The Eastasouth Journal of Information System and Computer Science
Vol. 1, No. 01, August, pp. 266 - 278

Data-Driven Network Operations: Framework for Telemetry, Analytics, and Automated Remediation

Manevannan Ramasamy
Software Engineering Senior Technical Leader, Milpitas, California, United States of America

Article Info

Article history:

Received Aug, 2023
Revised Aug, 2023
Accepted Aug, 2023

Keywords:

Automated Remediation;
Data Architecture;
Data Lineage;
Data Quality;
Network Analytics;
Network Operations;
Telemetry

ABSTRACT

The transition to data-driven decision-making, particularly when it comes to contemporary network operations, can not be achieved without an efficient form of automation, yet to approach it, it is required that more than high-volume data collection by telemetry is involved. This paper outlines a design of reliable network operations that includes telemetry ingestion, data quality management, temporal modeling and topology awareness, analytics and automated remediation as a part of a controlled data lifecycle. The architecture suggests a canonical event envelope to offer consistent identity, event time, semantics and provenance to heterogeneous sources. Topology model and temporal entity assist in giving operational context and purpose-specific data store and controlled transformation pipeline are useful in assisting assurance, capacity planning, anomaly detection and remediation processes. A feature registry makes reusable analytical representations and the evidence provenance keeps track of the provenance between the source observations, transformations, analytical outputs and operational actions. The framework specifically addresses the problem of incomplete, late, repetitive, and semantically incoherent information by incorporating measures of quality and decision processes that are conscious of the confidence. Framework assessment is a framework appraisal that gauges completeness, freshness, semantic consistency, lineage coverage, reproducibility, and downstream quality of choice. Based on the proposed architecture, credible automated remediation is achievable, as indicated by the capabilities to reuse the evidence of all decisions to improve the transparency, operational soundness and regulated reuse of network data throughout the transforming analytic and automation processes.

This is an open access article under the CC BY-SA license.

Corresponding Author:

Name: Manevannan Ramasamy
Institution: Software Engineering Senior Technical Leader, Milpitas, California, United States of America
Email: manevannan3@gmail.com

1. INTRODUCTION

Operational ability to convert heterogeneous operational observations into reliable information to interpolate, assure, plan and execute automated action is a growing aspect of modern network operations. The network monitoring platforms continuous addition of router, switches, wireless network, firewall, controller, identification, addressing system, application, cloud monitoring platforms, and synthetic monitoring systems. The different identifiers, timing conventions,

measurement, and sampling intervals, scheme and retention policy used in these sources to create telemetry. Whereas with visualization layers, most of these differences can be masked, this is not true of analytical and automated workflows. An insignificant data anomaly on a dashboard can prove to be a significant contributor to error upon consumption by a data anomaly detector, capacity predictor, dependency analyzer or remediation code [1], [2].

This problem is particularly pernicious when operational data are exposed to the different processes of analysis. Indicatively, two telemetry records can have the same physical interface with diverse naming convention, thereby not allowing proper correlation. A similar example is that a reset of an interface counter may be considered as an abrupt reduction in traffic where the analytical pipeline is not able to distinguish between changes in the state of the counter and a real change of behavior. Clock drift could lead to false sequence of events and late telemetry could be placing an otherwise healthy resource state in a past state of operation. Multiples of records may exaggerate measurements and when a field meaning changes it may invalidly compare over a long history. This kind of inconsistency may spread in transformation pipelines, and ultimately affect the decisions made in operations.

Reliable network data therefore requires more than the telemetry taking to be a systematic framework. RFC 9232 offers an operational template of network telemetry gathering and consumption and outlines an important foundation to understand telemetry creation, transportation and utilization [1] The following data-management layer of the article in question talks about the manner in which the operational observations can be modeled, and validated, contextualized, transformed by analysing them after the acquisition. This level of difference is important because even a telemetry pipeline that is initially designed with a high throughput does not necessarily offer the specified properties to produce reproducible analytics or even (stable) automation.

The development of homogeneous data contracts is one of the principles. The canonical event representation can provide common fields to refer to the source, the entity, the event time, the observation time, the semantics of measurement, indicators of quality and the provenance. Heterogeneous observations can be compared and correlated based on such normalization without the need to discard source-specific characteristics. The platform can also distinguish between current state and the past state and with topology models, measurements can be connected to interfaces, devices, services, applications and dependencies. All these models bring about the contextual backing needed in meaningful operational analytics.

A change to data quality being a stated architectural property, rather than an inferred assumption should also occur. None of these (missing, late, duplicated, out of order or semantically changed observations) can simply get through analytical workflows, and claim to represent high-quality observations. Quality metadata should attend to operational data, and should have an influence on downstream confidence. This can be used to allow analytical systems to distinguish between an actual operational abnormality and an anomaly that is caused by the incomplete or unreliable observations. It is also applicable in the controlled decision-making in situations where the automated remediation is based on evidence, depending on varying levels of confidence.

Provenance and reproducibility is another requirement. Analytic models, historical windows, transformations derived, topological relations, transformations based on the derivations are becoming prominent as grounds of network decisions. Without lineage, it can be difficult to know what observations were applied to come to a given conclusion and be able to replicate the same conclusion at some other later stage. A provenance of evidence therefore results in historic affinity between the source telemetry, transformation stages, derived

representations, analytical discoveries, and the end operational activity. A feature registry can also promote feature reuse of proven analytical features across assurance, forecasting and remediation processes.

The current article in turn suggests a data-driven network operations architecture with a canonical event contracts, time and topology-sensitive modeling, pipelines of transformation governed, purpose-specific data stores, analytical features that can be reused, and provenance of evidence. The architecture confronts the attention to data quality and reproducibility to several operational customers, and not just the speed of collection or the response time. It is evaluated based on completeness, freshness, semantic consistency, coverage of lineages, reproducibility, costs, and quality of downstream decisions. The unspoken rule is that automated remediation is more of a data-controlled process: it is plausible that more when the platform it happens on permits to reconstruct the evidence, transformations, context and confidence behind such an action.

2. RELATED WORK

Studies of data-driven network operations have over the years gradually become less about the conventional network monitoring and more about telemetry designs in which programmability, contextual data collection, structured state modeling and automatic operational intelligence are of interest. An example of standards based on which this evolution can be established is the Network Telemetry Framework according to which telemetry can be defined as a network-wide data collection and consumption system involving a variety of devices and operating under varying conditions [1]. The framework underlines that it is necessary to have the ability to support various sources of telemetry, transport, encoding models, and models of consumption. It is however not just the process of the acquisition of telemetry that is the only phase of a lifecycle of a data-driven operation. Once collected, operational observations must be normalized, correlated, contextualised and governed to be relied upon to deliver stable foundation to analytics and automated remediation.

Data provenance offers a good premise of relying on such operational data. The PROV-O ontology is a conceptual model of explaining the entities, activities and agents involved in information creation and transformation [2]. The concepts that it has developed, although originally applicable as a general provenance model can be applied to network operations quite easily since the telemetry observation may be put through a sequence of ingestion, transformation, aggregation and analytical processes. Provenance relationships can in turn assist in determining the origin of an operational observation, how it has been altered and which processing activities have led to a result being derived. This applies best in a circumstance where automatic actions are supposed to be done but then others must follow it to show evidence that led to the actions being taken.

More recent lineage-based methods generalize this to data-processing pipelines. The OpenLineage is a specification to capture lineage about datasets, job and processing events [3]. It is this lineage that can be employed to give an analysis reproducibility through connecting the outcomes of an analysis with previous data products and processing steps. This is a useful perspective to network operations since in most cases, telemetry is not directly consumed. The measurements may be filtered, resampled, augmented with topology information, aggregated into time windows, converted into features and then presented to anomaly detection or remediation systems. A lineage-conscious architecture is able to maintain such relationships and make it harder to deconstruct the decision on how an operation was made.

The other notable architectural layer is the network management interfaces. The gNMI specification is a homogenous interface to read and write structured network configuration and state data [4]. This helps to bring about regular

communication between the infrastructure components and network management systems. These interfaces may be useful as sources of operational state in a data driven architecture, and as sources of a mechanism wherein approved decisions can be specified in terms of configuration changes. But management interfaces so are not a full data governance model. They allow access to state and configuration, but other mechanisms are needed to be acceptable of temporal consistency, provenance, semantic normalization, and analytical reproducibility.

The Network Management Datastore Architecture extends to accommodate this point of view, by defining each conceptual separation between intended configuration, applied configuration and operational state [5]. This distinction has significant implications to analytics because at some points, the network decisions are based upon what is distinct between the way a system is or ought to behave. An operational data architecture can reflect an expression of a controlled behavior and intent by combining ideas of datastores with telemetry and lineage mechanisms. Such integration has the potential of improving contextual analysis, providing a superior foundation on how to determine deviation, the core root cause of this deviation, and what will be the proper course of action that should be undertaken to correct the deviation.

Programmable telemetry has been greatly studied in the measurement efficiency and flexibility. IntOpt offers an optimization model of network slice monitoring with in-band telemetry and P4 whereby telemetry specifications are coded as monitoring flows and during optimization, monitoring overhead and latency are minimized [6]. It is shown in the work that even the process of telemetry collection may be considered an optimization problem instead of a data-generation process that is not restricted. This is significantly associated with the data-driven network structures in which the speed of telemetry, scope of activities and overheads are exchanged with the analysis demands.

Similarly, adaptive monitoring has been studied in the context of software-defined networks. Self-adaptive decayralized monitoring strategy is a strategy that is dynamic whereby the monitoring activities vary with the dynamic nature of the network [7]. This direction is important in the sense that the outcome of a managerial policies on static collections is that they make one a sustained stream of unnecessary data at the time of things not changing and they make insufficient information at the time of things changing. Adaptive monitoring can therefore provide a conceptual framework that can be used to bridge data collection policy and both operational context, and downstream analytical needs.

Network-wide analytics processes have been more centrally processed to the source of the telemetry due to the use of programmable data planes as well. One example of the application of programmable switches in extending the measurements and even the analysis of the network is a P4-enabled architecture of heavy-hitter detection [8]. This kind of solution reduces the flow of all observed raw data to centralized systems and proves the benefit of implementing selected steps of the telemetry-to-analytics pipeline. This idea is extended by the selective in-band telemetry where telemetry information can be selectively processed depending on the operational demands in runtime-programmable fashion [9]. These tactics work together to bring out the importance of controlling the volume of telemetry, its relevance and the fineness of its gathering.

Earlier studies on adaptive monitoring had determined that the accuracy of monitoring and the resources used have to be balanced. Adaptive continuous network monitoring was suggested by A-GAP that has a clear accuracy objective [10]. This offers a good basis to architectures where telemetry observation is linked to quality or confidence data as opposed to it being considered equally valid. This perception is more vital when there is a missing, delayed, duplicated or inconsistent observations.

The expansion of network monitoring to expanded surveys defines the transition in the traditional monitoring tools to more enriched and flexible tools, which can absorb the new programmable and virtualized structures [11]. In the same way, the architecture building blocks of a centralized control, network management abstraction, and programmability have been provided by software-defined networking and network virtualization studies [12]. The developments made it affordable to detach measurement and control functions and analysis functions in other telemetry systems with a more elevated overview of the virtualized network layout.

Scalable traffic metrics have been utilized in the development of data-driven network analytics, as well. The OpenSketch project showed the potential of programmable measurement structure to effectively provide large scale traffic analysis, as well as resource usage management [13]. Sonata extended query-driven telemetry in the sense that it made possible complex queries of analysis to guide the collection and processing of measurements of the network [14]. These plans implement and publicize the reality that telemetry has to be functionally and analytically influenced, not to be piled up indiscriminately.

Despite these advancements, the present task distribution is across the telemetry acquisition, programmable measurement, network management, adaptive monitoring and provenance spheres. Somewise a relatively narrow emphasis has been on internalizing these issues into a single operation data architecture that also meets canonical data contracts, time and topology conscious modeling, data quality, provenance, analytical-reproducibility and automated data remediation. This is one of the gaps which are covered in the proposed architecture, which considers network telemetry more as a controlled data lifecycle, than as a measurement mechanism. It also maps heterogeneous observations to standardized representations and contextual models, quality indicators, analytics features, provenance relations and decision evidence, which ought to form the foundation of more transparent as well as reproducible data-driven network functions.

3. EVIDENCE-CENTRIC OPERATIONAL DATA ARCHITECTURE

The operational data within the network is handled in the proposed architecture as a managed piece of evidence lifecycle as opposed to a set of autonomous telemetry streams. Its primary objective is to construct a stable relationship between heterogeneous observations, the context of its operation, transformatory operations of analysis, conclusions drawn and subsequent remediation interventions. The architecture has six layers of inter-relations: telemetry acquisition and canonicalization, time-based entity and topology model, controlled storage, analysis processing, features and evidence model and quality conscious decision enforcement. This cluster separates data collection and interpretation and ensures the connections that are required in order to implement traceability and reproducibility.

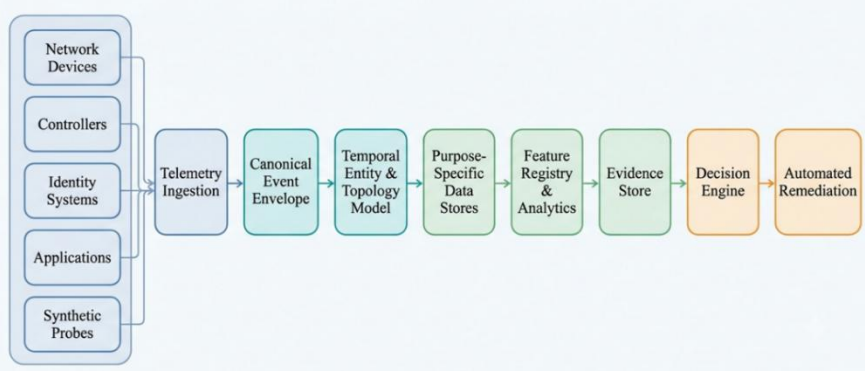


Figure 1. Evidence-Centric Network Operations Architecture

3.1 *Telemetry Acquisition and Canonical Event Envelope*

APNs on the acquisition layer are fed by network devices, controllers, management systems, identity services, addressing infrastructure, applications, synthetic probes and external operational tools. Sources differ in terms of identifiers, timestamps, sampling intervals, units, among others, schemas and collection mechanisms. The architecture introduces a canonical event envelope to payloads generated by sources, rather than converting all producers to the same payload format.

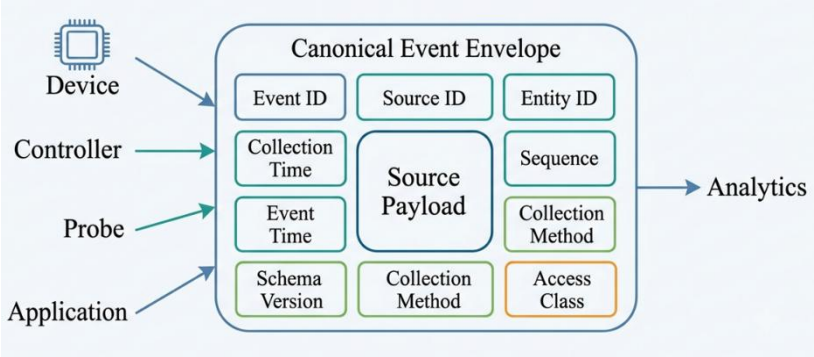


Figure 2. Canonical Event Envelope and Data Contract

Both common envelope of each observation consists of event identity, source identity, managed-entity reference, event time, collection time and in sequences, sequence information, schema identity and version, collection method, quality flags it has and access classification. The payload stores source-specific information, and a collection of selected mapped fields- are used to provide the cross-source analytics with a standard format. The methodology of the Network Telemetry Framework extends the concepts of downstream governance which have been furthered through the Network Telemetry Framework [1], and better grounds on concepts of collection and consumption of telemetry.

Limitations are also spelled out by the envelope. When compared with event time and collection time, the collection delay can be determined by the fact that gaps between sequences can result in loss of data. There is also a counter-reset signal to prevent the occurrence of a negative change in traffic being considered a reset. The quality flags are applied to indicate the circumstances such as interpolation, sampling, deduced identity, incomplete payloads or various limitations. Analytical consumers can hence specify what they will accept in terms of observations and the state of quality rather than assuming that all records were of equal quality.

The development of the schemas is controlled by clear compatibility rules. This is true despite the fact that additive fields will usually be compatible with the existing semantics, even though modified units, modified semantics or a structural structural incompatible change will result in a new version with a new schema. Manufacturers publish machine-readable contracts and sampling and consumers check these contracts as continuous integration is underway. The outcome of each analysis lists versions of the schema which were used to create it so that a person using the data in future can reproduce the version of the data interpretation which was used when a decision was made.

3.2 *Temporal Entity and Topology Model*

Network identity is dynamic, one of the most fundamental architectural issues. Physical device replacement can be done without losing a management address, switching

between access point can be done by a client and an interface identifier can be reused after replacement of hardware. The architecture thus separates identities of stable platforms and identities that are source specific and maintains time constrained identities between them.

Entity service is the responsibility of identity resolution, mapping rules, validity periods and identity confidence. The architecture is ambivalent and it demonstrates its level of confidence and by no means attempts to force ambiguous observations into a single fact as it is implicitly attempted with the rest of the explicit knowledge content. This is particularly important in a scenario that entails high-impact analytical decision making, whereby a minimum degree of identity-confidence can be indicated past which they are not supposed to permit automated action.

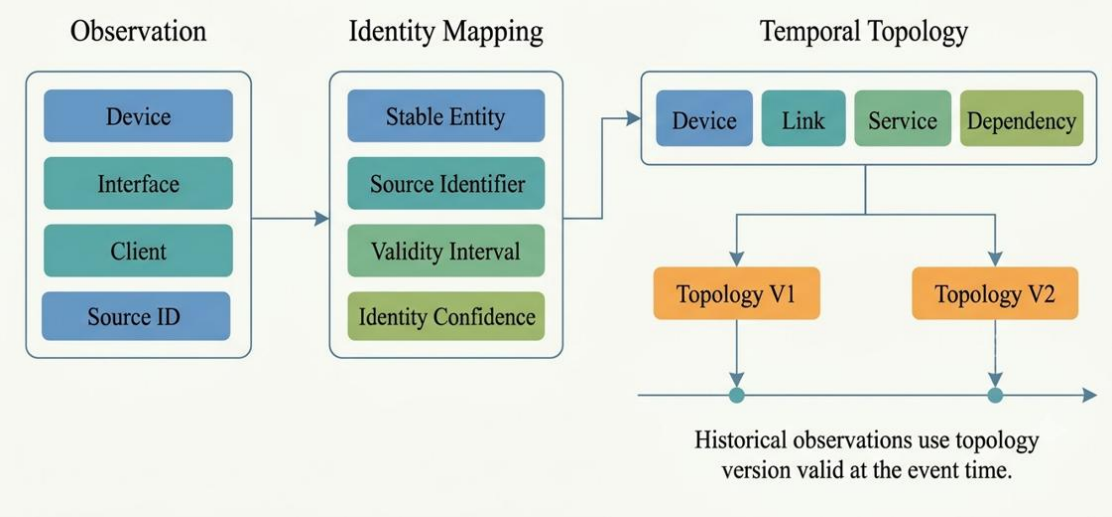


Figure 3. Temporal Entity and Topology Model

Topology is defined as a time based graph instead of a network diagram. Physical connections evolve, logical adjacencies evolve, forwarding paths evolve, site membership, policy relationship evolve, service dependency existence evolve, workload relationships evolve. There is an efficient range of validity and version thus associated with a given topology state. Analytical queries utilize the topology version that was valid at the time an observation was made as opposed to automatically taking the current network topology.

This methodology based on time does not permit re-writing the past using the assistance of later network transformations. It also provides more context of causality of analytics in that an observation can come across not just the observed but the relationships that are present in the network at that instant of observation.

3.3 Polyglot Storage and Governed Processing

Operation data is typified by varying access and retention requirements and hence cannot be stored using a solitary storage technology. Architecture, thus, makes use of purpose specific storage. A streaming log records the order of insertion and can be restarted, time-series storage supports metric windows and baselines, object storage supports cost-effective storage of raw evidence, graph storage supports topology and dependency travel and search indexes support exploration (exploratory queries) and diagnostics queries to logs.

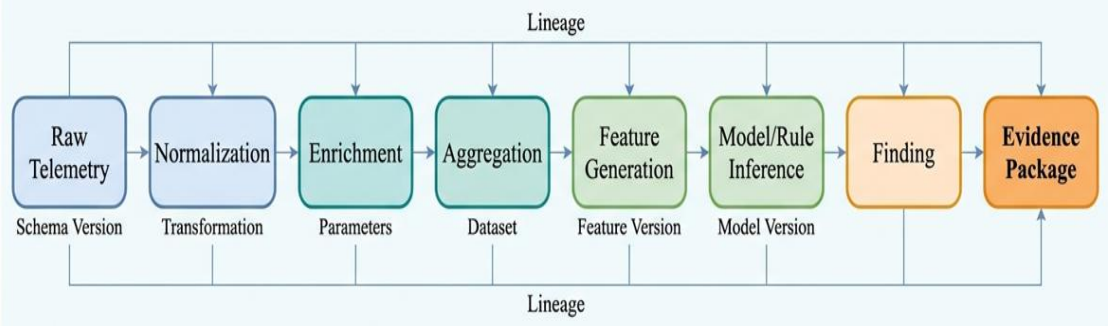


Figure 4. Governed Data-to-Evidence Lifecycle

In transaction management systems, configuration and operation intent is stored, rather than being a typical telemetry. Network Management Datastore Architecture establishes management-state concepts that provide a helpful foundation to determine the intended state, applied state, and operational state [5].

These storage places are connected with data transformations. Each processing pipeline will capture the input datasets, code of transformation or version of the rule, this specific set of parameters, its execution identity, the beginning and termination of execution and resulting dataset. This generates a machine readable chain between raw observations and analytical products.

The provenance model is conceptually similar to W3C PROV that makes relationships among entities, activities, and agents in the information creation and transformation process [2]. OpenLineage provides the conference of lineage on jobs, runs and datasets [3]. These processes are scaled to network analytics to enable a finding to be followed through ingestion, normalization, enrichment, aggregation, feature generation, and model or rule inference.

Retention is purpose-specific. Raw telemetry at high frequency may have a lower retention time and incident evidence, configuration changes, intent records and aggregate service indicators may have a higher retention time. The delete policies must be spread out across the derived datasets, indexes and across the analytical products. Information related to identity and location is granted access of appropriate restricted access and possibly tokenized when the identity is not needed to be explored in greater detail.

3.4 Feature and Evidence Governance

The analytical layer adds a feature registry that offers a controlled definition to each reuseable analytical input. These characteristics include their name, their owner, the fields their names are founded on, their transformation logic, their temporal window, their units, validity criteria, their expected missing data, the scope of the population to which they are applied and its version. This definition is equally widespread, and both training and online inference can work on the identical constituents of feature semantics and eliminates the risk of training-serving skew.

The results of the analysis are displayed in a form of evidence packages in which they are unable to be modified but introduced as temporary warnings. Each package is characterized by the goal of the operation that is affected, the timeline of observation, the topology version, the features version, the models and rule versions, the confidence level, supportive signals, incompatible signals and decision rendered. In case of a correction made in a finding, a new version is made; a connection is made to the first version instead of replacing the historical record.

The mechanism preserves the uncertainty and information on starting point of decision. It also facilitates analysis of impacts. Lineage can be used to find the impacted features, findings, and automated actions when a source data or transformation turns out to be unreliable. Provenance thus can be seen as a work control mechanism but not an archival facility.

3.5 Data Quality and Confidence Enforcement

Data quality standards exist, which are dependent on the purpose of the activity. Completeness will give the relevant scope of the expectancy against the observed observations. Freshness; This is a scale who measures age of the information at the time of making a decision. The measurement can be verified to validated measurements to establish the accuracy or by cross-source consistency. Semantic checks the units, identifiers and schema, and reliability checks the behavior of the source or pipeline.

These dimensions are correlated with familiar data-quality views that examine such characteristics as the accuracy, completeness, update status, relevance, consistency, reliability, presentation, and accessibility [4]. The architecture translates these attributes into the explicit quality indicators of data products and results of similar analysis.

Each consumer announces quality contract specifying minimum conditions that it would have to use. A historical capacity report is resilient to late or incomplete observations, but requires that recent telemetry and valid topology, high identity confidence, and continuous observation exist. Whether or not there is a quality contract breach, the platform can reduce the analytics confidence, look further, submit the decision to another opportunity, or act automatically.

4. ANALYTICS, DECISIONING, AND REMEDIATION INTEGRATION

The highest tier bridged evidence to operating analytics and restrained automation. Anomaly detection, capacity analysis, service assurance, root-cause analysis, and predictive models utilise canonical observations with context of the temporal entity and topology. Programmable telemetry techniques used demonstrate that collection can itself be optimized with respect to the objectives of monitoring and available resources [6], and adaptive monitoring experiments demonstrate how the observation strategies can be optimally adjusted to available resources [7].

The programmable design of the data-plane also demonstrates how functions of interest in analytics might be applied at the source of telemetry [8], and selective in-band telemetry touches on why careful choice of values flowing downstream processing is critical [9]. The relevance of the accuracy of monitoring to the collection overhead is also determined using previous adaptive monitoring [10]. Purpose-driven collection and quality-conscious consumption is a way of implementing these principles into the proposed architecture.

Finally, its evidence package and policy version used are mentioned on all automated remediation actions. The action is related then to the observations, topological state, feature definition, to the analytic reasoning, and to quality criteria underlying it. This forms a closed operation chain:

Observation → Context → Quality → Analysis→ Evidence => Decision+Action => Outcome.

The resulting architecture uses techniques in existing telemetry and network-management [11], and is augmented with abstraction and programmability of the software-defined and virtualized networking [12]. The concept of scalable measuring [13] and query-driven telemetry [14] also give evidence to the notion that data collection

should be strongly related to the analysis objectives. The architectural input that has been done is the integration of these attributes into a controlled life cycle of evidence wherein both the data quality, time provenance, provenance and traceability of decisions are enrolled as first-class members of automated functions of networks.

5. EVIDENCE-ORIENTED FRAMEWORK ASSESSMENT

Evaluation of the presented architecture suggests the use of formalized validation model that is intended to examine how the notion of the canonicalization, temporal context, data-quality management and provenance can improve the confidence levels of data-driven activities within the network. The situation is analysed considering the quality of operational data life cycle and its effects on downstream remedial and analytical decision. The framework rather than measuring telemetry collection according to volume or throughput measures the evidence of the operations in terms of whether it is an accurate, interpretable, traceable and reproducible as it traverses the architecture.

4.1 Data Integrity and Resilience Assessment

Controlled data-quality tests add to telemetry streams: representative imperfection of the data; late events; event duplication; event not produced; incomplete event records; and out of order event records. These conditions are typical to distributed network setups wherein the collection paths, buffering logic, source availability and schema versions can be different. The evaluation scale is employed to measure how well the architecture tracks, recognizes, and keeps such constraints without actively reforming invalid observations to appear sound data.

Identity-resolution tests bring about device replacement, reuse of management address, client mobility, interface reuse, and unidentifiable web of origin. The attained measurements determine the accuracy of the entity-mappings, the certainty allocated, and prevention of ambiguity and ease of maintaining identity relations over the course of history. Topology tests alter links, paths and dependencies, as well as relationships between sites around when an operational incident occurs. These tests identify that analytical processing is making use of the topology version in relation to the period of observation and not the wrong use of a subsequent network state.

Event-loss-detection, temporal-freshness, Identity-Mapping performance, semantic-validation, quality-flag-accuracy, lineage-completeness, replay-reproducibility and storage-overhead are the key measures of evaluation. All these are taken to gauge the capacity of the architecture to sustain operational sense in addressing the heterogeneous and imperfect data.

4.2 Analytical and Decision-Level Assessment

The second step is to recapitulate whether stronger data-contracting, and governance can bring about measurable enhancements to the downstream network knowledge. Two pipelines controlled and uncontrolled (uncontrolled is played without proposed governance mechanisms) are analyzed analytically (based on analytical model and decision logic), with an uncontrolled pipeline (where observations play without the proposed governance mechanisms) and the governed pipeline (including classical event envelope, time modeling, quality contract and provenance tracking). The preservation of the identical model and analytical goal separates the impact of data governance with an alteration in the capability of the algorithm.

Comparison deals with inaccurate outcomes, inaccurate outcomes, force of induction, consistency of diagnosis and automatic action behavior. Special attention is paid

to the cases when incomplete, stale, ambiguous and semantically inconsistent data may have to cause distrust or not to allow automated correction. It is not just the detection of an anomaly that is considered in the assessment, but also the suitability of the evidence used to come to the decision made as per the needs of its operational application.

4.3 Reproducibility and Cost Assessment

Reproducibility is achieved through the selection of historical results and their re-creation using remaining source observations, version of topology, version of feature definition, version of schema and version of logic of transformation. The reproducibility of a result is established by having the result regenerated as an acceptable tolerance of the original result.

Finally, storage and processing overhead occurs on raw, transformed, indexed and derived data. This causes the cost of stronger lineage and evidence governance operation to be operational and allows the reliability advantages of the architecture can be discounted into the resources demands of the architecture.

6. DISCUSSION

Data governance is not only essential to trustworthy network analytics, but governing might overwhelm operations with over-governance. The need to tediously model all the telemetry sources to then use the data might decrease onboarding time, increased maintenance effort, and reduce its use. The proposed architecture therefore considers the governance as a progressive-increase as compared to a common demand in all data sources.

Standard canonical mappings and automated schema discovery, as well as deduced metadata, can first add telemetry with minimal risk to the platform. Such systems avail few human interventions, which are helpful in providing some organization. The reduction of the effort to integrate new sources, additional to the reusable entity models and standard templates of transformation, are also standardized transformation, and quality indicators that are automatically generated. Over time, experienced mappings can be promoted to managed contracts and data products reusable as experience builds up in operations.

Data that has a direct impact on high-impact analytic or automated decisions receive tougher controls. Indicatively, capacity reporting with telemetry and data with automated remediation might have lower tolerance to moderate delays or incomplete observations, but might need higher assurances of freshness, identity assurance, topology soundness, semantic rigidity, and lineage completeness. It is a risk-conscious process that renders the intensity of governance pertinent to the potentiality of incorrect information to misbehave over the operations.

Progressive governance is supported by the architecture as well. A newly attached source can be initially placed in observational status and its schema, quality characteristics and identity mappings are taken into consideration. One reason is that built confidence will lead the source to change towards greater contractual requirements. This facilitates easy onboarding and maintenance of a route with more certainty.

The other factor that needs to be taken into consideration is the expense of preserving the historical lineage and evidence. This may not be practical in high-telemetry rates, as raw observations will have to be stored completely. The purpose-specific retention, aggregation, sampling and evidence preservation can therefore be balanced with traceability and storage requirements. Evidence of critical incidents and decision records must have more extended retention than high-frequency observations that are routine.

Finally, good governance is not pegged on the amount of regulations that are put on data producers. The strength it has is that it can assist in observing operational limitations, prevent erroneous decisions and

can reproduce the outcome of analysis with a high level of certainty. Risk-sensitive governance model, thus, allows flexibility of analytics that are more frequent, and tighter controls in cases where erroneous or poorly comprehended pieces of evidence may give launch to massive times of operation.

7. CONCLUSION

Data-driven network operations go beyond current collection of telemetry data, to require a temporal, versioned and traceable state of those evidences upon which network behavior is to be understood and manipulated. The proposed architecture fulfills this requirement using a canonical event envelope, temporal identity resolution, topology-sensitive modelling, purpose-specific storage, sophisticated transformation pipelines, the definition of features, provenance and consumer specific quality contracts. All of them assembled together will form a single operation data cycle of various networks sources.

The time of events, the identity of the source, schema versions, collection procedures and data constraints are explicit to both the canonical envelope and temporal entity and topology models preserve the context of observation generation at the time it was generated. Storage and transformation lineage specialisation to a selected purpose allows the effective analysis and traceability of raw observations to the final analytical products. The concept of feature governance also enhances uniform analytical semantics and lessens inconsistencies amid data preparation and downstream inference.

An evidence-based auto-remediation strategy is also defined by the architecture. Analytical findings may store the information needed to keep the observations that support it, variants of topology, feature definition, transformation variants and confidence information such that recreation of the decision and rethinking can be performed. When evidence does not pass through preset quality controls, automated processes might start timed out or stopped instead of bearing on unverified information.

Widely, the architecture harmonizes the data quality, provenance, time-based setting and reliability as an elemental component of plausible network automation. This provides scalable analytics and remediation in variable network environment.

REFERENCES

[1] H. Song, F. Qin, P. Martinez-Julia, L. Ciavaglia, and A. Wang, "Network Telemetry Framework," RFC Editor / Internet Engineering Task Force, 2022. doi: 10.17487/RFC9232.

[2] World Wide Web Consortium, "PROV-O: The PROV Ontology." W3C Recommendation, 2013. doi: https://www.w3.org/TR/2013/REC-prov-o-20130430/.

[3] OpenLineage Project, "OpenLineage Specification." OpenLineage, 2021. [Online]. Available: https://github.com/OpenLineage/OpenLineage/blob/main/spec/OpenLineage.md

[4] OpenConfig, "gRPC Network Management Interface (gNMI) Specification." OpenConfig, 2023. [Online]. Available: https://github.com/openconfig/reference/blob/master/rpc/gnmi/gnmi-specification.md

[5] M. Björklund, J. Schoenwaelder, P. Shafer, K. Watsen, and R. Wilton, "Network Management Datastore Architecture (NMDA)," RFC Editor / Internet Engineering Task Force, 2018. doi: 10.17487/RFC8342.

[6] D. Bhamare *et al.*, "IntOpt: In-band Network Telemetry optimization framework to monitor network slices using P4," *Comput. Networks*, vol. 216, 2022, doi: https://doi.org/10.1016/j.comnet.2022.109214.

[7] G. Tangari, D. Tuncer, M. Charalambides, Y. Qi, and G. Pavlou, "Self-Adaptive Decentralized Monitoring in Software-Defined Networks," *IEEE Trans. Netw. Serv. Manag.*, vol. 15, no. 4, pp. 1277–1291, 2018, doi: https://doi.org/10.1109/TNSM.2018.2874813.

[8] D. Ding, M. Savi, G. Antichi, and D. Siracusa, "An Incrementally-Deployable P4-Enabled Architecture for Network-Wide Heavy-Hitter Detection," *IEEE Trans. Netw. Serv. Manag.*, vol. 17, no. 1, pp. 75–88, 2020, doi: https://doi.org/10.1109/TNSM.2020.2968979.

[9] S. Tang, D. Li, B. Niu, J. Peng, and Z. Zhu, "Sel-INT: A Runtime-Programmable Selective In-Band Network Telemetry System," *IEEE Trans. Netw. Serv. Manag.*, vol. 17, no. 2, pp. 708–721, 2020, doi: https://doi.org/10.1109/TNSM.2019.2953327.

[10] A. G. Prieto and R. Stadler, "A-GAP: An Adaptive Protocol for Continuous Network Monitoring with Accuracy Objectives," *IEEE Trans. Netw. Serv. Manag.*, vol. 4, no. 1, pp. 2–12, 2007, doi: https://doi.org/10.1109/TNSM.2007.030101.

[11] S. Lee, K. Levanti, and H. S. Kim, "Network Monitoring: Present and Future," *Comput. Networks*, vol. 65, pp. 84–98, 2014, doi: https://doi.org/10.1016/j.comnet.2014.03.007.

[12] R. Jain and S. Paul, "Network Virtualization and Software Defined Networking for Cloud Computing: A Survey," *IEEE Commun. Mag.*, vol. 51, no. 11, pp. 24–31, 2013, doi: https://doi.org/10.1109/MCOM.2013.6658648.

[13] M. Yu, L. Jose, and R. Miao, "Software Defined Traffic Measurement with OpenSketch," in *10th USENIX Symposium on Networked Systems Design and Implementation (NSDI 13)*, 2013, pp. 29–42. [Online]. Available: https://www.usenix.org/conference/nsdi13/technical-sessions/presentation/yu

[14] A. Gupta *et al.*, "Sonata: Query-Driven Network Telemetry," arXiv, 2017. [Online]. Available: https://arxiv.org/abs/1705.01049