OPSAWG
Internet-Draft
Intended status: Informational
Expires: March 25, 2021

H. Song
Futurewei
F. Qin
China Mobile
P. Martinez-Julia
NICT
L. Ciavaglia
Nokia
A. Wang
China Telecom

September 21, 2020

Network Telemetry Framework
draft-ietf-opsawg-ntf-04

Abstract

Network telemetry is the technology for gaining network insight and facilitating efficient and automated network management. It engages various techniques for remote data collection, correlation, and consumption. This document provides an architectural framework for network telemetry, motivated by the network operation challenges and requirements. As evidenced by some key characteristics and industry practices, network telemetry covers technologies and protocols beyond the conventional network Operations, Administration, and Management (OAM). It promises better flexibility, scalability, accuracy, coverage, and performance and allows automated control loops to suit both today's and tomorrow's network operation. This document clarifies the terminologies and classifies the modules and components of a network telemetry system from several different perspectives. The framework and taxonomy help to set a common ground for the collection of related work and provide guidance for related technique and standard developments.

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 March 25, 2021.

Table of Contents

1. Introduction

Network visibility is the ability of management tools to see the state and behavior of a network. It is essential for successful network operation. Network telemetry is the process of measuring, correlating, recording, and distributing information about the behavior of a network. Network telemetry has been considered as an ideal means to gain sufficient network visibility with better flexibility, scalability, accuracy, coverage, and performance than some conventional network Operations, Administration, and Management (OAM) techniques.

However, the term of network telemetry lacks a solid and unambiguous definition. The scope and coverage of it cause confusion and misunderstandings. It is beneficial to clarify the concept and provide a clear architectural framework for network telemetry, so we can articulate the technical field, and better align the related techniques and standard works.

To fulfill such an undertaking, we first discuss some key characteristics of network telemetry which set a clear distinction from the conventional network OAM and show that some conventional OAM technologies can be considered a subset of the network telemetry technologies. We then provide an architectural framework for network telemetry by partitioning a network telemetry system into four modules each with the same building components and data abstracts. We show how the network telemetry framework can benefit the current and future network operations. Based on the distinction of modules and function components, we can map the existing and emerging techniques and protocols into the framework. The framework can also simplify the tasks for designing, maintaining, and understanding a network telemetry system. At last, we outline the evolution stages of the network telemetry system and discuss the potential security concerns.

The purpose of the framework and taxonomy is to set a common ground for the collection of related work and provide guidance for future technique and standard developments. To the best of our knowledge, this document is the first such effort for network telemetry in industry standards organizations.

2. Background

The term "big data" is used to describe the extremely large volume of data sets that can be analyzed computationally to reveal patterns, trends, and associations. Network is undoubtedly a source of big data because of its scale and all the traffic goes through it. It is easy to see that network OAM can benefit from network big data.

Today one can access advanced big data analytics capability through a plethora of commercial and open source platforms (e.g., Apache Hadoop), tools (e.g., Apache Spark), and techniques (e.g., machine learning). Thanks to the advance of computing and storage technologies, network big data analytics gives network operators an opportunity to gain network insights and move towards network autonomy. Some operators start to explore the application of Artificial Intelligence (AI) to make sense of network data. Software tools can use the network data to detect and react on network faults, anomalies, and policy violations, as well as predicting future events. In turn, the network policy updates for planning, intrusion prevention, optimization, and self-healing may be applied.

It is conceivable that an autonomic network [RFC7575] is the logical next step for network evolution following Software Defined Network (SDN), aiming to reduce (or even eliminate) human labor, make more efficient use of network resources, and provide better services more aligned with customer requirements. Intent-based Networking (IBN) [I-D.irtf-nmrg-ibn-concepts-definitions] provides the necessary mechanisms. Although it takes time to reach the ultimate goal, the journey has started nevertheless.

However, while the data processing capability is improved and applications are hungry for more data, the networks lag behind in extracting and translating network data into useful and actionable information in efficient ways. The system bottleneck is shifting from data consumption to data supply. Both the number of network nodes and the traffic bandwidth keep increasing at a fast pace. The network configuration and policy change at smaller time slots than before. More subtle events and fine-grained data through all network planes need to be captured and exported in real time. In a nutshell, it is a challenge to get enough high-quality data out of network efficiently, timely, and flexibly. Therefore, we need to examine the existing network technologies and protocols, and identify any

potential technique and standard gaps based on the real network and device architectures.

In the remaining of this section, first we clarify the scope of network data (i.e., telemetry data) concerned in the context. Then, we discuss several key use cases for today's and future network operations. Next, we show why the current network OAM techniques and protocols are insufficient for these use cases. The discussion underlines the need of new methods, techniques, and protocols which we assign under an umbrella term - network telemetry.

2.1. Telemetry Data Coverage

Any information that can be extracted from networks (including data plane, control plane, and management plane) and used to gain visibility or as basis for actions is considered telemetry data. It includes statistics, event records and logs, snapshots of state, configuration data, etc. It also covers the outputs of any active and passive measurements. Specially, raw data can be processed in network before sending to a data consumer. Such processed data are also telemetry data in the context. A classification of the telemetry data form is provided in Section 4.

2.2. Use Cases

These use cases are essential for network operations. While the list is by no means exhaustive, it is enough to highlight the requirements for data velocity, variety, volume, and veracity in networks.

Security: Network intrusion detection and prevention need monitor network traffic and activities, and act upon anomalies. Given the more and more sophisticated attack vector and higher and higher tolls due to security breach, new tools and techniques need to be developed, relying on wider and deeper visibility in networks.

Policy and Intent Compliance: Network policies are the rules that constraint the services for network access, provide service differentiation, or enforce specific treatment on the traffic. For example, a service function chain is a policy that requires the selected flows to pass through a set of ordered network functions. Intent, as defined in [I-D.irtf-nmrg-ibn-concepts-definitions], is a set of operational goal that a network should meet and outcomes that a network is supposed to deliver, defined in a declarative manner without specifying how to achieve or implement them. An intent requires a complex translation and mapping process before being applied on networks. While a policy or an intent is enforced, the compliance

needs to be verified and monitored continuously, and any violation needs to be reported immediately.

SLA Compliance: A Service-Level Agreement (SLA) defines the level of service a user expects from a network operator, which include the metrics for the service measurement and remedy/penalty procedures when the service level misses the agreement. Users need to check if they get the service as promised and network operators need to evaluate how they can deliver the services that can meet the SLA based on realtime network measurement.

Root Cause Analysis: Any network failure can be the cause or effect of a sequence of chained events. Troubleshooting and recovery require quick identification of the root cause of any observable issues. However, the root cause is not always straightforward to identify, especially when the failure is sporadic and the related and unrelated events are overwhelming and interleaved. While machine learning technologies can be used for root cause analysis, it up to the network to sense and provide the relevant data.

Network Optimization: This covers all short-term and long-term network optimization techniques, including load balancing, Traffic Engineering (TE), and network planning. Network operators are motivated to optimize their network utilization and differentiate services for better Return On Investment (ROI) or lower Capital Expenditures (CAPEX). The first step is to know the real-time network conditions before applying policies for traffic manipulation. In some cases, micro-bursts need to be detected in a very short time-frame so that fine-grained traffic control can be applied to avoid network congestion. The long-term network capacity planning and topology augmentation rely on the accumulated data of network operations.

Event Tracking and Prediction: The visibility of traffic path and performance is critical for services and applications that rely on healthy network operation. Numerous related network events are of interest to network operators. For example, Network operators want to learn where and why packets are dropped for an application flow. They also want to be warned of issues in advance so proactive actions can be taken to avoid catastrophic consequences.

2.3. Challenges

For a long time, network operators have relied upon SNMP [RFC3416], Command-Line Interface (CLI), or Syslog to monitor the network. Some other OAM techniques as described in [RFC7276] are also used to facilitate network troubleshooting. these conventional techniques are not sufficient to support the above use cases for the following

reasons, which explains why new standards and techniques keep emerging and the needs remain high:

- Most use cases need to continuously monitor the network and dynamically refine the data collection in real-time. The poll-based low-frequency data collection is ill-suited for these applications. Subscription-based streaming data directly pushed from the data source (e.g., the forwarding chip) is preferred to provide enough data quantity and precision at scale.

- Comprehensive data is needed from packet processing engine to traffic manager, from line cards to main control board, from user flows to control protocol packets, from device configurations to operations, and from physical layer to application layer. Conventional OAM only covers a narrow range of data (e.g., SNMP only handles data from the Management Information Base (MIB)). Traditional network devices cannot provide all the necessary probes. More open and programmable network devices are therefore needed.

- Many application scenarios need to correlate network-wide data from multiple sources (i.e., from distributed network devices, different components of a network device, or different network planes). A piecemeal solution is often lacking the capability to consolidate the data from multiple sources. The composition of a complete solution, as partly proposed by Autonomic Resource Control Architecture(ARCA) [I-D.pedro-nmrg-anticipated-adaptation], will be empowered and guided by a comprehensive framework.

- Some of the conventional OAM techniques (e.g., CLI and Syslog) lack a formal data model. The unstructured data hinder the tool automation and application extensibility. Standardized data models are essential to support the programmable networks.

- Although some conventional OAM techniques support data push (e.g., SNMP Trap [RFC2981][RFC3877], Syslog, and sFlow), the pushed data are limited to only predefined management plane warnings (e.g., SNMP Trap) or sampled user packets (e.g., sFlow). Network operators require the data with arbitrary source, granularity, and precision which are beyond the capability of the existing techniques.

- The conventional passive measurement techniques can either consume excessive network resources and render excessive redundant data, or lead to inaccurate results; on the other hand, the conventional active measurement techniques can interfere with the user traffic

and their results are indirect. Techniques that can collect direct and on-demand data from user traffic are more favorable.

2.4. Glossary

Before further discussion, we list some key terminology and acronyms used in this documents. We make an intended distinction between network telemetry and network OAM.

AI: Artificial Intelligence. In network domain, AI refers to the machine-learning based technologies for automated network operation and other tasks.

AM: Alternate Marking, a flow performance measurement method, specified in [RFC8321].

BMP: BGP Monitoring Protocol, specified in [RFC7854].

DNP: Dynamic Network Probe, referring to programmable in-network sensors for network monitoring and measurement.

DPI: Deep Packet Inspection, referring to the techniques that examines packet beyond packet L3/L4 headers.

gNMI: gRPC Network Management Interface, a network management protocol from OpenConfig Operator Working Group, mainly contributed by Google. See [gnmi] for details.

gRPC: gRPC Remote Procedure Call, a open source high performance RPC framework that gNMI is based on. See [grpc] for details.

IPFIX: IP Flow Information Export Protocol, specified in [RFC7011].

IOAM: In-situ OAM, a dataplane on-path telemetry technique.

NETCONF: Network Configuration Protocol, specified in [RFC6241].

NetFlow: A Cisco protocol for flow record collecting, described in [RFC3594].

Network Telemetry: Acquiring and processing network data remotely for network monitoring and operation. A general term for a large set of network visibility techniques and protocols, with the characteristics defined in this document. Network telemetry addresses the current network operation issues and enables smooth evolution toward future intent-driven autonomous networks.

NMS: Network Management System, referring to applications that allow network administrators manage a network's software and hardware components. It usually records data from a network's remote points to carry out central reporting to a system administrator.

OAM: Operations, Administration, and Maintenance. A group of network management functions that provide network fault indication, fault localization, performance information, and data and diagnosis functions. Most conventional network monitoring techniques and protocols belong to network OAM.

PBT: Postcard-Based Telemetry, a dataplane on-path telemetry technique.

SMIv2 Structure of Management Information Version 2, specified in [RFC2578].

SNMP: Simple Network Management Protocol. Version 1 and 2 are specified in [RFC1157] and [RFC3416], respectively.

YANG: The abbreviation of "Yet Another Next Generation". YANG is a data modeling language for the definition of data sent over network management protocols such as the NETCONF and RESTCONF. YANG is defined in [RFC6020].

YANG FSM: A YANG model that describes events, operations, and finite state machine of YANG-defined network elements.

YANG PUSH: A method to subscribe pushed data from remote YANG datastore on network devices. Details are specified in [RFC8641] and [RFC8639].

2.5. Network Telemetry

Network telemetry has emerged as a mainstream technical term to refer to the newer data collection and consumption techniques, distinguishing itself from the convention techniques for network OAM. Many such techniques have been widely deployed. The representative techniques and protocols include IPFIX [RFC7011] and gPRC [grpc]. Network telemetry allows separate entities to acquire data from network devices so that data can be visualized and analyzed to support network monitoring and operation. Network telemetry overlaps with the conventional network OAM and has a wider scope than it. It is expected that network telemetry can provide the necessary network insight for autonomous networks and address the shortcomings of conventional OAM techniques.

One difference between the network telemetry and the network OAM is that in general the network telemetry assumes machines as data consumer rather than human operators. Hence, the network telemetry can directly trigger the automated network operation, while the conventional OAM tools usually help human operators to monitor and diagnose the networks and guide manual network operations. The difference leads to very different techniques.

Although the network telemetry techniques are just emerging and subject to continuous evolution, several characteristics of network telemetry have been well accepted. Note that network telemetry is intended to be an umbrella term covering a wide spectrum of techniques, so the following characteristics are not expected to be held by every specific technique.

- Push and Streaming: Instead of polling data from network devices, the telemetry collector subscribes to the streaming data pushed from data sources in network devices.

- Volume and Velocity: The telemetry data is intended to be consumed by machines rather than by human being. Therefore, the data volume is huge and the processing is often in realtime.

- Normalization and Unification: Telemetry aims to address the overall network automation needs. The piecemeal solutions offered by the conventional OAM approach are no longer suitable. Efforts need to be made to normalize the data representation and unify the protocols.

- Model-based: The telemetry data is modeled in advance which allows applications to configure and consume data with ease.

- Data Fusion: The data for a single application can come from multiple data sources (e.g., cross-domain, cross-device, and cross-layer) and needs to be correlated to take effect.

- Dynamic and Interactive: Since the network telemetry means to be used in a closed control loop for network automation, it needs to run continuously and adapt to the dynamic and interactive queries from the network operation controller.

In addition, an ideal network telemetry solution may also have the following features or properties:

- In-Network Customization: The data can be customized in network at run-time to cater to the specific need of applications. This needs the support of a programmable data plane which allows probes with custom functions to be deployed at flexible locations.

- In-Network Data Aggregation and Correlation: Network devices and aggregation points can work out which events and what data needs to be stored, reported, or discarded thus reducing the load on the central collection and processing points while still ensuring that the right information is ready to be processed in a timely way.

- In-Network Processing: Sometimes it is not necessary or feasible to gather all information to a central point to be processed and acted upon. It is possible for the data processing to be done in network, allowing reactive actions to be taken locally.

- Direct Data Plane Export: The data originated from the data plane forwarding chips can be directly exported to the data consumer for efficiency, especially when the data bandwidth is large and the real-time processing is required.

- In-band Data Collection: In addition to the passive and active data collection approaches, the new hybrid approach allows to directly collect data for any target flow on its entire forwarding path [I-D.song-opsawg-ifit-framework].

It is worth noting that, a network telemetry system should not be intrusive to normal network operations, by avoiding the pitfall of the "observer effect". That is, it should not change the network behavior and affect the forwarding performance. Otherwise, the whole purpose of network telemetry is defied.

Although in many cases a network telemetry system involves a remote data collecting, processing, and reacting entity, it is important to understand that network telemetry does not infer the necessity of such an entity. Telemetry data producers and consumers can work in distributed or peer-to-peer fashions instead. In such cases, a network node can be the direct consumer of telemetry data from other nodes.

3. The Necessity of a Network Telemetry Framework

Network data analytics and machine-learning technologies are applied for network operation automation, relying on abundant and coherent data from networks. The single-sourced and static data acquisition cannot meet the data requirements. The scattered standards and diverse techniques are hard to be integrated. It is desirable to have a framework that classifies and organizes different telemetry data source and types, defines different components of a network telemetry system and their interactions, and helps coordinate and integrate multiple telemetry approaches from different layers. This allows flexible combinations for different applications, while

normalizing and simplifying interfaces. In detail, such a framework would benefit application development for the following reasons:

- The future autonomous networks will require a holistic view on network visibility. All the use cases and applications need to be supported uniformly and coherently under a single intelligent agent. Therefore, the protocols and mechanisms should be consolidated into a minimum yet comprehensive set. A telemetry framework can help to normalize the technique developments.

- Network visibility presents multiple viewpoints. For example, the device viewpoint takes the network infrastructure as the monitoring object from which the network topology and device status can be acquired; the traffic viewpoint takes the flows or packets as the monitoring object from which the traffic quality and path can be acquired. An application may need to switch its viewpoint during operation. It may also need to correlate a service and its impact on network experience to acquire the comprehensive information.

- Applications require network telemetry to be elastic in order to efficiently use the network resource and reduce the performance impact. Routine network monitoring covers the entire network with low data sampling rate. When issues arise or trends emerge, the telemetry data source can be modified and the data rate can be boosted.

- Efficient data fusion is critical for applications to reduce the overall quantity of data and improve the accuracy of analysis.

A telemetry framework collects together all of the telemetry-related works from different sources and working groups within IETF. This makes it possible to assemble a comprehensive network telemetry system and to avoid repetitious or redundant work. The framework should cover the concepts and components from the standardization perspective. This document clarifies the layered modules on which the telemetry is exerted and decomposes the telemetry system into a set of distinct components that the existing and future work can easily map to.

4. Network Telemetry Framework

The top level network telemetry framework partitions the network telemetry into four modules based on the telemetry data object source and represents their relationship. The next level framework reveals that each module replicates the same architecture comprising the same set of components. Throughout the framework, the same set of abstract data acquiring mechanisms and data types are applied. The

two-level architecture with the uniform data abstraction helps accurately pinpoint a protocol or technique to its position in a network telemetry system or disaggregate a network telemetry system into manageable parts.

4.1. Top Level Modules

Telemetry can be applied on the forwarding plane, the control plane, and the management plane in a network, as well as other sources out of the network, as shown in Figure 1. Therefore, we categorize the network telemetry into four distinct modules with each having its own interface to Network Operation Applications.

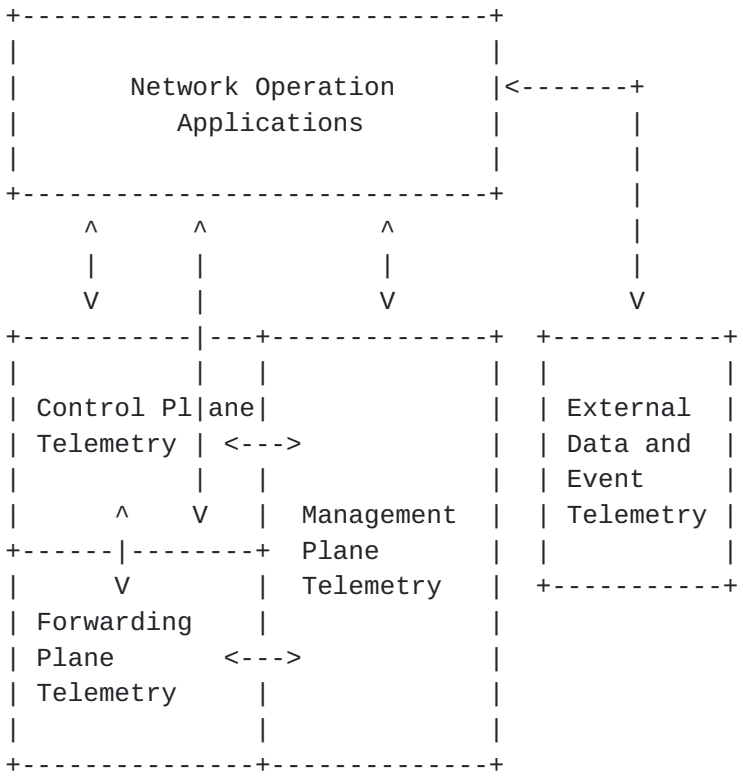

Figure 1: Modules in Layer Category of NTF

The rationale of this partition lies in the different telemetry data objects which result in different data source and export locations. Such differences have profound implications on in-network data programming and processing capability, data encoding and transport protocol, and data bandwidth and latency.

We summarize the major differences of the four modules in the following table. They are compared from six aspects: data object, data export location, data model, data encoding, telemetry protocol,

and transport method. Data object is the target and source of each module. Because the data source varies, the data export location varies. Because each data export location has different capability, the proper data model, encoding, and transport method cannot be kept the same. As a result, the suitable telemetry protocol for each module can be different. Some representative techniques are shown in the corresponding table blocks to highlight the technical diversity of these modules. The key point is that one cannot expect to use a universal protocol to cover all the network telemetry requirements.

Module	Control Plane	Management Plane	Forwarding Plane	External Data
Object	control protocol & signaling, RIB, ACL	config. & operation state, MIB	flow & packet QoS, traffic stat., buffer & queue stat.	terminal, social & environ- mental
Export Location	main control CPU, linecard CPU or fwding chip	main control CPU	fwding chip or linecard CPU; main control CPU unlikely	various
Data Model	YANG, custom	MIB, syslog, YANG, custom	template, YANG, custom	YANG
Data Encoding	GPB, JSON, XML, plain	GPB, JSON, XML	plain	GPB, JSON XML, plain
Protocol	gRPC,NETCONF, IPFIX,mirror	gPRC,NETCONF,	IPFIX, mirror	gRPC
Transport	HTTP, TCP, UDP	HTTP, TCP	UDP	HTTP,TCP UDP

Figure 2: Comparison of the Data Object Modules

Note that the interaction with the network operation applications can be indirect. Some in-device data transfer is possible. For example, in the management plane telemetry, the management plane may need to acquire data from the data plane. Some of the operational states can only be derived from the data plane such as the interface status and

statistics. For another example, the control plane telemetry may need to access the Forwarding Information Base (FIB) in data plane.

On the other hand, an application may involve more than one plane and interact with multiple planes simultaneously. For example, an SLA compliance application may require both the data plane telemetry and the control plane telemetry.

The requirements and challenges for each module are summarized as follows.

4.1.1. Management Plane Telemetry

The management plane of network elements interacts with the Network Management System (NMS), and provides information such as performance data, network logging data, network warning and defects data, and network statistics and state data. Some legacy protocols, such as SNMP and Syslog, are widely used for the management plane. However, these protocols are insufficient to meet the requirements of the future automated network operation applications.

New management plane telemetry protocols should consider the following requirements:

Convenient Data Subscription: An application should have the freedom to choose the data export means such as the data types and the export frequency.

Structured Data: For automatic network operation, machines will replace human for network data comprehension. The schema languages such as YANG can efficiently describe structured data and normalize data encoding and transformation.

High Speed Data Transport: In order to retain the information, a server needs to send a large amount of data at high frequency. Compact encoding formats are needed to compress the data and improve the data transport efficiency. The subscription mode, by replacing the query mode, reduces the interactions between clients and servers and helps to improve the server's efficiency.

4.1.2. Control Plane Telemetry

The control plane telemetry refers to the health condition monitoring of different network control protocols covering Layer 2 to Layer 7. Keeping track of the running status of these protocols is beneficial for detecting, localizing, and even predicting various network issues, as well as network optimization, in real-time and in fine granularity.

One of the most challenging problems for the control plane telemetry is how to correlate the End-to-End (E2E) Key Performance Indicators (KPI) to a specific layer's KPIs. For example, an IPTV user may describe his User Experience (UE) by the video fluency and definition. Then in case of an unusually poor UE KPI or a service disconnection, it is non-trivial to delimit and pinpoint the issue in the responsible protocol layer (e.g., the Transport Layer or the Network Layer), the responsible protocol (e.g., ISIS or BGP at the Network Layer), and finally the responsible device(s) with specific reasons.

Traditional OAM-based approaches for control plane KPI measurement include PING (L3), Tracert (L3), Y.1731 (L2), and so on. One common issue behind these methods is that they only measure the KPIs instead of reflecting the actual running status of these protocols, making them less effective or efficient for control plane troubleshooting and network optimization.

An example of the control plane telemetry is the BGP monitoring protocol (BMP), it is currently used to monitoring the BGP routes and enables rich applications, such as BGP peer analysis, AS analysis, prefix analysis, security analysis, and so on. However, the monitoring of other layers, protocols and the cross-layer, cross-protocol KPI correlations are still in their infancy (e.g., the IGP monitoring is missing), which require further research.

4.1.3. Data Plane Telemetry

An effective data plane telemetry system relies on the data that the network device can expose. The data's quality, quantity, and timeliness must meet some stringent requirements. This raises some challenges to the network data plane devices where the first hand data originate.

- A data plane device's main function is user traffic processing and forwarding. While supporting network visibility is important, the telemetry is just an auxiliary function, and it should not impede normal traffic processing and forwarding (i.e., the performance is not lowered and the behavior is not altered due to the telemetry functions).

- The network operation applications requires end-to-end visibility from various sources, which results in a huge volume of data. However, the sheer data quantity should not stress the network bandwidth, regardless of the data delivery approach (i.e., through in-band or out-of-band channels).

- The data plane devices must provide timely data with the minimum possible delay. Long processing, transport, storage, and analysis delay can impact the effectiveness of the control loop and even render the data useless.

- The data should be structured and labeled, and easy for applications to parse and consume. At the same time, the data types needed by applications can vary significantly. The data plane devices need to provide enough flexibility and programmability to support the precise data provision for applications.

- The data plane telemetry should support incremental deployment and work even though some devices are unaware of the system. This challenge is highly relevant to the standards and legacy networks.

The data plane programmability is essential to support network telemetry. Newer data plane forwarding chips are equipped with advanced telemetry features and provide flexibility to support customized telemetry functions.

4.1.3.1. Technique Taxonomy

There can be multiple possible dimensions to classify the data plane telemetry techniques.

Active, Passive, and Hybrid: The active and passive methods (as well as the hybrid types) are well documented in [RFC7799]. The passive methods include TCPDUMP, IPFIX [RFC7011], sflow, and traffic mirror. These methods usually have low data coverage. The bandwidth cost is very high in order to improve the data coverage. On the other hand, the active methods include Ping, Traceroute, OWAMP [RFC4656], TWAMP [RFC5357], and Cisco's SLA Protocol [RFC6812]. These methods are intrusive and only provide indirect network measurement results. The hybrid methods, including in-situ OAM [I-D.ietf-ippm-ioam-data], IPFPM [RFC8321], and Multipoint Alternate Marking [I-D.fioccola-ippm-multipoint-alt-mark], provide a well-balanced and more flexible approach. However, these methods are also more complex to implement.

In-Band and Out-of-Band: The telemetry data, before being exported to some collector, can be carried in user packets. Such methods are considered in-band (e.g., in-situ OAM [I-D.ietf-ippm-ioam-data]). If the telemetry data is directly exported to some collector without modifying the user packets, such methods are considered out-of-band (e.g., postcard-based INT). It is possible to have hybrid methods. For example, only

the telemetry instruction or partial data is carried by user packets (e.g., IPFPM [RFC8321]).

E2E and In-Network: Some E2E methods start from and end at the network end hosts (e.g., Ping). The other methods work in networks and are transparent to end hosts. However, if needed, the in-network methods can be easily extended into end hosts.

Information Type: Depending on the telemetry objective, the methods can be flow-based (e.g., in-situ OAM [I-D.ietf-ippm-ioam-data]), path-based (e.g., Traceroute), and node-based (e.g., IPFIX [RFC7011]). The various data objects can be packet, flow record, measurement, states, and signal.

4.1.4. External Data Telemetry

Events that occur outside the boundaries of the network system are another important source of network telemetry. Correlating both internal telemetry data and external events with the requirements of network systems, as presented in [I-D.pedro-nmrg-anticipated-adaptation], provides a strategic and functional advantage to management operations.

As with other sources of telemetry information, the data and events must meet strict requirements, especially in terms of timeliness, which is essential to properly incorporate external event information to management cycles. The specific challenges are described as follows:

- The role of external event detector can be played by multiple elements, including hardware (e.g. physical sensors, such as seismometers) and software (e.g. Big Data sources that analyze streams of information, such as Twitter messages). Thus, the transmitted data must support different shapes but, at the same time, follow a common but extensible schema.

- Since the main function of the external event detectors is to perform the notifications, their timeliness is assumed. However, once messages have been dispatched, they must be quickly collected and inserted into the control plane with variable priority, which will be high for important sources and/or important events and low for secondary ones.

- The schema used by external detectors must be easily adopted by current and future devices and applications. Therefore, it must be easily mapped to current information models, such as in terms of YANG.

Organizing together both internal and external telemetry information will be key for the general exploitation of the management possibilities of current and future network systems, as reflected in the incorporation of cognitive capabilities to new hardware and software (virtual) elements.

4.2. Second Level Function Components

Reflecting the best current practice, the telemetry module at each plane is further partitioned into five distinct components:

Data Query, Analysis, and Storage: This component works at the application layer. On the one hand, it is responsible for issuing data requirements. The data of interest can be modeled data through configuration or custom data through programming. The data requirements can be queries for one-shot data or subscriptions for events or streaming data. On the other hand, it receives, stores, and processes the returned data from network devices. Data analysis can be interactive to initiate further data queries. This component can reside in either network devices or remote controllers.

Data Configuration and Subscription: This component deploys data queries on devices. It determines the protocol and channel for applications to acquire desired data. This component is also responsible for configuring the desired data that might not be directly available form data sources. The subscription data can be described by models, templates, or programs.

Data Encoding and Export: This component determines how telemetry data are delivered to the data analysis and storage component. The data encoding and the transport protocol may vary due to the data exporting location.

Data Generation and Processing: The requested data needs to be captured, processed, and formatted in network devices from raw data sources. This may involve in-network computing and processing on either the fast path or the slow path in network devices.

Data Object and Source: This component determines the monitoring object and original data source. The data source usually just provides raw data which needs further processing. A data source can be considered a probe. A probe can be statically installed or dynamically installed.

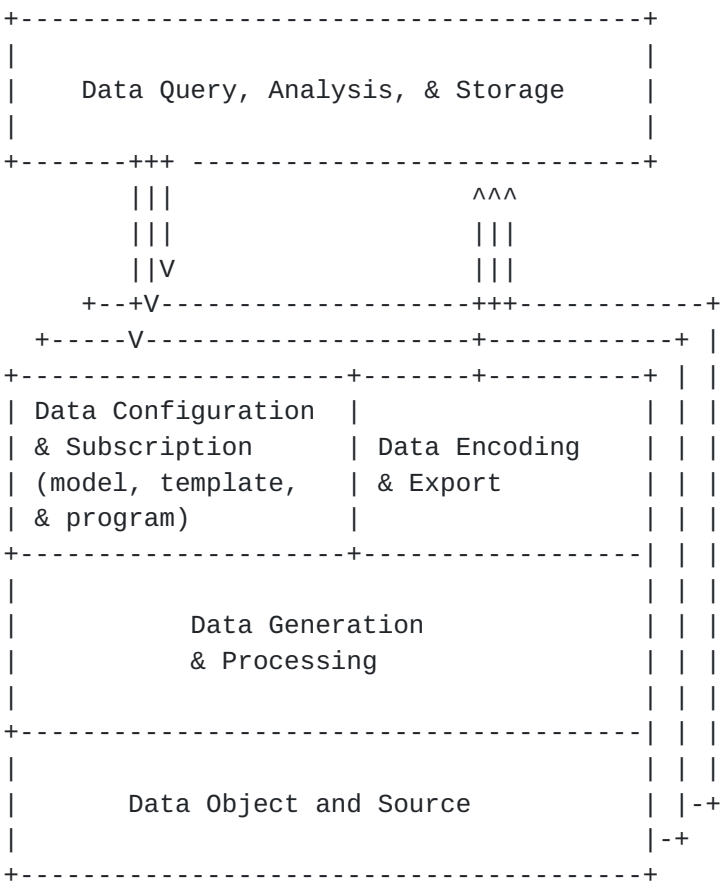

Figure 3: Components in the Network Telemetry Framework

4.3. Data Acquiring Mechanism and Type Abstraction

Broadly speaking, network data can be acquired through subscription (push) and query (poll). Subscription is a contract between publisher and subscriber. After initial setup, the subscribed data is automatically delivered to registered subscribers until the subscription expires. Subscription can be partitioned into two sub modes: the Publish-Subscription (Pub-Sub) mode and the Subscription-Publish (Sub-Pub) mode. In the Pub-Sub mode, a publisher publishes pre-defined data and any qualified subscribers can subscribe the data as-is. In the Sub-Pub mode, a subscriber initiates a data request and sends it to a publisher; the publisher will deliver the requested data when available.

In contrast, query is used when a querier expects immediate and one-off feedback from network devices. The queried data may be directly extracted from some specific data source, or synthesized and processed from raw data. Query suits for interactive network telemetry applications.

There are four types of data from network devices:

Simple Data: The data that are steadily available from some data store or static probes in network devices. such data can be specified by YANG model.

Complex Data: The data need to be synthesized or processed in network from raw data from one or more network devices. The data processing function can be statically or dynamically loaded into network devices.

Event-triggered Data: The data are conditionally acquired based on the occurrence of some events. An event can be modeled as a Finite State Machine (FSM).

Streaming Data: The data are continuously or periodically generated. It can be time series or the dump of databases. The streaming data reflect realtime network states and metrics and require large bandwidth and processing power.

The above data types are not mutually exclusive. For example, event-triggered data can be simple or complex, and streaming data can be simple, complex, or triggered by events. The relationships of these data types are illustrated in Figure 4.

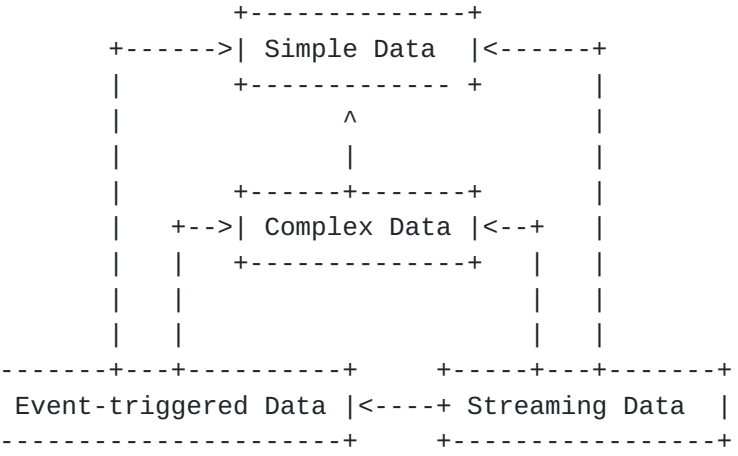

Figure 4: Data Type Relationship

Subscription usually deals with event-triggered data and streaming data, and query usually deals with simple data and complex data. But the other ways are also possible. The conventional OAM techniques are mostly about querying simple data. While these techniques are still useful, more advanced network telemetry techniques are designed

mainly for event-triggered or streaming data subscription, and complex data query.

4.4. Existing Works Mapped in the Framework

The following two tables provide a non-exhaustive list of existing works (mainly published in IETF and with the emphasis on the latest new technologies) and shows their positions in the framework. More details can be found in Appendix A.

The first table is based on the data acquiring mechanisms and data types.

	Query	Subscription
Simple Data	SNMP, NETCONF, YANG, BMP, SMIv2, gRPC	SNMP, NETCONF YANG, gRPC
Complex Data	DNP, YANG FSM gRPC, NETCONF	DNP, YANG PUSH gPRC, NETCONF
Event-triggered Data	N/A	gRPC, NETCONF, YANG PUSH, DNP YANG FSM
Streaming Data	N/A	gRPC, NETCONF, IOAM, PBT, DNP IPFIX, IPFPM

Figure 5: Existing Work Mapping I

The second table is based on the telemetry modules and components.

	Management Plane	Control Plane	Forwarding Plane
data config. & subscribe	gRPC, NETCONF, SMIv2,YANG PUSH	NETCONF/YANG	NETCONF/YANG, YANG FSM
data gen. & process	DNP, YANG	DNP, YANG	IOAM, PBT, IPFPM, DNP
data export	gRPC, NETCONF YANG PUSH	BMP, NETCONF	IPFIX

Figure 6: Existing Work Mapping II

5. Evolution of Network Telemetry

Network telemetry is a fast evolving technical area. As the network moves towards the automated operation, network telemetry undergoes several stages of evolution. Each stage is built upon the techniques enabled by previous stages.

Stage 0 - Static Telemetry: The telemetry data source and type are determined at design time. The network operator can only configure how to use it with limited flexibility.

Stage 1 - Dynamic Telemetry: The custom telemetry data can be dynamically programmed or configured at runtime, allowing a tradeoff among resource, performance, flexibility, and coverage. DNP is an effort towards this direction.

Stage 2 - Interactive Telemetry: The network operator can continuously customize the telemetry data in real time to reflect the network operation's visibility requirements. At this stage, some tasks can be automated, although ultimately human operators will still need to sit in the middle to make decisions.

Stage 3 - Closed-loop Telemetry: Human operators are completely excluded from the control loop. The intelligent network operation engine automatically issues the telemetry data requests, analyzes the data, and updates the network operations in closed control loops.

The most of the existing technologies belong to stage 0 and stage 1. Individual stage 2 and stage 3 applications are also possible now.

However, the future autonomic networks may need a comprehensive operation management system which relies on stage 2 and stage 3 telemetry to cover all the network operation tasks. A well-defined network telemetry framework is the first step towards this direction.

6. Security Considerations

The complexity of network telemetry raises significant security implications. For example, telemetry data can be manipulated to exhaust various network resources at each plane as well as the data consumer; falsified or tampered data can mislead the decision making and paralyze networks; wrong configuration and programming for telemetry is equally harmful.

Given that this document has proposed a framework for network telemetry and the telemetry mechanisms discussed are distinct (in both message frequency and traffic amount) from the conventional network OAM concepts, we must also reflect that various new security considerations may also arise. A number of techniques already exist for securing the forwarding plane, the control plane, and the management plane in a network, but it is important to consider if any new threat vectors are now being enabled via the use of network telemetry procedures and mechanisms.

Security considerations for networks that use telemetry methods may include:

- Telemetry framework trust and policy model;
- Role management and access control for enabling and disabling telemetry capabilities;
- Protocol transport used telemetry data and inherent security capabilities;
- Telemetry data stores, storage encryption and methods of access;
- Tracking telemetry events and any abnormalities that might identify malicious attacks using telemetry interfaces.

Some of the security considerations highlighted above may be minimized or negated with policy management of network telemetry. In a network telemetry deployment it would be advantageous to separate telemetry capabilities into different classes of policies, i.e., Role Based Access Control and Event-Condition-Action policies. Also, potential conflicts between network telemetry mechanisms must be detected accurately and resolved quickly to avoid unnecessary network

telemetry traffic propagation escalating into an unintended or intended denial of service attack.

Further study of the security issues will be required, and it is expected that the secuirty mechanisms and protocols are devloped and deployed along with a network telemetry system.

7. IANA Considerations

This document includes no request to IANA.

8. Contributors

The other contributors of this document are listed as follows.

- Tianran Zhou
- Zhenbin Li
- Zhenqiang Li
- Daniel King
- Adrian Farrel
- Alexander Clemm

9. Acknowledgments

We would like to thank Randy Presuhn, Joe Clarke, Victor Liu, James Guichard, Uri Blumenthal, Giuseppe Fioccola, Yunan Gu, Parviz Yegani, Young Lee, Qin Wu, and many others who have provided helpful comments and suggestions to improve this document.

10. Informative References

[gnmi] "gNMI - gRPC Network Management Interface", <https://github.com/openconfig/reference/tree/master/rpc/gnmi>.

[grpc] "gPPC, A high performance, open-source universal RPC framework", <https://grpc.io>.

[I-D.fioccola-ippm-multipoint-alt-mark] Fioccola, G., Cociglio, M., Sapio, A., and R. Sisto, "Multipoint Alternate Marking method for passive and hybrid performance monitoring", draft-fioccola-ippm-multipoint-alt-mark-04 (work in progress), June 2018.

[I-D.ietf-grow-bmp-adj-rib-out]
Evens, T., Bayraktar, S., Lucente, P., Mi, K., and S. Zhuang, "Support for Adj-RIB-Out in BGP Monitoring Protocol (BMP)", draft-ietf-grow-bmp-adj-rib-out-07 (work in progress), August 2019.

[I-D.ietf-grow-bmp-local-rib]
Evens, T., Bayraktar, S., Bhardwaj, M., and P. Lucente, "Support for Local RIB in BGP Monitoring Protocol (BMP)", draft-ietf-grow-bmp-local-rib-07 (work in progress), May 2020.

[I-D.ietf-ippm-ioam-data]
Brockners, F., Bhandari, S., and T. Mizrahi, "Data Fields for In-situ OAM", draft-ietf-ippm-ioam-data-10 (work in progress), July 2020.

[I-D.ietf-netconf-udp-pub-channel]
Zheng, G., Zhou, T., and A. Clemm, "UDP based Publication Channel for Streaming Telemetry", draft-ietf-netconf-udp-pub-channel-05 (work in progress), March 2019.

[I-D.irtf-nmrg-ibn-concepts-definitions]
Clemm, A., Ciavaglia, L., Granville, L., and J. Tantsura, "Intent-Based Networking - Concepts and Definitions", draft-irtf-nmrg-ibn-concepts-definitions-02 (work in progress), September 2020.

[I-D.kumar-rtgwg-grpc-protocol]
Kumar, A., Kolhe, J., Ghemawat, S., and L. Ryan, "gRPC Protocol", draft-kumar-rtgwg-grpc-protocol-00 (work in progress), July 2016.

[I-D.openconfig-rtgwg-gnmi-spec]
Shakir, R., Shaikh, A., Borman, P., Hines, M., Lebsack, C., and C. Morrow, "gRPC Network Management Interface (gNMI)", draft-openconfig-rtgwg-gnmi-spec-01 (work in progress), March 2018.

[I-D.pedro-nmrg-anticipated-adaptation]
Martinez-Julia, P., "Exploiting External Event Detectors to Anticipate Resource Requirements for the Elastic Adaptation of SDN/NFV Systems", draft-pedro-nmrg-anticipated-adaptation-02 (work in progress), June 2018.

[I-D.song-ippm-postcard-based-telemetry]
Song, H., Zhou, T., Li, Z., Shin, J., and K. Lee, "Postcard-based On-Path Flow Data Telemetry", draft-song-ippm-postcard-based-telemetry-07 (work in progress), April 2020.

[I-D.song-opsawg-dnp4iq]
Song, H. and J. Gong, "Requirements for Interactive Query with Dynamic Network Probes", draft-song-opsawg-dnp4iq-01 (work in progress), June 2017.

[I-D.song-opsawg-ifit-framework]
Song, H., Qin, F., Chen, H., Jin, J., and J. Shin, "In-situ Flow Information Telemetry", draft-song-opsawg-ifit-framework-12 (work in progress), April 2020.

[I-D.zhou-netconf-multi-stream-originators]
Zhou, T., Zheng, G., Voit, E., and A. Clemm, "Subscription to Multiple Stream Originators", draft-zhou-netconf-multi-stream-originators-10 (work in progress), November 2019.

[RFC1157] Case, J., Fedor, M., Schoffstall, M., and J. Davin, "Simple Network Management Protocol (SNMP)", RFC 1157, DOI 10.17487/RFC1157, May 1990, <https://www.rfc-editor.org/info/rfc1157>.

[RFC2578] McCloghrie, K., Ed., Perkins, D., Ed., and J. Schoenwaelder, Ed., "Structure of Management Information Version 2 (SMIv2)", STD 58, RFC 2578, DOI 10.17487/RFC2578, April 1999, <https://www.rfc-editor.org/info/rfc2578>.

[RFC2981] Kavasseri, R., Ed., "Event MIB", RFC 2981, DOI 10.17487/RFC2981, October 2000, <https://www.rfc-editor.org/info/rfc2981>.

[RFC3416] Presuhn, R., Ed., "Version 2 of the Protocol Operations for the Simple Network Management Protocol (SNMP)", STD 62, RFC 3416, DOI 10.17487/RFC3416, December 2002, <https://www.rfc-editor.org/info/rfc3416>.

[RFC3594] Duffy, P., "PacketCable Security Ticket Control Sub-Option for the DHCP CableLabs Client Configuration (CCC) Option", RFC 3594, DOI 10.17487/RFC3594, September 2003, <https://www.rfc-editor.org/info/rfc3594>.

[RFC3877] Chisholm, S. and D. Romascanu, "Alarm Management Information Base (MIB)", RFC 3877, DOI 10.17487/RFC3877, September 2004, <https://www.rfc-editor.org/info/rfc3877>.

[RFC4656] Shalunov, S., Teitelbaum, B., Karp, A., Boote, J., and M. Zekauskas, "A One-way Active Measurement Protocol (OWAMP)", RFC 4656, DOI 10.17487/RFC4656, September 2006, <https://www.rfc-editor.org/info/rfc4656>.

[RFC5357] Hedayat, K., Krzanowski, R., Morton, A., Yum, K., and J. Babiarz, "A Two-Way Active Measurement Protocol (TWAMP)", RFC 5357, DOI 10.17487/RFC5357, October 2008, <https://www.rfc-editor.org/info/rfc5357>.

[RFC6020] Bjorklund, M., Ed., "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)", RFC 6020, DOI 10.17487/RFC6020, October 2010, <https://www.rfc-editor.org/info/rfc6020>.

[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>.

[RFC6812] Chiba, M., Clemm, A., Medley, S., Salowey, J., Thombare, S., and E. Yedavalli, "Cisco Service-Level Assurance Protocol", RFC 6812, DOI 10.17487/RFC6812, January 2013, <https://www.rfc-editor.org/info/rfc6812>.

[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>.

[RFC7276] Mizrahi, T., Sprecher, N., Bellagamba, E., and Y. Weingarten, "An Overview of Operations, Administration, and Maintenance (OAM) Tools", RFC 7276, DOI 10.17487/RFC7276, June 2014, <https://www.rfc-editor.org/info/rfc7276>.

[RFC7540] Belshe, M., Peon, R., and M. Thomson, Ed., "Hypertext Transfer Protocol Version 2 (HTTP/2)", RFC 7540, DOI 10.17487/RFC7540, May 2015, <https://www.rfc-editor.org/info/rfc7540>.

[RFC7575] Behringer, M., Pritikin, M., Bjarnason, S., Clemm, A., Carpenter, B., Jiang, S., and L. Ciavaglia, "Autonomic Networking: Definitions and Design Goals", RFC 7575, DOI 10.17487/RFC7575, June 2015, <https://www.rfc-editor.org/info/rfc7575>.

[RFC7799] Morton, A., "Active and Passive Metrics and Methods (with Hybrid Types In-Between)", RFC 7799, DOI 10.17487/RFC7799, May 2016, <https://www.rfc-editor.org/info/rfc7799>.

[RFC7854] Scudder, J., Ed., Fernando, R., and S. Stuart, "BGP Monitoring Protocol (BMP)", RFC 7854, DOI 10.17487/RFC7854, June 2016, <https://www.rfc-editor.org/info/rfc7854>.

[RFC8321] Fioccola, G., Ed., Capello, A., Cociglio, M., Castaldelli, L., Chen, M., Zheng, L., Mirsky, G., and T. Mizrahi, "Alternate-Marking Method for Passive and Hybrid Performance Monitoring", RFC 8321, DOI 10.17487/RFC8321, January 2018, <https://www.rfc-editor.org/info/rfc8321>.

[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>.

Appendix A. A Survey on Existing Network Telemetry Techniques

In this non-normative appendix, we provide an overview of some existing techniques and standard proposals for each network telemetry module.

A.1. Management Plane Telemetry

A.1.1. Push Extensions for NETCONF

NETCONF [RFC6241] is one popular network management protocol, which is also recommended by IETF. Although it can be used for data collection, NETCONF is good at configurations. YANG Push [RFC8641][RFC8639] extends NETCONF and enables subscriber applications to request a continuous, customized stream of updates from a YANG datastore. Providing such visibility into changes made upon YANG configuration and operational objects enables new capabilities based on the remote mirroring of configuration and

operational state. Moreover, distributed data collection mechanism [I-D.zhou-netconf-multi-stream-originators] via UDP based publication channel [I-D.ietf-netconf-udp-pub-channel] provides enhanced efficiency for the NETCONF based telemetry.

A.1.2. gRPC Network Management Interface

gRPC Network Management Interface (gNMI) [I-D.openconfig-rtgwg-gnmi-spec] is a network management protocol based on the gRPC [I-D.kumar-rtgwg-grpc-protocol] RPC (Remote Procedure Call) framework. With a single gRPC service definition, both configuration and telemetry can be covered. gRPC is an HTTP/2 [RFC7540] based open source micro service communication framework. It provides a number of capabilities which are well-suited for network telemetry, including:

- Full-duplex streaming transport model combined with a binary encoding mechanism provided further improved telemetry efficiency.

- gRPC provides higher-level features consistency across platforms that common HTTP/2 libraries typically do not. This characteristic is especially valuable for the fact that telemetry data collectors normally reside on a large variety of platforms.

- The built-in load-balancing and failover mechanism.

A.2. Control Plane Telemetry

A.2.1. BGP Monitoring Protocol

BGP Monitoring Protocol (BMP) [RFC7854] is used to monitor BGP sessions and intended to provide a convenient interface for obtaining route views.

The BGP routing information is collected from the monitored device(s) to the BMP monitoring station by setting up the BMP TCP session. The BGP peers are monitored by the BMP Peer Up and Peer Down Notifications. The BGP routes (including Adjacency_RIB_In [RFC7854], Adjacency_RIB_out [I-D.ietf-grow-bmp-adj-rib-out], and Local_Rib [I-D.ietf-grow-bmp-local-rib] are encapsulated in the BMP Route Monitoring Message and the BMP Route Mirroring Message, in the form of both initial table dump and real-time route update. In addition, BGP statistics are reported through the BMP Stats Report Message, which could be either timer triggered or event-driven. More BMP extensions can be explored to enrich the applications of BGP monitoring.

A.3. Data Plane Telemetry

A.3.1. The Alternate Marking technology

The Alternate Marking method is efficient to perform packet loss, delay, and jitter measurements both in an IP and Overlay Networks, as presented in [RFC8321] and [I-D.fioccola-ippm-multipoint-alt-mark].

This technique can be applied to point-to-point and multipoint-to-multipoint flows. Alternate Marking creates batches of packets by alternating the value of 1 bit (or a label) of the packet header. These batches of packets are unambiguously recognized over the network and the comparison of packet counters for each batch allows the packet loss calculation. The same idea can be applied to delay measurement by selecting ad hoc packets with a marking bit dedicated for delay measurements.

Alternate Marking method needs two counters each marking period for each flow under monitor. For instance, by considering n measurement points and m monitored flows, the order of magnitude of the packet counters for each time interval is n*m*2 (1 per color).

Since networks offer rich sets of network performance measurement data (e.g packet counters), traditional approaches run into limitations. One reason is the fact that the bottleneck is the generation and export of the data and the amount of data that can be reasonably collected from the network. In addition, management tasks related to determining and configuring which data to generate lead to significant deployment challenges.

Multipoint Alternate Marking approach, described in [I-D.fioccola-ippm-multipoint-alt-mark], aims to resolve this issue and makes the performance monitoring more flexible in case a detailed analysis is not needed.

An application orchestrates network performance measurements tasks across the network to allow an optimized monitoring and it can calibrate how deep can be obtained monitoring data from the network by configuring measurement points roughly or meticulously.

Using Alternate Marking, it is possible to monitor a Multipoint Network without examining in depth by using the Network Clustering (subnetworks that are portions of the entire network that preserve the same property of the entire network, called clusters). So in case there is packet loss or the delay is too high the filtering criteria could be specified more in order to perform a detailed analysis by using a different combination of clusters up to a per-flow measurement as described in IPFPM [RFC8321].

In summary, an application can configure end-to-end network monitoring. If the network does not experiment issues, this approximate monitoring is good enough and is very cheap in terms of network resources. However, in case of problems, the application becomes aware of the issues from this approximate monitoring and, in order to localize the portion of the network that has issues, configures the measurement points more exhaustively. So a new detailed monitoring is performed. After the detection and resolution of the problem the initial approximate monitoring can be used again.

A.3.2. Dynamic Network Probe

Hardware-based Dynamic Network Probe (DNP) [I-D.song-opsawg-dnp4iq] provides a programmable means to customize the data that an application collects from the data plane. A direct benefit of DNP is the reduction of the exported data. A full DNP solution covers several components including data source, data subscription, and data generation. The data subscription needs to define the complex data which can be composed and derived from the raw data sources. The data generation takes advantage of the moderate in-network computing to produce the desired data.

While DNP can introduce unforeseeable flexibility to the data plane telemetry, it also faces some challenges. It requires a flexible data plane that can be dynamically reprogrammed at run-time. The programming API is yet to be defined.

A.3.3. IP Flow Information Export (IPFIX) protocol

Traffic on a network can be seen as a set of flows passing through network elements. IP Flow Information Export (IPFIX) [RFC7011] provides a means of transmitting traffic flow information for administrative or other purposes. A typical IPFIX enabled system includes a pool of Metering Processes collects data packets at one or more Observation Points, optionally filters them and aggregates information about these packets. An Exporter then gathers each of the Observation Points together into an Observation Domain and sends this information via the IPFIX protocol to a Collector.

A.3.4. In-Situ OAM

Traditional passive and active monitoring and measurement techniques are either inaccurate or resource-consuming. It is preferable to directly acquire data associated with a flow's packets when the packets pass through a network. In-situ OAM (iOAM) [I-D.ietf-ippm-ioam-data], a data generation technique, embeds a new instruction header to user packets and the instruction directs the network nodes to add the requested data to the packets. Thus, at the

path end, the packet's experience gained on the entire forwarding path can be collected. Such firsthand data is invaluable to many network OAM applications.

However, iOAM also faces some challenges. The issues on performance impact, security, scalability and overhead limits, encapsulation difficulties in some protocols, and cross-domain deployment need to be addressed.

A.3.5. Postcard Based Telemetry

PBT [I-D.song-ippm-postcard-based-telemetry] is an alternative to IOAM. PBT directly exports data at each node through an independent packet. PBT solves several issues of IOAM. It can also help to identify packet drop location in case a packet is dropped on its forwarding path.

A.4. External Data and Event Telemetry

A.4.1. Sources of External Events

To ensure that the information provided by external event detectors and used by the network management solutions is meaningful for the management purposes, the network telemetry framework must ensure that such detectors (sources) are easily connected to the management solutions (sinks). This requires the specification of a simple taxonomy of detectors and match it to the connectors and/or interfaces required to connect them.

Once detectors are classified in such taxonomy, their definitions are enlarged with the qualities and other aspects used to handle them and represented in the ontology and information model (e.g. YANG). Therefore, differentiating several types of detectors as potential sources of external events is essential for the integrity of the management framework. We thus differentiate the following source types of external events:

- Smart objects and sensors. With the consolidation of the Internet of Things~(IoT) any network system will have many smart objects attached to its physical surroundings and logical operation environments. Most of these objects will be essentially based on sensors of many kinds (e.g. temperature, humidity, presence) and the information they provide can be very useful for the management of the network, even when they are not specifically deployed for such purpose. Elements of this source type will usually provide a specific protocol for interaction, especially one of those protocols related to IoT, such as the Constrained Application

Protocol (CoAP). It will be used by the telemetry framework to interact with the relevant objects.

- Online news reporters. Several online news services have the ability to provide enormous quantity of information about different events occurring in the world. Some of those events can impact on the network system managed by a specific framework and, therefore, it will be interested on getting such information. For instance, diverse security reports, such as the Common Vulnerabilities and Exposures (CVE), can be issued by the corresponding authority and used by the management solution to update the managed system if needed. Instead of a specific protocol and data format, the sources of this kind of information usually follow a relaxed but structured format. This format will be part of both the ontology and information model of the telemetry framework.

- Global event analyzers. The advance of Big Data analyzers provides a huge amount of information and, more interestingly, the identification of events detected by analyzing many data streams from different origins. In contrast with the other types of sources, which are focused in specific events, the detectors of this source type will detect very generic events. For example, a sports event takes place and some unexpected movement makes it highly interesting and many people connects to sites that are covering such event. The systems supporting the services that cover the event can be affected by such situation so their management solutions should be aware of it. In contrast with the other source types, a new information model, format, and reporting protocol is required to integrate the detectors of this type with the management solution.

Additional types of detector types can be added to the system but they will be generally the result of composing the properties offered by these main classes. In any case, future revisions of the network telemetry framework will include the required types that cover new circumstances and that cannot be obtained by composition.

A.4.2. Connectors and Interfaces

For allowing external event detectors to be properly integrated with other management solutions, both elements must expose interfaces and protocols that are subject to their particular objective. Since external event detectors will be focused on providing their information to their main consumers, which generally will not be limited to the network management solutions, the framework must include the definition of the required connectors for ensuring the

interconnection between detectors (sources) and their consumers within the management systems (sinks) are effective.

In some situations, the interconnection between the external event detectors and the management system is via the management plane. For those situations there will be a special connector that provides the typical interfaces found in most other elements connected to the management plane. For instance, the interfaces will accomplish with a specific information model (YANG) and specific telemetry protocol, such as NETCONF, SNMP, or gRPC.

Authors' Addresses

Haoyu Song
Futurewei
2330 Central Expressway
Santa Clara
USA

Email: hsong@futurewei.com

Fengwei Qin
China Mobile
No. 32 Xuanwumenxi Ave., Xicheng District
Beijing, 100032
P.R. China

Email: qinfengwei@chinamobile.com

Pedro Martinez-Julia
NICT
4-2-1, Nukui-Kitamachi
Koganei, Tokyo 184-8795
Japan

Email: pedro@nict.go.jp

Laurent Ciavaglia
Nokia
Villarceaux 91460
France

Email: laurent.ciavaglia@nokia.com

Aijun Wang
China Telecom
Beiqijia Town, Changping District
Beijing, 102209
P.R. China

Email: wangaj.bri@chinatelecom.cn