

Quality of Outcome

Agenda

We will talk about:

1. ID changes since last meeting
2. Implementation status and ongoing projects
3. Points for discussion
 - a. Loss calculation

ID Changes

- Added content from draft-teigen-ippm-app-quality-metric-reqs as Motivation and Background
- Addressed many comments.
 - Limit the range of the QoO calculation to the range [0, 100]
 - Fixed many typos and some stylistic issues with the text.
 - Added more references
- Thanks to Will Hawkins, Greg Mirsky, Kevin Smith, Michael Welzl, Mehmet Şükrü Kuran, Dave Taht, Luis Miguel Contreras Murillo and the IPPM chairs for their invaluable feedback!

Outstanding issues with the ID text

- Remove frequent use of "we", and instead use a more formal tone.
- **The throughput aspect of both the measurements and the requirements needs more work.**
- Clarify that latency can be measured in different ways (i.e. not TR-452.1 exclusive)
- **Improve the section about how to create a network requirement**
- Add an example of the passive measurement method that supports the measurement of latency distribution
- Add additional information on whether the list of the measurement parameters in Section 3 is sufficient to "ensure network measurements can be analyzed for precision and confidence".

Ongoing projects related to QoO

We have ongoing collaboration with several partners, working to improve the QoO metric.

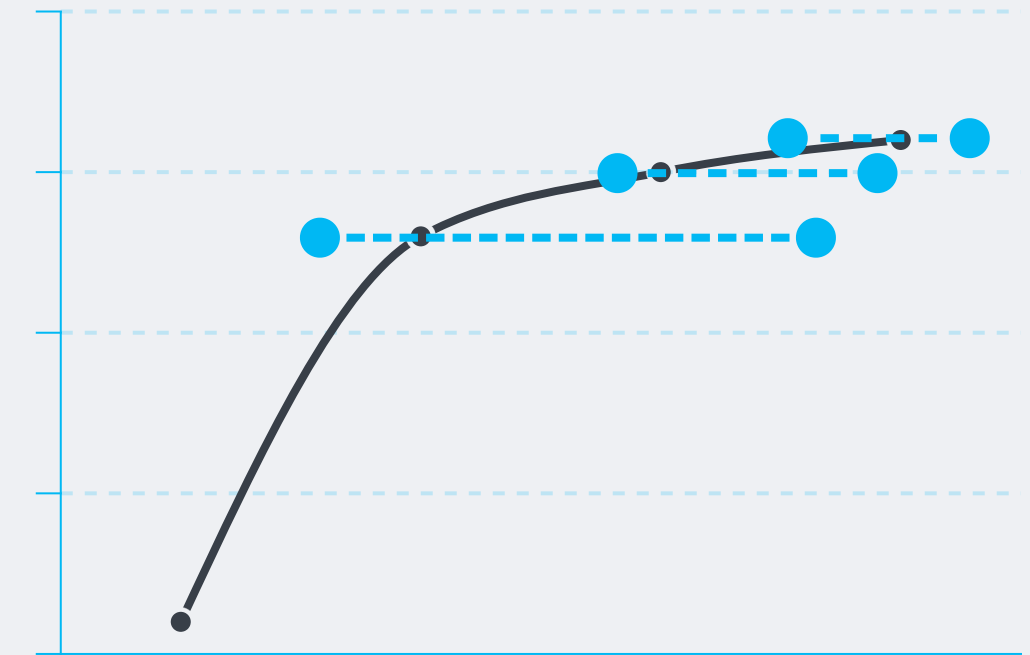
1. Defining network requirements for video conferencing in Messenger and WhatsApp in collaboration with Meta
2. Working with Cablelabs on exposing QoO-based root-cause analysis APIs from CPEs
3. Exploring an open-source implementation of Quality Attenuation + QoO in collaboration with MLab, Will Hawkins and Predictable Network Solutions Ltd. (aka. PNSol)
4. Measuring the impact of WiFi Low Latency APIs on QoO scores in collaboration with an ISP partner
5. Measuring QoO from CPEs and other network equipment with an ISP partner

Please reach out if you want to collaborate!

New learnings — Packet Loss — Question to the audience

- Many apps treats high latency as packet loss
 - For these apps we can model packet loss as high latency
- However, recent experience in lab testing has shown us that this is not always the case.
- To counter this, we suggest adding NRP and NRPoU for packet loss separately.
- It will still be part of the MIN analysis
 - I.e., the app performance is always bound by the weakest network dimension (whether it is packet loss, P99,P95..., etc latency,
- Pro:
 - Can map more applications requirements
- Con:
 - More complexity, both in setting network requirements and in the calculation.
- Thoughts? Input wanted?

As is: Latency NRP and NRPoUs mapped on a CDF of latency measurements:



Suggestion: Adding a packet loss (at L4) NRP and NRPoU. Example:

Network Requirement for Perfection:

0.0% packet loss



Network Requirement

Point of Unusability

5.0% packet loss

Example: If you measure 2.5% packet loss, and the latency is closer to the NRP. The QoO would be 50%

magnus@domos.no

bjorn@domos.no

19.03.2024