Internet Research Task Force
Internet-Draft
Intended status: Informational
Expires: 30 August 2026

C. Zhou
D. Chen
China Mobile
P. Martinez-Julia
NICT
Q. Ma
Huawei

26 February 2026

Data Collection Requirements and Technologies for Network Digital Twin
draft-zcz-nmrg-digitaltwin-data-collection-04

Abstract

A Network Digital Twin is a virtual representation of a real network, which is meant to be used by a management system to analyze, diagnose, emulate, and then control the real network based on data, models, and interfaces. The construction and state update of a Network Digital Twin requires obtaining real-time information of the physical network it represents (i.e., telemetry data). This document aims to describe the data collection requirements and provide data collection methods or tools to build the data repository for building and updating a network digital twin.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

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 30 August 2026.

Copyright Notice

Copyright (c) 2026 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 Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

Table of Contents

1. Introduction 3
2. Definitions and Acronyms 3
3. Data Collection Requirements for Network Digital Twin 4
3.1. Target-driven and On-demand Collection 4
3.2. Diverse Tools for Various Data Collection 4
3.3. Lightweight and Efficient Collection 5
3.4. Open and Standardized Interfaces 5
3.5. Naming for Caching 6
3.6. Efficient Multi-Destination Delivery 6
4. Data Collection Technologies for Network Digital Twin 6
4.1. Existing Data Collection Methods/Tools 6
4.2. Innovation Directions on Data Collection 7
5. Knowledge and Instruction Driven Data Collection Method for Network Digital Twin 8
5.1. Overview 8
5.2. Efficient Data Collection Mechanism 8
5.3. Data Collection Process 10
5.4. Query and Aggregation Functions 11
6. Summary 12
7. Security Considerations 13
8. IANA Considerations 13
9. References 13
9.1. Normative References 13
9.2. Informative References 13
Authors' Addresses 14

1. Introduction

With the deployment of Internet of Things (IoT), cloud computing and data center, etc., the scale of the current network is expanded gradually. However, the increase of network scale also leads to an increase in the complexity of the current network, and it induces plenty of problems. In order to improve the autonomy ability of network and reduce potential negative effects on physical and virtual networks, we consider that an endogenous intelligent and autonomous network architecture which achieves self-optimization and decision is indispensable (in general, self-management and self-operation). The digital twin technology addresses the challenge of building self-management systems because it can optimize and validate policies through real-time and interactive mapping with physical entities [I-D.irtf-nmrg-network-digital-twin-arch].

Data is the cornerstone required for constructing a digital twin for a network, namely a Network Digital Twin (NDT). In the face of large network scale, data collection, storage and management are faced with great challenges. So, data collection methods and tools should meet the requirements of target-driven, diversity, lightweight and efficiency, while being open and standardized. Among all the requirements, achieving a lightweight and efficient data collection method is of the most importance. If the full-data collection method is adopted, huge storage space and bandwidth resource are needed, especially for complex scenarios that require real-time data and traffic from multi-source and heterogeneous devices. Therefore, it is extremely important to agree on lightweight and efficient data collection, aggregation, and correlation methods, toward building the transmission of monitoring information (telemetry data), processing, and storage required to build an effective NDT system.

This document aims to describe the data collection requirements and propose efficient data collection methods or tools to build the data repository for network digital twin.

2. Definitions and Acronyms

PN: Physical Network

IMC: Instruction Management Center

DSC: Data Storage Center

NDT: Network Digital Twin

TSE: Telemetry Streaming Element

RDF: Resource Description Framework

CEP: Complex Event Processing

3. Data Collection Requirements for Network Digital Twin

3.1. Target-driven and On-demand Collection

The monitoring data of a network is the basis to build an NDT system. Such data is collected from physical and virtual networks. It includes, but is not limited to, the following types:

* Provisional and operational status of physical or virtual devices, as well as the network topology with all network elements.

* Configuration data that is required to transform a network system from its initial default state into its current state.

* Running status of physical, logical, or virtual ports and links.

* Logs and events records of all the network elements.

* Statistics (packet loss, traffic throughput, latency, etc.) of flows and ports.

* Various data regarding users and services.

* Life-cycle operation data of all network elements.

* All above data in time series.

The collection of the monitoring information from a network required for maintaining an NDT (telemetry data) should be in target-driven and on-demand mode. It is not always necessary to collect all monitoring information from the network (telemetry data) listed above because of the high cost of resources (CPU, memory, bandwidth etc.). The type, frequency and method of data collection aim to meet the application of an NDT depend on the specific network topology and application requirements.

3.2. Diverse Tools for Various Data Collection

The different types of monitoring information required to maintain an NDT (telemetry data) have several characteristics. Some data (e.g. hardware status, environmental data, etc.) requires lower collecting frequency, while others (e.g. flow status, link fault, etc.) need higher level of real-time. Some data (e.g. device status, port statistics, etc.) can be collected directly and simply via normal tools, while others (e.g. per-flow latency, traffic matrix, etc.) can only be acquired through complex network measurement technologies. It is unrealistic to find or define a uniform data collection method that is suitable for all types of data. Therefore, multiple tools or methods are needed to collect the different types of data required to build the NDT entity.

3.3. Lightweight and Efficient Collection

Data collection tools and methods should be as lightweight as possible, so as to reduce the occupation of network equipment resources and ensure that data collection does not affect the normal operation of the network. The major requirements are listed as below.

* Data collection tools and methods need to improve efficiency of execution, reduce the cost of computing, storage and communication bandwidth.

* The collection of redundant data should be avoided or minimized.

* For the data set that needs to be collected, making full use of the data compression technology, to reduce the resource cost in the collection phase. There must be lossy or lossless compression methods available to data sources, which will be applied together with other functions before data is transmitted.

3.4. Open and Standardized Interfaces

Data collection interfaces used to build the NDT should be open and standardized to help avoid either hardware or software vendor lock, and facilitate inter-operability among different vendors. The major requirements of data collection interfaces are:

* Support configuration management, including the data collection channel, frequency or period, etc.

* Support several rate options (e.g. minute-level, 10-second level, second level (near real time), and millisecond-level) to accommodate different data requirements from applications.

* Be extensible so that more features can be added in future with limited parameter changes and with backward compatibility.

* Be able to provide secure and reliable information exchange mechanism.

* Be able to enforce federation policies to allow information to be exchangeable among domains while ensuring authorization and scoping is controlled.

3.5. Naming for Caching

Both raw monitoring information (telemetry data) and knowledge items obtained from monitoring must be able to be addressed uniquely. This means to give a unique identifier or "name" to each data or knowledge item that references it. This name will be used by caching mechanisms to store the data and provide it for clients that request it, which will also use such name.

Global names and federated names must be supported. A name schema, name hierarchy, and name part ontology must be defined and maintained together with other naming systems, such as DNS for global names.

3.6. Efficient Multi-Destination Delivery

The maintenance of NDT systems will not be the sole purpose of monitoring information and knowledge communication. Other applications would also request raw monitoring information (telemetry data) or knowledge items. They can use the name to identify it. The monitoring system (telemetry system), following the recommendations of RFC 9232 [RFC9232], will deliver the requested data or knowledge items to the requesters as much efficiently as possible. On the one hand, items will be provided by the closest cache to the destination of the data. On the other hand, items will be replicated in the best nodes, following an efficient multi-cast spanning tree. Different underlying protocols can be used to achieve this mechanism.

Delivering knowledge items instead of raw telemetry data enables digital twins to be aware of the context of data and highly relieve from complex processing, which will be performed by the entities which are best suited for running each type of processing.

4. Data Collection Technologies for Network Digital Twin

4.1. Existing Data Collection Methods/Tools

Currently, some widely-used tools, such as SNMP, RESTCONF [RFC8040], NETCONF [RFC6241], Telemetry, INT (In-band Network Telemetry), DPI (Deep Packet Inspection), IPFIX [RFC7011], etc. can be candidate tools to collect data for network digital twin. YANG data model and associated mechanisms defined in [RFC8639][RFC8641] enable subscriber-specific subscriptions to a publisher’s event streams, and can help subscriber applications to request a continuous and customized stream of updates from a YANG datastore. Appendix-A in

[RFC9232] gives a survey on existing network telemetry techniques, which explores an overview of management plane, control plane and data plane telemetry techniques and standards.

Moreover, some new innovation methods can help increase the data collection efficiency. For example, [I-D.ietf-opsawg-collected-data-manifest] proposes a YANG model to store contextual information along with the collected data in order to keep the collected data exploitable; [RFC9506] addresses the network performance measurement problem under encrypted transport protocols, via proposing some hybrid measurement methods based on marking bits in packet headers without relying on external network management systems. [RFC7594] introduces a measurement method named Large-Scale Measurement of Broadband Performance (LMAP) that works in a coordinated fashion to perform network performance measurement tasks.

4.2. Innovation Directions on Data Collection

Current data collection methods and tools (YANG, xCONF, SNMP, Telemetry, etc.) listed above can help acquire network data to build an NDT system, which may be with low maturity and low-level capabilities of data service and data modelling. To build a more mature NDT system with high-level capabilities, it is necessary to explore more innovative data collection technologies. The following are several potential innovation directions.

* High-performance data collection technology based on programmable circuits, which offer the potential for hardware acceleration and customization.

* Measurement methods for complex monitoring information such as network performance and network traffic.

* Distributed and collaborative data collection techniques for integrating and fusing data from multiple data sources, and the time synchronization problem of data acquisition.

* Provision of processed information, jointly and separately, by applying the function indicated by data requester.

* Assessment of federation policies in data provisioning to enable cross-domain data provision and implement multi-domain digital twin scenarios.

* Investigating self-adaptive and self-learning data collection techniques that can dynamically adjust data collection parameters, methods, and priorities based on network conditions and user requirements.

* Exploring machine learning and AI techniques to enhance the efficiency and accuracy of data collection processes by identifying patterns, correlations and anomalies in network data.

5. Knowledge and Instruction Driven Data Collection Method for Network Digital Twin

5.1. Overview

An NDT’s data repository sub-system manages all network data, in real time, from the PN to the NDT. Sufficient and timely data are always required to construct the twin entity and various data models. However the existing methods collect the full data from the PN for modeling, and do not consider problems like time-lag, insufficient storage resources, low computational efficiency and waste of bandwidth resources caused by data transmission.

This section proposes an efficient data collection method, named "knowledge and instruction driven data collection". This data collection method is based on sending instructions to the elements of the PN for them to pre-process the data (data cleaning or knowledge representation) before sending it back to be applied to the NDT.

5.2. Efficient Data Collection Mechanism

The management system structure consists of the PN and the NDT. The PN includes multiple Data Storage Centers (DSC) and Telemetry Streaming Element (TSE), and the NDT includes the Instruction Management Center (IMC) and Data Storage Center (DSC). The TSE has multiple functions, including data collection, data aggregation, data correlation, knowledge representation and query, etc. In addition, a Complex Event Processing (CEP) engine is integrated into TSE to perform queries to the streamed data. The IMC has two functions: one is used to manage the registration of the DSC in the PN side, and its registration information can include various key information such as the IP address of the DSC in the PN side, choose data type, and various index names in the data, data source name and data size, etc. The other is used to adaptively configure data collection instructions according to the collection requirements of the DSC in the NDT side and search for IP addresses to send instructions. The instruction-carrying information includes rule-based mathematical expressions, executable models in ".exe" format, dynamic collection frequency, parameter lists, program text files in ".m" format, text files with parameter configuration, and other types of files. Instructions are flexible and programmable, and can be created, modified, combined, and deleted at any time according to requirements. When the DSC of the NDT side requests data to the IMC, the IMC searches the IP address of the DSC in the database with the registration information, which is built according to critical information, such as data type and data name, and functional instructions for data processing or knowledge representation can be implemented depending on the demand configuration. The DSC of the NDT side stores the effective information after data processing and knowledge representation returned by the TSE.

The DSC in the PN side has two functions. On the one hand, it stores data of various types, such as performance indicators, operational status, log, traffic scheduling, business requirements, etc. On the other hand, it has the function of automatically parsing the instructions sent by the TSE. Then the operating environment of the instruction is configured according to the instruction needs, and data processing or knowledge representation is performed based on the instruction. Data processing mainly includes data cleaning, filling missing data, normalization, conflict verification, etc. Knowledge representation refers to the representation of the original data as a data structure that can be used for efficient computation. Such representation results are similar to machine language, which is conducive to the rapid and accurate construction of the model. The role of knowledge representation is to represent the original data as a data structure that can be used to efficiently calculate.

```
       +------------------------------+   +-----------------------+
       |   Physical  Network          |   |  Netowrk Digital Twin |
       | +-----+    +-----+  +------+ |   |  +------+  +-------+  |
       | |     |    |     |  |      | |   |  |      |  |       |  |
       | | DSC |... | DSC |  | TSE  | |   |  |  IMC |  |  DSC  |  |
       | |     |    |     |  |      | |   |  |      |  |       |  |
       | +-+---+    +--+--+  +---+--+ |   |  +---+--+  +----+--+  |
       |   |           |         |    |   |      |          |     |
       +------------------------------+   +-----------------------+
           |           |         |               |          |
           | 1.1. Register       |               |          |
           +-----------+--------->               |          |
           |           |         |               |          |
           |           | 1.2. Register           |          |
           |           +--------->               |          |
           |           |         | 1.3. Register |          |
           |           |         +--------------->          |
           |           |         |             2. Data req. |
           |           |         |               <----------+
           |           |         | 3. Query and instruction |
           |           |         |    configuration         |
           |           |         |               +          |
           |           |         4. Send instructions       |
           |           |         <---------------+          |
           |           |         |               |          |
           |           |   5. Parse and execute  |          |
           |           |      instruction        |          |
           | 6. Data subscript.  |               |          |
           <---------------------+               |          |
           | 7. Knowledge        |               |          |
           |    representation   |               |          |
           |     8. Data pushing |               |          |
           +--------------------->               |          |
           |           | 9. Data aggregation and |          |
           |           |    correlation          |          |
           |           |         | 10. Send processed data  |
           |           |         +-------------------------->
           |           |         |               |          |
```

Figure 1: Data Collection Process

5.3. Data Collection Process

The specific process is as follows:

* The DSC in the PN side registers into the TSE. The TSE registers into the IMC. Both provide their IP addresses, the data type, the data source, the data size, etc.

* The DSC in the NDT side sends the data collection request to the IMC.

* According to the data collection request, the IMC intelligently queries the registration addressing information and configures the data processing instruction.

* The IMC in the NDT side sends the corresponding instruction according to the query result to the TSE.

* After receiving the instructions, the TSE parses them and executes them. The query function can be performed by the CEP engine, which receives all monitoring information (telemetry data) and processes it with all queries provided.

* The TSE sends data subscription to DSC in the PN side.

* The DSC in the PN side represents the data semantically in RDF form or sends the data in raw form to the TSE for it to make the semantic representation.

* The DSC in the PN side pushes the data or knowledge item to the TSE.

* The TSE aggregates and correlates the collected data or knowledge items. Then, according to the actual needs, generates aggregated data or knowledge items.

* The TSE sends the resulting data or knowledge items to the DSC in the NDT side.

5.4. Query and Aggregation Functions

The TSE supports an arbitrary number of queries and aggregation functions. As a minimum, it will support:

* A function to apply a particular calculation to the values retrieved from a specified metric for a specified period of time. The basically supported calculations must be:

  - Average: Returns the single number resulting from averaging all values in the period.

  - Maximum: Returns the single number that represents the highest value in the period.

  - Minimum: Returns the single number that represents the lowest value in the period.

- Percentile X: Returns the percentile of calculated at position X (from 0, which is the minimum, to 100, which is the maximum).

- Moving Average X: Transforms all values of the specified period by calculating every value as the average of the previous X values (or less if there are not enough).

- Filter Previous X: Removes the values that change less than X percent from the previous value.

- Filter Average X: Removes the values that change less than X percent from the average value.

- Filter Moving Average X Y: Removes the values that change less than Y percent from the value of the moving average for X previous values.

* A function to represent the collected values in a semantic structure following some ontology, information model, and data format (YANG). This will enforce semantic constraints to the values, such as avoiding negative measures of some parameters (e.g., bandwidth usage).

* A function to analyze the collected values to detect some pattern (provided) and, if so, trigger some notification that other module can use to execute some action.

The particular behavior of the three functions will be described in a high-level language that is transformed to the specific code used by the device, such as [P4].

6. Summary

This draft describes the requirements for data collection and provides the data collection methods or tools required to build the data repository for maintaining NDT systems. These data collection methods or tools should meet the requirement of target-driven, diversity, lightweight and efficiency, while being open and standardized. Among all the requirements, lightweight and efficiency requirements are the most important. Thus, this draft provides a lightweight and efficient method for data collection that is particularly optimized for maintaining NDT systems. Going forward, more methods (transformation and aggregation functions) and tools (solutions) shall be studied to extend the contents of this draft.

7. Security Considerations

TBD.

8. IANA Considerations

This document has no requests to IANA.

9. References

9.1. Normative References

[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>.

[RFC8639] Voit, E., Clemm, A., Gonzalez Prieto, A., Nilsen-Nygaard, E., and A. Tripathy, "Subscription to YANG Notifications", RFC 8639, DOI 10.17487/RFC8639, September 2019, <https://www.rfc-editor.org/info/rfc8639>.

[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>.

[RFC9232] Song, H., Qin, F., Martinez-Julia, P., Ciavaglia, L., and A. Wang, "Network Telemetry Framework", RFC 9232, DOI 10.17487/RFC9232, May 2022, <https://www.rfc-editor.org/info/rfc9232>.

9.2. Informative References

[I-D.ietf-opsawg-collected-data-manifest] Claise, B., Quilbeuf, J., Lopez, D., Martinez-Casanueva, I. D., and T. Graf, "A Data Manifest for Contextualized Telemetry Data", Work in Progress, Internet-Draft, draft-ietf-opsawg-collected-data-manifest-10, 20 October 2025, <https://datatracker.ietf.org/doc/html/draft-ietf-opsawg-collected-data-manifest-10>.

[I-D.irtf-nmrg-network-digital-twin-arch]
Zhou, C., Yang, H., Duan, X., Lopez, D., Pastor, A., Wu, Q., Boucadair, M., and C. Jacquenet, "Network Digital Twin: Concepts and Reference Architecture", Work in Progress, Internet-Draft, draft-irtf-nmrg-network-digital-twin-arch-11, 6 July 2025, <https://datatracker.ietf.org/doc/html/draft-irtf-nmrg-network-digital-twin-arch-11>.

[P4] The P4 Language Consortium, "P4 Language Specification (https://p4.org/p4-spec/docs/P4-16-v-1.2.3.html)", 11 July 2022.

[RFC6241] Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed., and A. Bierman, Ed., "Network Configuration Protocol (NETCONF)", RFC 6241, DOI 10.17487/RFC6241, June 2011, <https://www.rfc-editor.org/info/rfc6241>.

[RFC7011] Claise, B., Ed., Trammell, B., Ed., and P. Aitken, "Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of Flow Information", STD 77, RFC 7011, DOI 10.17487/RFC7011, September 2013, <https://www.rfc-editor.org/info/rfc7011>.

[RFC7594] Eardley, P., Morton, A., Bagnulo, M., Burbridge, T., Aitken, P., and A. Akhter, "A Framework for Large-Scale Measurement of Broadband Performance (LMAP)", RFC 7594, DOI 10.17487/RFC7594, September 2015, <https://www.rfc-editor.org/info/rfc7594>.

[RFC8040] Bierman, A., Bjorklund, M., and K. Watsen, "RESTCONF Protocol", RFC 8040, DOI 10.17487/RFC8040, January 2017, <https://www.rfc-editor.org/info/rfc8040>.

[RFC9506] Cociglio, M., Ferrieux, A., Fioccola, G., Lubashev, I., Bulgarella, F., Nilo, M., Hamchaoui, I., and R. Sisto, "Explicit Host-to-Network Flow Measurements Techniques", RFC 9506, DOI 10.17487/RFC9506, October 2023, <https://www.rfc-editor.org/info/rfc9506>.

Authors’ Addresses

Cheng Zhou
China Mobile
Beijing
100053
China
Email: zhouchengyjy@chinamobile.com

Danyang Chen
China Mobile
Beijing
100053
China
Email: chendanyang@chinamobile.com

Pedro Martinez-Julia
NICT
4-2-1, Nukui-Kitamachi, Koganei, Tokyo
184-8795
Japan
Email: pedro@nict.go.jp

Qiufang Ma
Huawei
Nanjing
210012
China
Email: maqiufang1@huawei.com