28, 2022

B. Claise
J. Quilbeuf
Huawei
D. Lopez
Telefonica I
T. Graf
Swisscom
October 25, 2021

A Data Manifest ~~for Streaming Telemetry~~
Contextualized Telemetry Data
draft-claise-opsawg-collected-data-manifest-00

Abstract

Most network ~~equipment~~equipment feature telemetry as a mean to monitoring their status. Several protocols exist to this end~~er~~, for example, the model-driven telemetry governed by YANG models. ~~Some of These~~ these protocols provide the data itself, without any contextual information about the collection method. This can render the data unusable if that context is lost, for instance, when the data is stored without the ~~relevant~~relevant information. This ~~draft document~~ proposes a YANG data model to store that contextual information, ~~that must be stored~~ along with the collected data in order to keep the collected data exploitable in the future.

Commenté [BMI1]: As there are specs out there where "some" context of the telemetry data is also shared as part of the model. See for example, draft-ietf-dots-telemetry (measurement interval, sampling period, etc.)

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <https://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on April 28, 2022.

Copyright Notice

Copyright (c) 2021 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Terminology	2
2. Introduction	3
3. Platform Manifest	4
3.1. Overview of the model	4
3.2. YANG module ietf-collected-data-platform-manifest	6
4. Collection Manifest	8
4.1. Overview of the model	8
4.2. YANG module ietf-collected-data-manifest	9
5. Mapping data to data manifest	11
6. Security Considerations	12
7. IANA Considerations	12
8. Contributors	12
9. Open Issues	12
10. References	12
10.1. Normative References	12
10.2. Informative References	13
Appendix A. Changes between revisions	13
Acknowledgements	13
Authors' Addresses	13

1. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

Data Manifest: all the necessary metadata required to interpret the telemetry information.

Platform Manifest: part of the Data Manifest that completely identifies the platform producing the data.

2. Introduction

Network elements use Model-driven Telemetry (MDT) to continuously stream information, both counters and state information. This streamed information is used for network monitoring, directly in the data collection ~~or, in a~~ closed-loop automation systems. Also, that streamed data can be stored, e.g., ~~or~~ in a database (sometimes called a big data lake) for further analysis.

When streaming YANG-structured data using ~~objects with~~ YANG-PUSH-Push [RFC8641], there is a semantic definition in the corresponding YANG module definition. On top of that definition, ~~it's~~ it is also important to ~~understand~~ maintain contextual information about the collection environment.

As an example, a database could contain a specific counter time series. When analyzing the data, ~~it's~~ it is important to understand that this counter was requested from the network element at specific cadence, as this exact cadence might not be observed in the time series, potentially implying that the network element was under stress. The same time series might report some values as 0 ~~or~~ might even omit some values in the series. This might be explained by a too small observation period, compared to the minimum-observed-period [I-D.claire-netconf-metadata-for-collection]. Again, knowing the conditions under which the counter was collected and streamed is key. Indeed, taking into account the value of 0 might lead to ~~the~~ a wrong conclusion that the counter dropped to zero. This document specifies the data manifest, which contains the required information to characterize how and when the telemetry information ~~were~~ was metered.

Precisely identifying the device used for producing the data (that is the platform manifest) ~~is also key~~ may also be useful to complete the collection context. As an example, knowing the exact device software specification might reveal a particularity in the observed data, explained by a specific bug, or a specific bug fix. On top of that, in particular for MDT, it is crucial to know the set of YANG modules supported by the device, ~~along with their deviation~~. In some cases, there might even be some backwards incompatible changes in native modules between one OS version to the next one. These information must be compiled in a platform manifest that must be included in the data manifest.

Some related YANG modules have been specified to ~~discover~~ retrieve the device capabilities:

- o [I-D.ietf-netconf-notification-capabilities] which models the device capabilities regarding the production and export of telemetry data.

Commenté [BMI2]: No sure what is meant here, especially in reference to "used ..."

Commenté [BMI3]: A network inventory may catalog some generic information about these "devices". Having an identifier to the device may allow to correlate the two sets of information (telemetry vs its source).

However, I guess you meant here more than that an identifier.

Commenté [BMI4]: You may consider changing it to "source" as the device may not be sufficient to characterize the data origin.

Think about a device that embeds several network functions (CGN, FW, RR, etc.), each maintained by a distinct vendor and each may be a source of telemetry data.

Commenté [BMI5]: This is required to check the reliability of the collected data, not only to exploit it in the future.

- o [I-D.claise-netconf-metadata-for-collection], which is based on the previous draft to define the optimal settings to stream ~~specific~~specific items (i.e., per sensor-path).

While these YANG modules are important to discover the capabilities before applying the telemetry configuration (such as on-change), some of this information is part of the context for the streaming data. Our goal is to represent the data manifest for a given device. This manifest contains two parts, the platform manifest and the data-collection manifest. The platform manifest is "pretty" stable and should change only when the device is updated or patched. On the other hand, the data collection is likely to change each time a new MDT subscription is requested and might even change if the device load increases and collection periods are updated. To separate these two parts, we enclose each of them in its own module. We first present the module for the platform manifest in Section 3 and then the module for the collection manifest in Section 4. The full data manifest is obtained by combining these two modules.

This data manifest instance file MUST be streamed all with the data and stored along with the collected data. In case the collected data are moved

to a different place (typically a database), ~~the-its companion~~ data manifest MUST

follow the collected data. This can render the data unusable if that context is lost, for instance when the data is stored without the ~~relevant~~relevant information. The data manifest MUST be encoded with the

the YANG instance data file format

[I-D.ietf-netmod-yang-instance-file-format]. The YANG instance data file MUST be updated when the data manifest information changes (for example, when a router is upgraded), and the new timestamps MUST be used [I-D.ietf-netmod-yang-instance-file-format].

3. Platform Manifest

3.1. Overview of the model

Figure 1 contains the YANG tree diagram [RFC8340] of the ietf-collected-data-platform-manifest module.

Commenté [BMI6]: Move to Section 4...where it makes sense to discuss the normative behavior. Thanks.

Commenté [BMI7]: ... if the context is not part of the data definition itself.

Commenté [BMI8]: If the context is not part of the collected data, how the association between the two is maintained?

```

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

```

Commenté [BMI9]: This may not identify the piece of software that sourced the data (think about a set of service functions) unless this is covered by “software-version”, but the version does not reflect the vendor

Figure 1: YANG tree diagram for ietf-collected-data-platform-manifest module

The platform manifest contains ~~all-the-a comprehensive set of~~ information to ~~unambiguously~~ ~~identify~~ ~~characterize~~ a ~~device~~ ~~data source~~. The platform is identified by a set of parameters (`'platform'`, `'software-version'`, `'software-flavor'`, `'os-version'`, `'os-type'`) that are aligned with the YANG Catalog www.yangcatalog.org ~~draft~~ [I-D.clacla-netmod-model-catalog] so that the YANG catalog could be used to retrieve the YANG modules a posteriori.

The platform manifest also includes the module-set, as defined in the YANG Library [RFC8525]. That module set is particularly useful to define ~~to define~~ the sensor paths, as they are based on module names.

3.2. YANG module ietf-collected-data-platform-manifest

```
<CODE BEGINS> file "ietf-collected-data-platform-
manifest@2021-10-15.yang"

module ietf-collected-data-platform-manifest {
  yang-version 1.1;
  namespace "urn:ietf:params:xml:ns:yang:ietf-collected-data-plaform-
manifest";
  prefix platform-manifest;

  import ietf-yang-library {
    prefix yanglib;
    reference
      "RFC8525: YANG Library";
  }

  organization
    "IETF NETCONF (Network Configuration) Working Group";
  contact
    "WG Web: <https://datatracker.ietf.org/wg/netconf/>
    WG List: <mailto:netconf@ietf.org>
    Author: Benoit Claise <mailto:benoit.claise@huawei.com>
    Author: Jean Quilbeuf <mailto:jean.quilbeuf@huawei.com>";
  description
    "This module describes the platform information to be used as
    context of data collection from a given network element. The
    contents of this model must be streamed along with the data
    streamed from the network element so that the platform context
    of the data collection can be retrieved later.

    The data content of this model should not change except on
    upgrade or patching of the device.

    The key words 'MUST', 'MUST NOT', 'REQUIRED', 'SHALL',
    'SHALL NOT', 'SHOULD', 'SHOULD NOT', 'RECOMMENDED',
    'NOT RECOMMENDED', 'MAY', and 'OPTIONAL' in this document
    are to be interpreted as described in BCP 14 (RFC 2119)
    (RFC 8174) when, and only when, they appear in all
    capitals, as shown here.

    Copyright (c) 20212022 IETF Trust and the persons identified as
    authors of the code. All rights reserved.

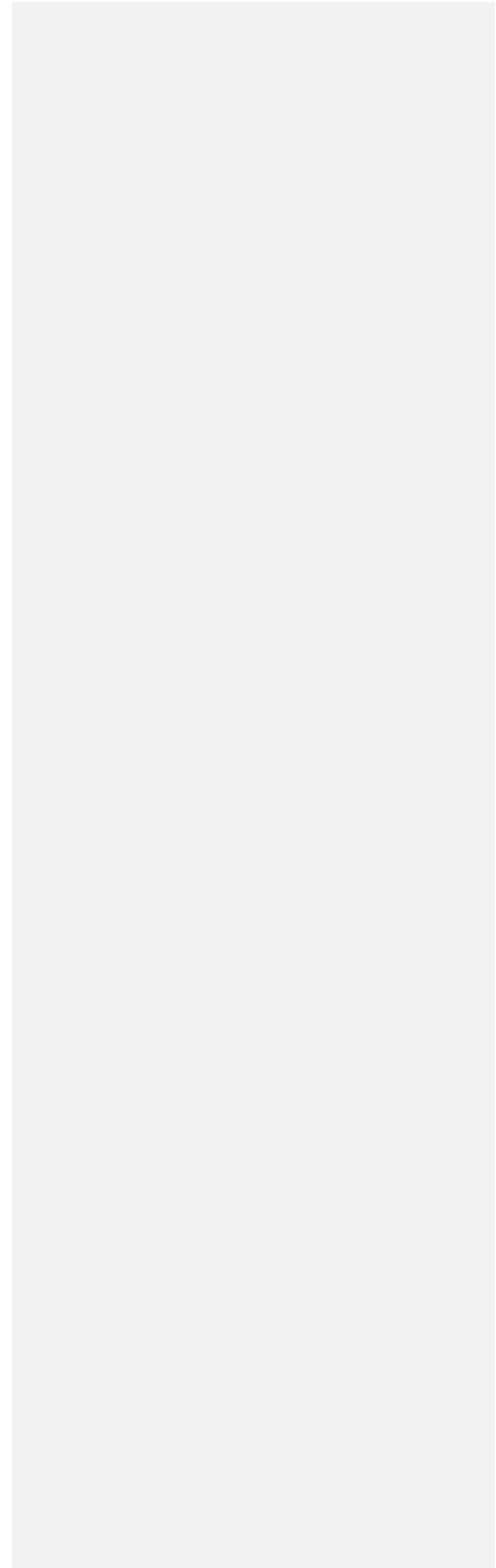
    Redistribution and use in source and binary forms, with or
    without modification, is permitted pursuant to, and subject
    to the license terms contained in, the Simplified BSD License
    set forth in Section 4.c of the IETF Trust's Legal Provisions
    Relating to IETF Documents
```

```
(https://trustee.ietf.org/license-info).
This version of this YANG module is part of RFC XXXX; see the
RFC itself for full legal notices.  ";

revision 2021-10-15 {
  description
    "Initial revision";
  reference
    "RFC xxxx: Title to be completed";
}

container platform {
  description
    "Contains information about the platform that allows to identify
and understand the individual data collection information.  ";
  leaf platform {
    type string;
    description
      "Platform on which this module is implemented.";
  }
  leaf software-version {
    type string;
    description
      "Name of the version of software. With respect to most network
device appliances,
this will be the operating system version. But for other
YANG module
implementation, this would be a version of appliance
software. Ultimately,
this should correspond to a version string that will be
recognizable by
the consumers of the platform.";
  }
  leaf software-flavor {
    type string;
    description
      "A variation of a specific version where
YANG model support may be different. Depending on the
vendor, this could
be a license, additional software component, or a feature
set.";
  }
  leaf os-version {
    type string;
    description
      "Version of the operating system using this module. This is
primarily useful if
the software implementing the module is an application that
requires a specific
operating system.";
  }
  leaf os-type {
    type string;
    description
      "Type of the operating system using this module. This is
primarily useful if
the software implementing the module is an application that
requires a
```

Commenté [BMI10]: Should this be structured as a PEN?




```

        specific operating system.";
    }
    list module-set {
        key "name";
        description
            "The list of all modules defined on the device. The name of the
modules
        can be used in sensor-paths to identfyidentify a specific
collected object.";
        uses yanglib:module-set-parameters;
    }
}
}

<CODE ENDS>

```

4. Collection Manifest

4.1. Overview of the model

Figure 2 contains the YANG tree diagram [RFC8340] of the ietf-collected-data-manifest module.

```

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

```

Figure 2: YANG tree diagram for ietf-collected-data-manifest module

The data-collection container contains the information related to individual items collection. This subtree currently contains only information about MDT collection. It should be extended and extendable to represent other kinds of data collection.

With MDT collection, the granularity of the collection is defined by the sensor path. Note that all devices do not support an arbitrary granularity up to the leaf, usually for performance reasons. Each sensor-path currently collected by the device should show up in the mdt-collection-item list.

For each sensor-path, the collection context must be specified including:

- o on-change: when set to true, an update is sent as soon as and only when a value changes. This is also known as Event-Driven

Commenté [BMI11]: Shouldn't this be unsigned ?

Commenté [BMI12]: Idem

Telemetry (EDT). When set to false, the values are sent regularly.

- o suppress-redundancy (only when on-change is false): reduce ~~bandwidth~~bandwidth usage by sending a regular ~~update~~update only if the value is different from the previous ~~update~~update.
- o requested-period (only when on-change is false): period between two updates requested by the client for this sensor-path
- o current-period (only when on-change is false): current period between two updates

This information is crucial to understand the collected values. For instance, the "on-change" and "suppress-redundancy" options, if set, might remove a lot of messages from the database because values are sent only when there is a change.

4.2. YANG module ietf-collected-data-manifest

<CODE BEGINS> file "ietf-collected-data-manifest@2021-10-15.yang"

```
module ietf-collected-data-manifest {
  yang-version 1.1;
  namespace "urn:ietf:params:xml:ns:yang:ietf-collected-data-manifest";
  prefix data-manifest;

  organization
    "IETF NETCONF (Network Configuration) Working Group";
  contact
    "WG Web: <https://datatracker.ietf.org/wg/netconf/>
    WG List: <mailto:netconf@ietf.org>
    Author: Benoit Claise <mailto:benoit.claise@huawei.com>
    Author: Jean Quilbeuf <mailto:jean.quilbeuf@huawei.com>";
  description
    "This module describes the context of data collection from a
    given network element. The contents of this model must be
    streamed along with the data streamed from the networknetwork
    element so that the context of the data collection can
    be retrieved later.

    This module must be completed with
    ietf-collected-data-platform-manifest
    to capture the whole context of a data collection session.

    The key words 'MUST', 'MUST NOT', 'REQUIRED', 'SHALL',
    'SHALL NOT', 'SHOULD', 'SHOULD NOT', 'RECOMMENDED',
    'NOT RECOMMENDED', 'MAY', and 'OPTIONAL' in this document
```

are to be interpreted as described in BCP 14 (RFC 2119) (RFC 8174) when, and only when, they appear in all capitals, as shown here.

Copyright (c) ~~2021~~-2022 IETF Trust and the persons identified as authors of the code. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, is permitted pursuant to, and subject to the license terms contained in, the Simplified BSD License set forth in Section 4.c of the IETF Trust's Legal Provisions Relating to IETF Documents (<https://trustee.ietf.org/license-info>). This version of this YANG module is part of RFC XXXX; see the RFC itself for full legal notices. ";

```
revision 2021-10-15 {
  description
    "Initial revision";
  reference
    "RFC xxxx: Title to be completed";
}
```

```
container data-collection {
  description
    "Defines the information for each collected objetobject";
  list mdt-collection-item {
    description
      "Status of the collection for the given sensor-path";
    key "sensor-path";
    leaf sensor-path {
      description
        "The sensor path that was configured to collect information
```

```
about
    this object. Sensor path syntax is
      module_name:node_name ( / (module_name)? node_name)*
    where module_name is the name of a module and node_name is the
    name of a node (list, container or leaf). Module name is only
    necessary if the preceding node in the path is not defined in
    the same module.
```

```
and
    Module name must be defined in the platform-manifest module
    every node name must be defined in the preceding node in the
    path.";
```

```
    type string;
  }
  leaf requested-period {
    description
      "Requested period, in milisecondmillisecond, between two
sucesivessuccessive updates.";
```

```
    type int64;
```

```

        // when on-change is false;
    }
    leaf current-period {
        description
            "Current period, in millisecond millisecond, between two
successive successive update updates";
        type int64;
        // when on-change is false;
    }
    leaf on-change {
        description
            "Whether the sensor path is collected only when there is a
change,
            i.e. Event-Driven Telemetry is enabled.";
        type boolean;
    }
    leaf suppress-redundancy {
        description
            "Whether the information is sent at every period or only when
there
            is a change between two successive pollings..";
        type boolean;
    }
    }
    // we could augment here with other kind of collection items
}
}

<CODE ENDS>

```

5. ~~Mapping data-Data to data-Data manifest~~Manifest

The data should be mapped to the data manifest. Since the data manifest will not change as frequently as the data itself, it makes sense to map several data to the same data manifest. Somehow, the collected data must include a metadata pointing to the corresponding data manifest.

The platform manifest is likely to remain the same until the device is updated. So, the platform manifest only ~~need~~ needs to be collected once per streaming session and updated after a device reboot.

For MDT, we can rely on the sensor-path to map the collected data to the data manifest. In that sense, collecting one instance of the data-manifest per device is sufficient to get the data manifest of all data connected from that device.

Commenté [BMI13]: I see this answers one of the previous comments I had above. A forward citation would be sense there.

6. Security Considerations

7. IANA Considerations

This document includes no request to IANA.

8. Contributors

9. Open Issues

- o Do we want to the hardware specifications, next to the OS information?
- o Do we want to handle the absence of values, i.e. add information about missed collection or errors in the collection context ? It could also explain why some values are missing. On the other hand, this might also be out scope.
- o How do we handle other kinds of collection than MDT like netflow, SNMP, CLI ? How do we map the collected data to the data-manifest ?
- o Align the terms with the YANG Push specifications. Ex: sensor-path to subscription (TBC)
- o Better explain the on-change example.

10. References

10.1. Normative References

- [I-D.ietf-netmod-yang-instance-file-format]
Lengyel, B. and B. Claise, "YANG Instance Data File Format", draft-ietf-netmod-yang-instance-file-format-21 (work in progress), October 2021.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <<https://www.rfc-editor.org/info/rfc2119>>.
- [RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <<https://www.rfc-editor.org/info/rfc8174>>.
- [RFC8340] Bjorklund, M. and L. Berger, Ed., "YANG Tree Diagrams", BCP 215, RFC 8340, DOI 10.17487/RFC8340, March 2018, <<https://www.rfc-editor.org/info/rfc8340>>.

- [RFC8525] Bierman, A., Bjorklund, M., Schoenwaelder, J., Watsen, K., and R. Wilton, "YANG Library", RFC 8525, DOI 10.17487/RFC8525, March 2019, <<https://www.rfc-editor.org/info/rfc8525>>.
- [RFC8641] Clemm, A. and E. Voit, "Subscription to YANG Notifications for Datastore Updates", RFC 8641, DOI 10.17487/RFC8641, September 2019, <<https://www.rfc-editor.org/info/rfc8641>>.

10.2. Informative References

- [I-D.clacla-netmod-model-catalog]
Clarke, J. and B. Claise, "YANG module for yangcatalog.org", draft-clacla-netmod-model-catalog-03 (work in progress), April 2018.
- [I-D.claise-netconf-metadata-for-collection]
Claise, B., Nayyar, M., and A. Sesani, "Per-Node Capabilities for Optimum Operational Data Collection", draft-claise-netconf-metadata-for-collection-02 (work in progress), July 2021.
- [I-D.ietf-netconf-notification-capabilities]
Lengyel, B., Clemm, A., and B. Claise, "YANG Modules describing Capabilities for Systems and Datastore Update Notifications", draft-ietf-netconf-notification-capabilities-21 (work in progress), October 2021.

Appendix A. Changes between revisions

Initial version

Acknowledgements

...

Authors' Addresses

Benoit Claise
Huawei

Email: benoit.claise@huawei.com

Jean Quilbeuf
Huawei

Email: jean.quilbeuf@huawei.com

Diego R. Lopez
Telefonica I
Don Ramon de la Cruz, 82
Madrid 28006
Spain

Email: diego.r.lopez@telefonica.com

Thomas Graf
Swisscom
Binzring 17
Zurich 8045
Switzerland

Email: thomas.graf@swisscom.com