Network Working Group
Internet-Draft
Intended status: Informational
Expires: 27 October 2026

J. Dong, Ed.
Huawei Technologies
M. McBride, Ed.
Futurewei
F. Clad, Ed.
Cisco Systems
Z. Zhang
HPE
Y. Zhu
China Telecom
X. Xu
R. Zhuang
China Mobile
R. Pang
China Unicom
H. Lu
Y. Liu
Tencent
L. Contreras
Telefonica
M. Durmus
Turkcell
R. Rahman
Equinix

25 April 2026

Fast Network Notifications Problem Statement
draft-ietf-rtgwg-net-notif-ps-01

Abstract

Many network applications, ranging from Artificial Intelligence (AI) /Machine Learning (ML) training or inference to cloud services, require high bandwidth, low delay, low jitter and minimal packet loss in data transfer, which requires that the networks can be adaptive in the presence of faults, degradations, or congestion. However, existing traffic management mechanisms often face limitations in responsiveness, coverage, and operational complexity, particularly in high-speed and large-scale network environments. A good and timely understanding of network operational status can help to enable faster response to critical events, so as to enable the selection of paths with reduced latency and improve network utilization. This document describes the existing problems and the need for fast network notification solutions.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

Many network applications, ranging from AI/ML training or inference to cloud services, require high bandwidth, low delay, low jitter and minimal packet loss in data transfer, which requires that the networks can be adaptive in the presence of faults, degradations, or congestion. To meet these requirements, networks employ mechanisms such as traffic engineering (TE), load balancing, flow control, and protection switching. However, existing solutions often face limitations in responsiveness, coverage, and operational complexity, particularly in high-speed and large-scale environments.

Many network devices are capable of detecting congestion, microbursts, queue buildup and other localized impairments at fine-grained time scales, ranging from microseconds to sub-millisecond, depending on hardware capabilities and deployment requirements. These detection capabilities substantially outpace the time required for such information to be disseminated to other relevant nodes for their actions, creating a gap between what the detecting node can observe and when recipients can react. Fast network notification identifies the need for complementary mechanisms that enable low-latency notification of network conditions, allowing actions taken in the data plane to more closely align with the capabilities of contemporary forwarding hardware. The information delivered by fast network notification may also be used for actions taken in the control plane or management plane.

This document summarizes the limitations of existing mechanisms that prevent them being used for rapid notification of critical network events, including link or node failures and congestion. It also identifies the need for fast network notification which is critical for enabling fast reaction. In the context of this document, fast does not imply a single, rigid numerical time threshold. Instead, it characterizes a class of mechanisms to minimize the notification delivery time so that the latency of the notification is in the order of sub-milliseconds or milliseconds, depending on the operational objective and the range of the network domain, and can be substantially shorter than the Round-Trip-Time (RTT) of the network traffic involved. The scope of this work is limited to fast notification of network conditions. Improvements such as reduced

packet loss or faster mitigation are possible results of the actions consuming such notifications, but are not themselves goals or requirements of the notification mechanism.

[I-D.geng-fantel-fantel-gap-analysis] provides a gap analysis of existing solutions and where they are deficient in supporting high demand services. This document describes the set of problems which the a network notification solution needs to address. The problems described in this document apply across a range of network scenarios and topologies. However, the mechanisms used to provide notifications, and the feasibility of meeting specific timeliness requirements, may differ depending on topology and deployment context. Further discussions of the requirements for a Fast Network Notification system can be found in [I-D.geng-fantel-fantel-requirements]. This document does not assume one-size-fits-all.

2. Glossary

BFD: Bidirectional Forwarding Detection [RFC5880]

ECN: Explicit Congestion Notification [RFC3168]

FRR: Fast Re-Route [RFC4090] [RFC5714]

IOAM: In-situ Operations, Administration, and Maintenance [RFC9197]

3. Why Fast Network Notification is Needed

Current network mechanisms were not designed for the responsiveness and scale required by todays' dynamic environments. Techniques such as load balancing, protection switching, and flow control rely on feedback loops that are often too slow, too coarse, or too resource-intensive. This results in performance bottlenecks, delayed recovery, and inefficiencies in large-scale AI, cloud, and WAN deployments. A fast network notification mechanism could help to address these gaps by providing lightweight, real-time, actionable alerts that complement existing tools and enable faster, more accurate traffic manipulation decisions.

In particular, the detection and propagation of network events (e.g., failure, congestion or state change) must occur within a timeframe short enough to meaningfully influence traffic engineering and load-balancing decisions before congestion or micro-loops occur or develop. In backbone or datacenter networks, this typically implies a target of notification delivery in the order of milliseconds, with some environments requiring sub-millisecond performance. The precise requirement is driven by:

* The speed at which traffic shifts can induce overload.

* The granularity of TE tuning (fine-grained vs. coarse-grained).

* The propagation diameter of the network notification.

* The responsiveness of the control-plane and forwarding-plane components.

* The number of network nodes which generate the notification, and the number of nodes which need to receive the information.

* The volume of information that needs to be reported, and the rate of change of the information.

Therefore, this document focuses on notification mechanisms capable of operating within these millisecond/sub-millisecond ranges, rather than mechanisms whose latency spans tens or hundreds of milliseconds, which are insufficient for preventing transient overload under rapid traffic transitions.

4. The Problem with Existing Mechanisms

Current network traffic manipulation mechanisms such as TE, load balancing, flow control, and protection, have deficiencies in providing the low-latency, high-granularity responsiveness needed in modern, dynamic networks, at least in part due to the lack of dynamic network state information. This results in suboptimal performance, low reliability and delayed recovery. Fast network notification is a set of solutions to address this by enabling real-time, lightweight notifications that enhance the responsiveness for traffic engineering, congestion mitigation, and failure protection. There is a demonstrable need for a standardized framework to define these fast network notification mechanisms, requirements and integration strategies.

There follows a summary of the limitations of existing mechanisms:

* Slow Dissemination: Existing control protocols (e.g., routing protocol, etc.) may be used for dissemination of dynamic network state information, while they usually rely on control plane based hop-by-hop distribution, which causes delay when the recipient is multiple hops away. With modern high-throughput environments (AI/ML clusters, multi-DC WANs), this delay is often prohibitive. Explicit Congestion Notification (ECN) [RFC3168] needs congestion signals to be sent back to the sender, which introduces Round-Trip-Time (RTT) delay and can be slow if the source node is far away, and it relies on the source node to take action in the

transport layer. What is needed is a lightweight signaling method that can provide real-time alerts (e.g., at the sub-milliseconds level or in the order of a few milliseconds) on failures, congestion, or threshold breaches, enabling prompt actions (e.g., in the range of one millisecond to tens of milliseconds) in the network layer.

* Coarse-Grained Signals: Classic ECN [RFC3168] uses a 2-bit field in packet header to convey the ECN capability and congestion indication, which inherently limits the information it can report to the receiving nodes. What would be useful is a set of notifications that aren’t just "on-off" state reports, but can also convey more information like congestion level/utilization information, latency spikes, queue buildup or flow characteristics, so that it can trigger precise responses like rerouting, rate adjustment, or protection switching for specific flows.

* Limited Visibility on Network Conditions: Current load-balancing, flow-control, and FRR techniques are limited by their lack of visibility over downstream or cross-domain network conditions, reducing their effectiveness and leading to suboptimal decisions. For example, the Point of Local Repair (PLR) executing FRR makes its decision based on its local view of the topology and network status. It may switch traffic to a backup path and cause cascading congestion on that path, as it lacks visibility into the state of the entire backup path. Similarly, traditional load-balancing is based on local link utilization information, which may cause some paths overloaded while others remain underutilized. This local view of network status prevents precise and optimized decisions and adjustments. It would be helpful to send fast network notifications to upstream nodes so that they can perform action based on a wider view of network conditions.

* Overhead and Scalability Challenges: The distribution of high-volume network operational status information or frequent signaling introduces bandwidth and processing overhead. At scale, this becomes a bottleneck rather than a solution. IOAM [RFC9197] and similar tools provide detailed telemetry information, but the collection and feedback loops are controller-centric. They cannot be used to deliver lightweight, rapid alerts for immediate action on specific network nodes. Carrying dynamic network state information in control protocols (e.g., routing protocols) also increases the overhead and churn of the control plane, which may have negative impact to the core functionality of the protocol. It would be useful to have solutions designed to avoid the overhead and churn introduced by telemetry flooding or route distribution, so it can adapt to large-scale networks and dynamic traffic patterns (e.g., AI workloads, cloud WAN bursts).

4.1. Example: AI Training Cluster with Fiber Link Failure

Consider a large-scale AI training job distributed across multiple data centers. These clusters exchange terabits of data per second between Graphics Processing Unit (GPU) nodes, requiring ultra-low latency and high throughput to maintain synchronization.

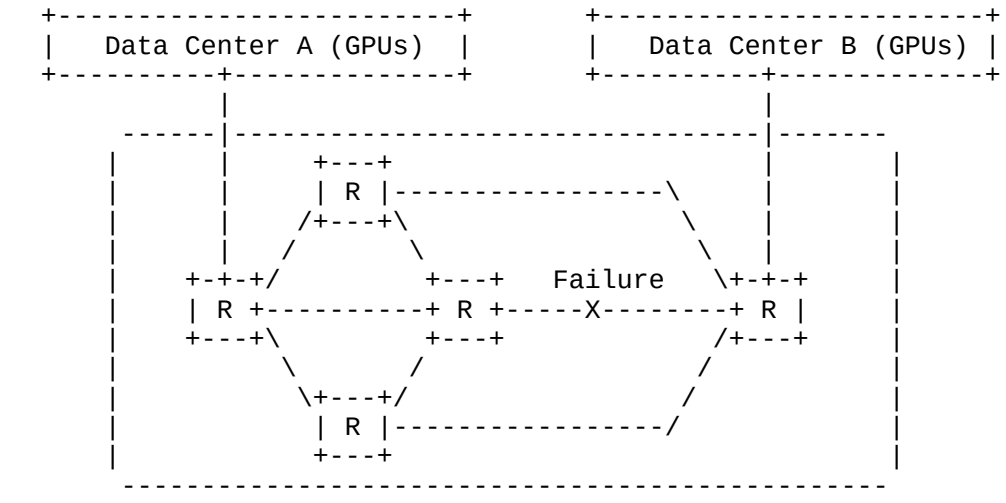

Figure 1: Distributed AI Training Clusters with Fiber Link Failure

As depicted in the above figure, a single fiber link failure event can disrupt the entire training run, leading to:

* Delays in job completion (hours to days for large models)

* Massive energy and compute cost waste due to resynchronization

* Degraded convergence accuracy if synchronization windows are missed

4.1.1. Limitations of Existing Mechanisms

Today’s mechanisms provide partial solutions but are not fast or precise enough for these scenarios:

* BFD [RFC5880]: Provides fast fault detection in the bidirectional path between two forwarding engines. BFD can be one of the detection mechanisms for link or path failures, while it is not used to notify the failure to nodes other than the BFD endpoints in the network. BFD is preconfigured with periodic message exchange, while fast network notifications needs to be event-driven.

* FRR [RFC4090][RFC5714] /Route convergence: Without fast notification, the failure detection can take tens of milliseconds, followed by either local repair (FRR) or route convergence. The former lacks visibility of the global network situation and thus may cause congestion on the backup paths, while the latter may breach strict synchronization requirements of the AI/ML application.

In practice, this means that by the time a fiber link failure is detected and recovery mechanisms are invoked, critical GPU synchronization barriers may already have been missed, forcing rollbacks or restarts of the training process.

4.1.2. How Fast Network Notifications Help

Fast network notification mechanisms could improve the response to fiber link failures and congestion in distributed AI/ML clusters:

* Real-Time Alerts: Nodes adjacent to the failure or congestion could react in the order of sub-millisecond or milliseconds to send lightweight notifications to nodes whose forwarding paths might be affected.

* Action-Oriented Response: Upon receiving the notification, routing and load balancing mechanisms could very quickly shift traffic to backup paths or alternative DC interconnects.

* Granularity: Notifications could carry more detailed information than "link failure/congestion," e.g., indicating specific link utilization, queue buildup or microburst congestion, allowing differentiated responses to different traffic flows.

* Complementary: The fast network notification solutions are complementary to OAM mechanisms and the control plane or management plane information collection mechanisms, such as BFD, IGP and Telemetry, it would bridge the time gap between event onset and slower control plane or telemetry-driven responses, and enable network-wide optimization.

By deploying fast network notifications, large AI/ML workloads can maintain synchronization across data centers even during transient failures or congestion, protecting job completion time and resource utilization.

Existing Approach:

* BFD detects failure after tens of ms

* FRR may cause congestion on backup paths

* Reroute/convergence delays impact GPU sync

* Result: Training stalls, compute resources wasted, job completion delayed

Fast Notifications Approach:

* Forwarding plane detects failure at the level of sub-millisecond

* Fast network notification alerts upstream nodes of failure or congestion in real time

* Regional or global TE steers traffic quickly to alternate link/path without causing new congestion

* Result: Training continues with minimal disruption

5. Fast Network Notifications Problem Statement

A set of problems which need to be considered for fast network notifications are described in the following subsections.

5.1. Information of Fast Network Notifications

The information carried in the fast network notifications, by the originating node, can be one or multiple of the following:

* Event Type: This can be used to indicate the type of events (e.g., failure, congestion, performance degradation, etc.).

* Location of Event: This can be used to indicate the location where the event occurred in the network (e.g., the identifier of the link, the node, or the queue, etc.).

* Fine-grained Network Status information: This can include quantifiable network metrics like link utilization, queue length, level of congestion, link or node delay, jitter, packet loss, etc.

* Path Identification information: This can be used to indicate the path which is affected by the event.

* Flow Identification information: This can include the identification or the 5-tuple of a flow which is affected by the event.

Other information related to the network status change and need to be actioned in a timely manner may also be carried in the fast network notifications. For a specific network scenario, some of the information are mandatory, while others may be optional. There is a need to work on the information model of fast network notifications to better understand what needs to be carried in the notifications.

5.2. Recipients of Fast Network Notifications

The primary purpose of fast network notification is to enable recipient nodes to take prompt actions. Information delivered by fast network notification can be used by recipient nodes to trigger actions in the data plane, and may also be used for actions in the control plane or management plane. The specific mechanisms for realizing such actions are out of the scope of this document. Table 1 provides some illustrative examples of potential recipients of fast network notifications and describes how they may benefit from the information received.

Recipient Type	Role	Example Benefit
Adjacent Routers / Switches	Data-plane neighbors that forward packets	Enable local repair (e.g., FRR, ECMP adjustments)
Non-Adjacent Routers / Switches	Remote upstream forwarding elements	Accelerated awareness of failure/congestions on specific nodes
Ingress Routers / Switches	Traffic entry points of a network domain	Re-map affected flows before forwarding into failed regions
End Hosts / Edge Nodes	Optional subscribers, policy-driven	Adapt sending rate, select alternate uplinks
Network Controller	Optional subscribers, policy-driven	Accelerated awareness of failure/congestion for global TE/LB

Table 1: Recipient Types

Table 1 has three columns. The fist column lists the type of recipients. The second column shows the example of the role that the node is responsible for within the network that could benefit from fast network notifications. The third column indicates examples of how fast notification could benefit the node in fulfilling its role. It should be noted that for different network scenarios, different recipient types may be involved. For a specific scenario, the recipients of fast network notifications may be determined by the reporting node via configuration or signaling mechanisms. In some cases, the recipients may subscribe to specific types of notifications based on their roles or interests. A subscription-based approach allows that each recipient receives only the information relevant to its function, thus may reduce unnecessary overhead.

5.3. Delivery of Fast Network Notifications

Depending on the position and number of the recipient nodes, fast network notifications may be sent via one of the following delivery modes:

* Unicast directly to the recipient node

* Multicast to a group of recipient nodes

* Hop-by-hop to a series of receipt nodes along a specified path

* Flooding in a specified range of the network

The mechanisms to support the above delivery mode needs to make sure the notification is sent to the recipient nodes in a timely manner. It could be based on existing messaging and transport mechanisms, or a new protocol may be introduced. It should be noted that for different network scenarios, different delivery modes may be used.

5.4. Actions to Fast Network Notifications

Once a fast network notification is received, the recipient needs to take appropriate actions to help mitigating the event reported in the fast network notification. The action can be based on the information carried in the fast network notification, or it can be based on both the information in the notification and the information obtained by the recipient in other ways. The action to be performed by the recipient may be explicitly indicated in the notification, or it may be implicitly determined by the type of information carried in the notification. How the actions are performed will be described in other documents produced by the WGs that develop the associated protocols. The possible actions in response to the notification can be, but not limited, to one or multiple of the following:

* Switches all traffic from a path to other available paths

* Steers specific traffic flows to alternate links or paths

* Modifies the load balancing ratio among a group of paths

* Sends the notification further to other recipients

Whether the actions need to be explicitly indicated in the notification, and if so, which ones, requires further consideration. It is noted that in some of the cases as described in Section 5.2, multiple recipients may receive the same notification, then some action may be taken by multiple recipients. The sender of the fast network notification needs to take this into consideration if some coordination in the actions is needed. The mechanism for action coordination is for further study and is out of the scope of this document.

5.5. Scaling Concerns

The challenges of a fast notification system are exacerbated by the the size of the network (number of nodes and links to report issues), the volume of information that needs to be reported, the number of nodes that need to receive the information, and the rate of change of the information.

* Network size is directly related to the amount of information that may need to be reported because each node or link in the network may generate the information described in Section 5.1. The system that is built needs to be able to handle the total data set that could be generated in the network.

* The volume of information that is generated is directly related to the type of information gathered (see Section 5.1), the size of the network (as previously mentioned), and the number of issues that need to be reported. It should be assumed in the design stage that if anything can go wrong, it will. Thus the system must be able to cope with issues reported by a high percentage of the network’s nodes and links.

* As noted in Section 5.2 , notifications may need to be delivered to a number of points in the network. This has a direct impact on the load placed on the network by reporting the information, and combined with the two previous points, this can introduce loading stress on the parts of the network responsible for forwarding and processing notifications.

* Finally, it is important to understand where in the notification system is responsible for handling the effects of rapid changes in the issues that need to be reported. For example, in the case of a link that is "flapping" (going down and up again in a quick cycle) it is crucial to design whether the reports are "damped" at the reporting node, are filtered at some transit node, or are required to reach the receivers. In the case that some node that is not the receiver is required to reduce the notification reports, it is important to clearly specify how this is done and how it is controlled. For example, a device could be configured to only report a degradation at once, but to delay reporting an improvement for a number of seconds to check that it is stable.

6. Operational Considerations

Fast network notifications introduce additional traffic to the network. During network events such as failures or congestion, the notification system itself must not exacerbate the situation; instead, it should actively assist in mitigating the impact. Mechanisms such as rate limiting and traffic prioritization for fast network notifications should be considered. Depending on the operational requirements, fast network notifications should be configurable to be triggered for specific event types, so that it aligns with the network operation policies.

7. IANA Considerations

This document has no IANA actions.

8. Security Considerations

Fast network notifications, if not properly authenticated and rate-limited, could be exploited as a vector for Denial-of-Service (DoS) attacks. An attacker able to inject or flood spurious notifications may trigger unnecessary re-convergence, path changes or repeated state updates, overwhelming both recipient nodes and higher-level applications. An attacker may cause the sender of fast network notifications overwhelmed by making some network state flapping, so that the node is busy with sending notifications. Fast network notifications may reveal sensitive information about the network, in some scenarios such information may be made visible to external entities, either by inspecting the notifications, or by registering as a consumer of the notifications. Implementations must therefore ensure integrity protection, origin authentication, and appropriate rate controls on sending and receiving fast network notification messages.

This document does not specify security mechanisms, but highlights that any solution must consider trust boundaries around notification subscriptions, authorization of notification sources and protection of potentially sensitive operational data. These aspects are expected to be addressed by solution proposals based on deployment requirements and threat models.

9. Acknowledgement

The authors would like to thank Alia Atlas, David Black, Jeffrey Haas, Tony Li, Carlos J. Bernardos, Fan Zhang, Adrian Farrel, Joel Halpern and Dan for their valuable comments and discussion.

10. Contributors

The following people contributed substantially to the content of this document.

Zafar Ali
Cisco
zali@cisco.com

Tianran Zhou
Huawei
zhoutianran@huawei.com

Xuesong Geng
Huawei
gengxuesong@huawei.com

11. References

11.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

11.2. Informative References

[I-D.geng-fantel-fantel-gap-analysis] Geng, X., Dong, J., Cheng, W., Li, D., Zhu, Y., and H. Zhengxin, "Gap Analysis of Fast Notification for Traffic Engineering and Load Balancing", Work in Progress, Internet-Draft, draft-geng-fantel-fantel-gap-analysis-02, 26 February 2026, <https://datatracker.ietf.org/doc/html/draft-geng-fantel-fantel-gap-analysis-02>.

[I-D.geng-fantel-fantel-requirements] Geng, X., Dong, J., Zhu, Y., Li, D., Cheng, W., and C. Liu, "Requirements of Fast Notification for Traffic Engineering and Load Balancing", Work in Progress, Internet-Draft, draft-geng-fantel-fantel-requirements-03, 26 February 2026, <https://datatracker.ietf.org/doc/html/draft-geng-fantel-fantel-requirements-03>.

[RFC3168] Ramakrishnan, K., Floyd, S., and D. Black, "The Addition of Explicit Congestion Notification (ECN) to IP", RFC 3168, DOI 10.17487/RFC3168, September 2001, <https://www.rfc-editor.org/info/rfc3168>.

[RFC4090] Pan, P., Ed., Swallow, G., Ed., and A. Atlas, Ed., "Fast Reroute Extensions to RSVP-TE for LSP Tunnels", RFC 4090, DOI 10.17487/RFC4090, May 2005, <https://www.rfc-editor.org/info/rfc4090>.

[RFC5714] Shand, M. and S. Bryant, "IP Fast Reroute Framework", RFC 5714, DOI 10.17487/RFC5714, January 2010, <https://www.rfc-editor.org/info/rfc5714>.

[RFC5880] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD)", RFC 5880, DOI 10.17487/RFC5880, June 2010, <https://www.rfc-editor.org/info/rfc5880>.

[RFC9197] Brockners, F., Ed., Bhandari, S., Ed., and T. Mizrahi, Ed., "Data Fields for In Situ Operations, Administration, and Maintenance (IOAM)", RFC 9197, DOI 10.17487/RFC9197, May 2022, <https://www.rfc-editor.org/info/rfc9197>.

Authors' Addresses

Jie Dong (editor)
Huawei Technologies
Email: jie.dong@huawei.com

Mike McBride (editor)
Futurewei
Email: mmcbride7@gmail.com

Francois Clad (editor)
Cisco Systems
Email: fclad.ietf@gmail.com

Jeffrey Zhang
HPE
Email: zhaohui.zhang@hpe.com

Yongqing Zhu
China Telecom
Email: zhuyq8@chinatelecom.cn

Xiaohu Xu
China Mobile
Email: xuxiaohu_ietf@hotmail.com

Rui Zhuang
China Mobile
Email: zhuangruiyjy@chinamobile.com

Ran Pang
China Unicom
Email: pangran@chinaunicom.cn

Hao Lu
Tencent
Email: vickkylu@tencent.com

Yadong Liu
Tencent
Email: zeepliu@tencent.com

Luis M. Contreras
Telefonica
Email: luismiguel.contrerasmurillo@telefonica.com

Mehmet Durmus
Turkcell
Email: mehmet.durmus@turkcell.com.tr

Reshad Rahman
Equinix
Email: reshad@yahoo.com