Qosium Overview

Qosium is real-time and passive QoS and QoE measurement solution. This section goes into more details of what Qosium truly is and what you can achieve with it.

Table of Contents

1. Qosium

Qosium is a real-time *passive network performance measurement solution*. It measures the *QoS* of existing application traffic in the network and does not burden networks with artificial test traffic. This is the main distinguishing difference to *active network performance measurement tools*. The only requirement for the measured application/service is it runs over IP or directly over Ethernet. No plugins, additional metadata, or any other kind of support is required from the measured target application. As Qosium is a passive measurement solution, the measured application is not aware of the measurement and is not disrupted in any means. With Qosium, everything remains the same for applications and network systems, except that now you know in real-time what the network connection quality is for the applications.

Qosium measures one-way network QoS performance always from the real application's point of view. The solution can perform measurements in real-time, while it can also record the same statistics for later analysis, even per packet. Qosium is at its best when measuring the QoS of real-time applications (e.g., VoIP, industry machinery control, video streams) that pose strict demands for the network performance. However, Qosium does not have limits for applications that are being measured. Qosium does not either have limits on network technologies over which you measure. The underlying access technology is invisible for Qosium measurement. A network connection can be wireless, wired, or a combination of these. Nevertheless, in wireless technologies, Qosium can also collect radio interface statistics, enabling handy analysis and monitoring of both the application QoS and radio link condition and events.

Most commonly, Qosium is used to measure the traffic flowing between two points in the network path. The reason for this is that QoS statistics are always measures between two points in the network. To get one-way results, you need to measure in two points of interest simultaneously. In Qosium, this is referred to as a *two-point measurement*. A wide variety of platforms as *Measurement points* are supported. They can be end-devices, servers, embedded devices, or other network devices located within the network path (e.g., wireless access points, network controllers, VPN servers, and routers). The measurement can be anything from end-to-end QoS measurement to individual link measurement (e.g., a wireless link).

Another option is to use Qosium in a *single-point measurement*. The most significant difference is that the available statistics are more limited than in the two-point measurement, missing the one-way QoS statistics, which you cannot get in a single point. Consider, for example, delay over communications network: the duration of long takes for data to travel from one point to another. You need to know when that particular data has been sent on the other to measure delay. However, certain protocols, such as RTP, allow some QoS calculations to be performed with a single Qosium instance. Even this is still a two-point measurement. The other measurement point is the RTP stream source machine, including its information into the packet header, allowing QoS calculations to be performed in the other end. However, even in this case, you get the results only for the receiving direction in the one-point measurement. Having one-way statistics for both directions would necessitate having another measurement in the other point. Thus, if interested in QoS, Qosium's two-point measurement is the way to go regardless of applications.

Qosium measures the traffic passing through the network interface it is set to monitor. You can set a *Packet filter* to specify the application traffic Qosium considers in the measurement. By using the measurement filters, you can focus the Qosium measurement only on the applications whose QoS you are interested in knowing.

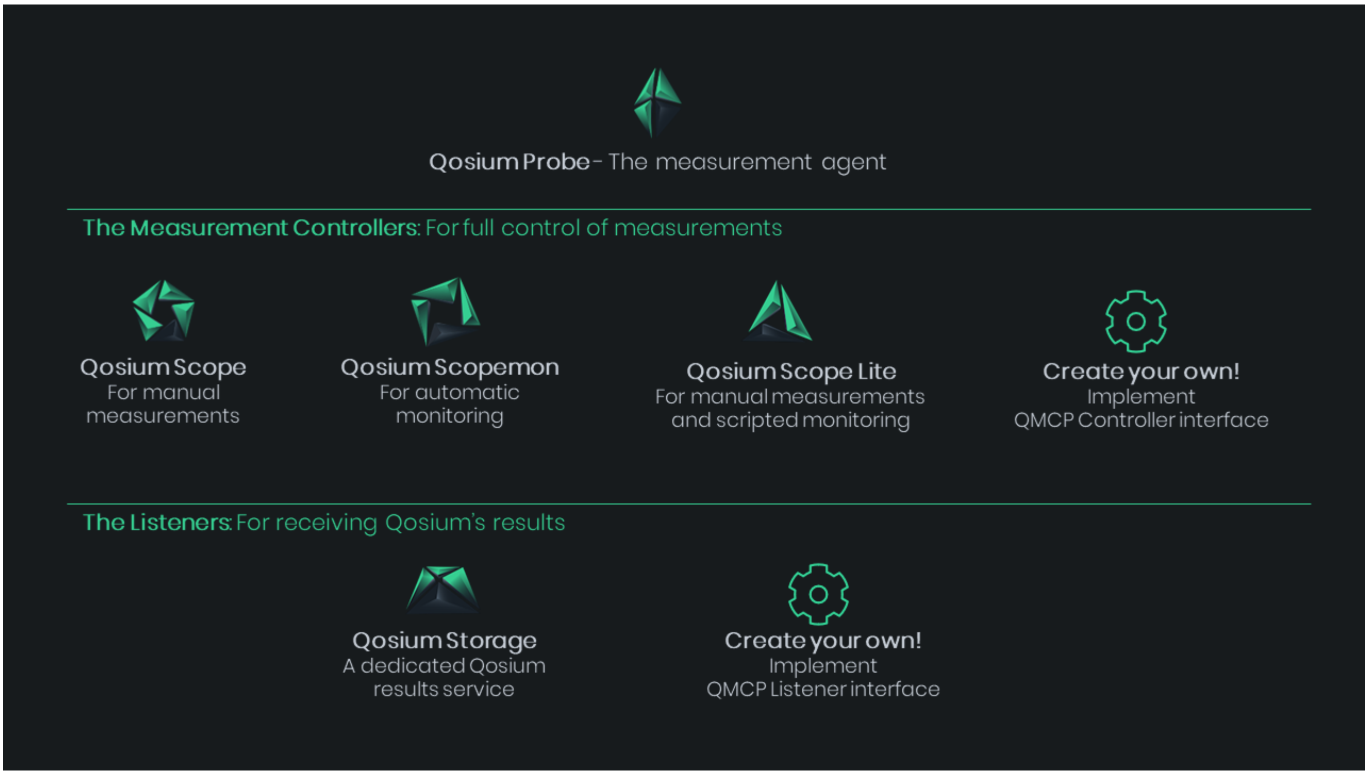
2. Basic Functionality

Qosium is a product family constituting a unity with wide measurement and monitoring capabilities for different purposes. The key component is the measurement agent, **Qosium Probe**. In the best case, Probes are installed directly into the measurement points, i.e., the network devices that are of your interest from the measurement perspective. Sometimes, however, such installation is not possible. In this case, it is conceivable to install Qosium Probe nearby the measurement point, and, for example, mirror the measured traffic to the device where Probe runs. Probe performs most of the actual measurement functionalities, including packet capturing and results calculation. Still, it is the least visible part of Qosium. This is caused by the operational philosophy of the Qosium solution. Probe operates in the background in the platform it is installed, typically as a system service. Thus, Qosium Probes are not used directly, but instead, the measurement is operated by using a *Measurement controller*. As Qosium Probes don’t consume resources from the hosting machine before commanded to start measuring, they can be pre-installed to devices. Qosium Probe is a multi-thread server, allowing multiple simultaneous independent measurements. Thus, only a single Probe per measurement point is required, even though the same measurement point was involved in several separate measurements.

Measurement controllers form a group of products for different purposes. The most visual one is **Qosium Scope**, which is a full-scale analyzer for creating, controlling, and parameterizing Qosium measurements. In addition, it collects and visualizes results in many different ways from numerical values through graphs and meters to heatmaps. All the features of Qosium are available on Scope. Qosium Scope is at its best in manual measurements, such as measurement campaigns, laboratory works, and troubleshooting network issues. Because of its wide visualization capabilities, it can also be used for educational purposes. **Qosium Scope Lite** is another measurement controller for manual measurements. Scope Lite, however, does not come with a nice graphical appearance like Qosium Scope, but operates purely over a command line interface, being, thus, suitable for limited environments. As a command line software, you can utilize Scope Lite through shell scripts.

While Qosium Scope can be used for monitoring, it is not at its best in automated network quality monitoring. Instead, another measurement controller, called **Qosium Scopemon**, is built particularly for this purpose. Select Scopemon when heading for automatic long-term monitoring. It has an optional user interface with simple results visualization, but it is mainly meant for quick informative purposes. The key idea of Scopemon is to perform the measurements automatically and either store the results locally or send them to specific storage for later analysis. After Scopemon is parameterized, its monitoring functionalities are fully automatic. It is ideal for, e.g., monitoring certain critical network paths or critical services that they operate as supposed to. A feature for bringing alarm triggering, for example, due to QoS degradation, is also implemented to Scopemon.

The third category of Qosium products is *Results systems*. These are *Listeners* that collect Qosium measurement results but cannot control Qosium measurements. Kaitotek’s results system is called **Qosium Storage**. It comprises three components: a results listener server, a database optimized for Qosium results, and a visualization server. Qosium Scopemon often accompanies Qosium Storage: Scopemon performs the measurements in the network in a distributed manner, while Storage acts as a centralized place where the results are accessible. The visualization server of Qosium Storage allows customizable overall statistics for quick analysis and, naturally, access to the individual measurement results. The results can be output to the same CSV-format that Qosium’s Measurement controllers use, meaning that the results can be, e.g., drag and dropped back to Qosium Scope for analysis purposes. Alternatively, results can be analyzed further in almost any statistical tool as well. Qosium Storage also offers a REST API for fetching results directly to external systems, enabling, e.g., alarm management. In the future, Qosium Storage will also have geographical visualization capabilities, like Qosium Scope, allowing drawing real-time heatmaps.



Qosium Probe - The measurement agent
The Measurement Controllers: For full control of measurements
Qosium Scope
For manual measurements
Qosium Scopemon
For automatic monitoring
Qosium Scope Lite
For manual measurements and scripted monitoring
Create your own!
Implement QMCP Controller interface
The Listeners: For receiving Qosium's results
Qosium Storage
A dedicated Qosium results service
Create your own!
Implement QMCP Listener interface

3. Measurement Results

The essential results Qosium provides are QoS results, which tell how well the network is performing from the perspective of the monitored application(s)/service(s). However, Qosium brings many other kinds of results, including traffic statistics, flow statistics, radio interface statistics and events, and *QoE* results. Qosium always does the measurement on a packet basis. Still, typically results are monitored as averaged values over the user-defined *Averaging interval*. Collecting also the packet-based results ends up receiving a lot of real-time results data, which is often useful only in detailed troubleshooting.

Qosium also supports taking traffic capturing over a network, which can be handy in some specific troubleshooting situations. However, packet capture is optional, which can also be left out of your Qosium Probe build.

All Qosium results come with a single timestamp. The results can be tied to location when available for Qosium Probe. Thus, this enables vast possibilities in visualization, as, in addition to graphs, meters, and numeric values, you can also draw heatmaps from any Qosium result.

A detailed presentation of the available results is given here.

4. Integrating Qosium's Features to Your Solutions

All Qosium components communicate with each other by using *QMCP*, which is developed by Kaitotek specifically for measurement purposes. Why our own protocol? Because it is the only way to minimize the overhead of measurement control. Generally, a measurement system should always be as light-weight as possible and interfere with the measured target system as little as just necessary. QMCP was the answer to this. In addition to its use in the core of the Qosium solution, it also allows for easy integration to a 3rd party software.

The Results systems, like Qosium Storage, use QMCP Listener API. It is a very straightforward *API* for getting Qosium results reception directly to your systems, for example, managing networks and creating a situation awareness. For integration, Kaitotek provides this API with instructions to customers upon their request. It is currently available for C++ and Java but can be extended to other programming languages as well.

Measurement controllers use QMCP Controller API. It is naturally a bit more complex than the Listener API but still sufficiently easy to take into use. Kaitotek provides this API to customers who plan to develop their own measurement controller. The API allows full control of Qosium measurements. Once implemented, Kaitotek’s Measurement controllers are no longer needed, but you can control Qosium Probes and gather results directly by your own controller. Consider, for example, if you have an intelligent network management solution. With this API, you can get the missing passive QoS measurement capability directly to your solution without a need to bring new parallel full solutions.

5. Glossary

Quality of Service

Indicates the overall performance of the connection/network and its ability to serve applications.

For more information, see our article on Quality of Service.

Quality of Experience

Indicates the overall performance of a network application, and how user's experience is affected by the network conditions.

For more information, see our article on Quality of Experience.

QoS Measurement Control Protocol

Kaitotek's proprietary protocol for controlling measurements and gathering measurement results.

QMCP is a protocol made by Kaitotek to optimize QoS measurement control communications. TCP is used in the transport layer (currently), but QMCP controls its sessions. All Qosium products use QMCP.

Application Programming Interface

An interface in software, which allows a third party to interact with the software programmatically.

API is analogous to a user interface: Whereas a UI offers software-human interaction, an API offers software-software interaction.