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 ... 3
2. Basic Functionality ... 3
3. Measurement Results ... 5
4. Integrating Qosium's Features to Your Solutions ... 5
5. Measurement Results in Qosium ... 6
5.1. Overview ... 6
5.2. Result Files ... 7
5.3. Result Types ... 7
5.4. Average Results ... 7
5.4.1. General Statistics ... 7
5.4.2. Single-Point General Statistics ... 8
5.4.3. Physical Layer Statistics ... 9
5.4.4. Traffic Statistics ... 10
5.4.5. QoS Statistics ... 10
5.4.6. QoE Statistics ... 12
5.5. Packet Results ... 13
5.5.1. Packet Statistics ... 13
5.5.2. Packet QoS Statistics ... 13
5.6. Flow Results ... 14
5.7. Pcap Results ... 15
6. Qosium Editions and Versions ... 16
6.1. Qosium Editions ... 16
6.2. Qosium Family Component Versions ... 17
7. Glossary ... 18

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

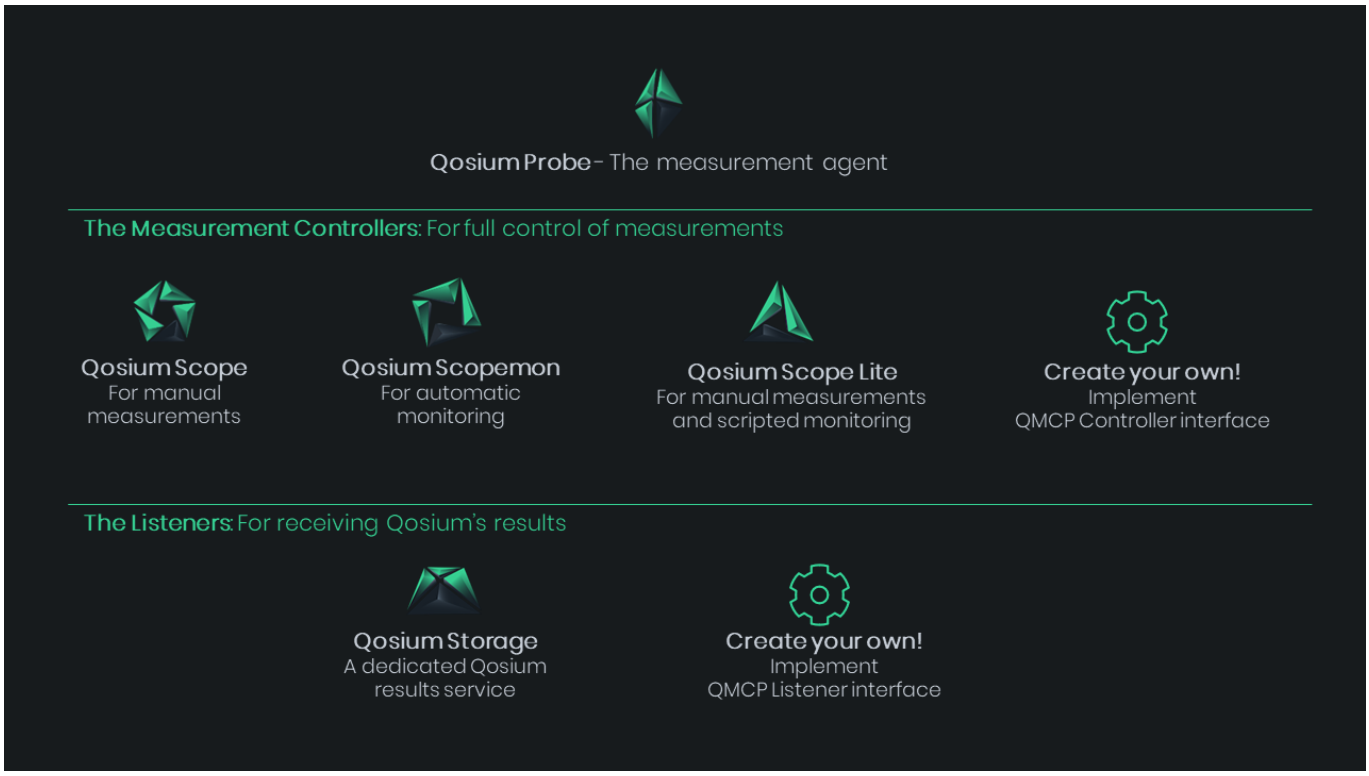
Qosium is a product family constituting a unity with wide measurement and monitoring capabilities for different purposes. The key component is the measurement agent, *Qosium Probe*. In the best case, Probes are installed directly into the measurement points, i.e., the network devices that are of your 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.



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. This 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.

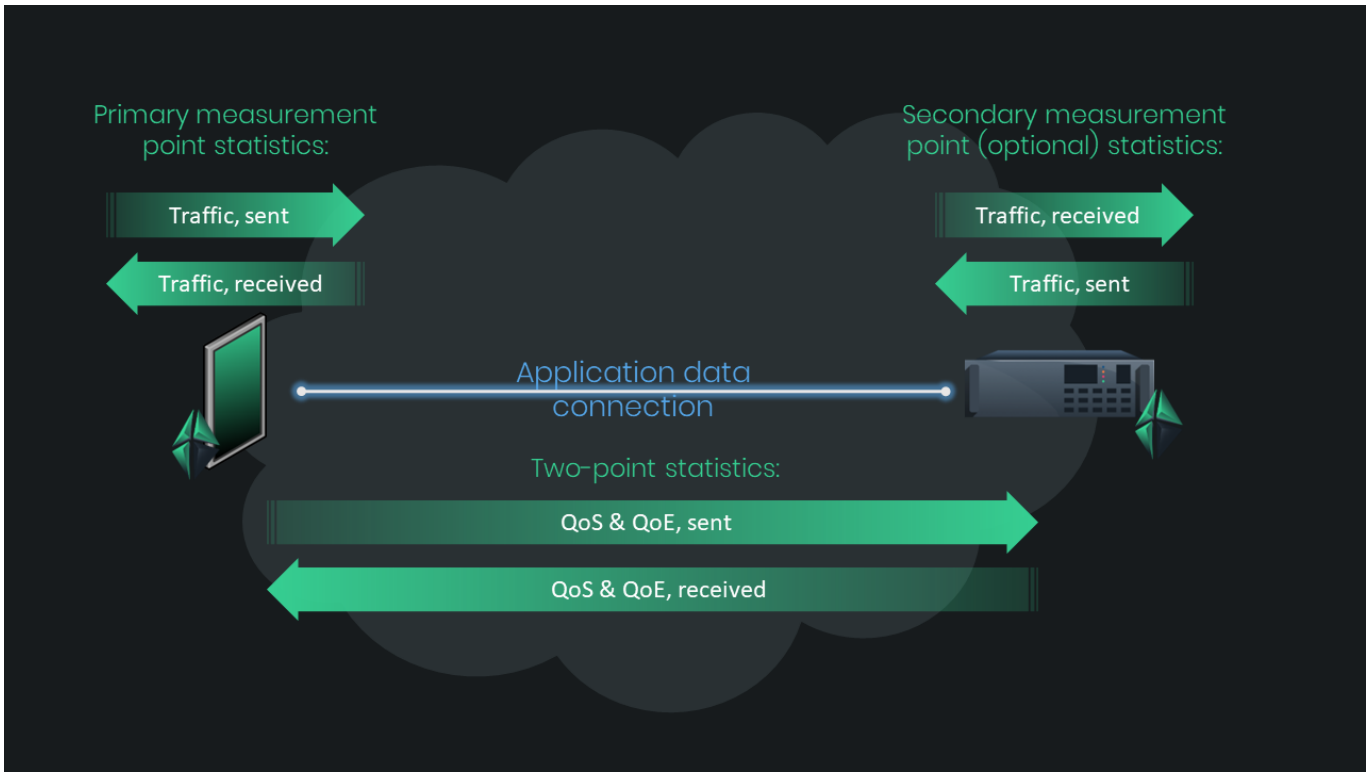
5. Measurement Results in Qosium

Qosium has four types of results: Average, flow, packet, and pcap results.

5.1. Overview

Qosium produces various statistics that can be evaluated in real-time, e.g., with Qosium Scope. The results can also be printed into files for later analysis. During measurement, the results can be injected directly to separate results servers, such as Kaitotek's Qosium Storage or any software that implements the *QMCP* Listener interface. Further, if you have implemented the QMCP Controlling interface, you can control Qosium measurements and gather all the results directly in your software.

It is important to understand how statistics and their direction are dependent on the measurement points. The figure below opens this a bit. As seen, *QoS* and (*QoE* statistics are always two-point statistics. Think, for example, delay/latency: network-wise, it does not exist in a single point. Instead, delay tells how long it takes for a data packet to travel from one point to another. In contrast, traffic statistics (and some general statistics and physical layer statistics as well) are calculated in single points only. Bear in mind also that sent traffic for the primary measurement point is received traffic from the perspective of the secondary measurement point.



5.2. Result Files

The results file formats are the same between different Qosium components. Thus, e.g., an average results file generated by Qosium Storage can be read and analyzed using Qosium Scope. The only thing to notice is that the results files evolve with Qosium versions. This does not happen often, but sometimes. Thus, sometimes an older Qosium Scope cannot read results files generated using a newer Qosium component. In this case, please ask Kaitotek to update your Qosium.

The results files are in *CSV* format, allowing easy import to external statistical tools (e.g., Excel, MATLAB, and R). Absence of a result in a file is notated as *N/A*.

5.3. Result Types

Qosium can produce four types of results:

- **Average results** - Average results are averaged over **Averaging Interval**, which is a parameter you can define. The average results constitute the most important statistic set with over 50 specific statistics to be evaluated.
- **Packet results** - In a two-point measurement, Packet results give accurate QoS statistics for every single measured data packet. They are useful, e.g., in solving flickering QoS problems. In a single-point measurement, these results provide packet basic information.
- **Flow results** - Flow results give a straightforward view of who is communicating with who, how, and how much traffic there is flowing. Flows consuming high bandwidth are easily detected.
- **Pcap results** - These results provide you with full packet captures of desired traffic in the popular Pcap format, allowing you further to analyze the details with protocol analyzers such as Wireshark. This is an optional feature.

Most of the Qosium's statistics are unidirectional, so they are available for received (downlink) and sent (uplink) directions.

The units of the statistics might be different in different tools (e.g., Scope vs. Scope Lite) and views (e.g., numeric view vs. graphs). There can also be some differences in the statistics' names depending on where it is used (e.g., real-time visualization vs. results file).

5.4. Average Results

5.4.1. General Statistics

This category includes general measurement-level statistics that are informative rather than directly related to the networking performance.

Name	Description	Unit
Control Packet Number	The number of the control packet (an increasing counter)	Count
Measurement Duration	The duration of the measurement.	Seconds

Name	Description	Unit
True Averaging Period	The true (accurate) averaging period length. Why is there such a value when one has particularly specified, e.g., 1 s averaging interval? In a non-real-time operating system, it is challenging to achieve highly accurate timings if not operating the program in a way that consumes a lot of processor resources. And, while this is not the way Qosium is built, then the true averaging interval will mostly have a slightly different value from the set one.	Seconds
Timestamp	The timestamp at the time when the current results were calculated. This value is taken at the Primary Probe.	Unix epoch

5.4.2. Single-Point General Statistics

This category includes single-point general statistics that are Probe-specific.

Name	Description	Unit
Malformed packets	The number of packets whose internal structure is not correct.	Packets
Malformed packets, total	The number of packets whose internal structure is not correct (total, measurement)	Packets
Drops, NIC	The number of interface's dropped packets during the measurement. Many Pcap versions (e.g., WinPcap) do not support this yet. This is calculated on the Primary Probe side only.	Packets
Drops, NIC, total	The same as above, but the total value over the measurement.	Packets
Drops, Pcap	The number of Pcap's dropped packets. NOTE: If this starts to increase, it typically means that the Pcap driver is not able to capture the packets as fast as there is traffic, meaning typically that the computer is running out of resources. This is calculated on the Primary Probe side only.	Packets
Drops, Pcap, total	The same as above, but the total value over the measurement.	Packets
Location: Accuracy	Accuracy of the current location.	Meters
Location: Altitude	Altitude of the current location.	Meters
Location: Duration from last update	The age of the last location update. This parameter is handy in checking that the positioning is working correctly.	Seconds
Location: Heading	Heading at the current location if the node is moving.	Degrees
Location: Latitude	Latitude coordinate of the current location.	Degrees [-90°, 90°]

Name	Description	Unit
Location: Longitude	Longitude coordinate of the current location.	Degrees [-180°, 180°]
Location: Mode	Operation mode of the positioning (Qosium specific):	
Location: Speed	Speed of at the current location if the node is moving:	Meters / s
Short packets	The number of packets that are too short for analysis.	Packets
Short packets, total	The number of packets that are too short for analysis (total, measurement).	Packets

5.4.3. Physical Layer Statistics

This category includes single-point Probe-specific statistics related to the physical layer properties, typically radio interface, including signal performance, cell information, etc. The availability of these statistics is dependent on the platform where Qosium Probe is running.

Name	Description	Unit
Base Station MAC address	The current Base Station MAC address (e.g., in WLAN).	String
Cell ID	The cell ID of the current mobile network. Note: in the case of *5G-NSA*, the Cell ID is the Physical Cell ID instead of the Cell ID identifying the LTE cell.	Number
Handover events	This statistic can be directly used to show whether or not a handover occurred during the last averaging interval. Currently, if the serving base station has changed during the averaging interval, value 1 is given as an output and 0 otherwise. Multiple handovers during an averaging interval cannot be currently counted. Thus, also, if the device visits another base station but comes back to the earlier one during the averaging interval, this event cannot be yet detected, and zero handovers are given as an output.	Count
Network type	The type of the network (e.g., LTE, WLAN, 5G).	Coded value / clear text
SINR	The signal to noise and interference ratio at the interface.	dB
Signal (RSRP)	The signal strength (Reference Signal Received Power) at the network interface.	dBm
Signal (RSRQ)	The signal quality (Reference Signal Received Quality) at the network interface.	dB
Signal (RSSI)	The signal strength (Received Signal Strength Indication) at the network interface.	dBm

5.4.4. Traffic Statistics

Traffic information that the single-point measurement can provide includes load, packet sizes, etc. These statistics are available from both primary and secondary Probe.

Name	Description	Unit
Duplicates	The observed duplicate packets (based on the packet identification method).	Packets
Duplicates, total	The observed duplicate packets as above, but summed over the measurement.	Packets
Packet size	Average packet size during the last averaging interval. Note: The packet size means the total size at the link layer, so there is the link-layer header (Ethernet, etc.) also included.	Bytes
Protocol	Dominating protocol number in the measured stream.	Protocol number
Traffic, bits/s	The traffic load in bits/s (offered load and throughput).	Bits / s
Traffic, bytes	The amount of transmitted/received data.	Bytes
Traffic, bytes, total	The amount of transmitted/received data summed over the measurement.	Bytes
Traffic, packets	The amount of transmitted/received packets.	Packets
Traffic, packets, total	The amount of transmitted/received packets summed over the measurement.	Packets
Traffic, packets/s	The traffic load calculated in pkts/s (offered load and throughput).	Packets / s

Be aware of duplicate packets

Some duplicate packets are normal since not all traffic is unique, and also, because of retransmissions (e.g., by TCP), there are real payload-level duplicates. However, if there are a lot of duplicate packets when compared to the number of total packets, it might be an indication that the selected Packet Identification Method does not work correctly in the context of the measured stream. QoS will, generally, not be calculated for duplicated packets. Thus, the overall QoS calculation accuracy may weaken in the presence of duplicated packets.

5.4.5. QoS Statistics

QoS statistics are mostly two-point statistics that are calculated for a flow over a network path. They describe the flow's behavior over the network path and include important statistics like delay, packet loss, etc. Some of the QoS statistics can also be calculated in a single-point measurement for a restricted set of protocols (e.g., RTP and MPEG-2 TS).

All the QoS statistics Qosium produces are one-way statistics between the measurement points.

These statistics are available in a two-point measurement only.

Definition of a connection break

Note: a *connection break* as a Qosium statistic means that there were one or more sequential lost packets detected. For example, the following sequence during an averaging interval contains two connection breaks: a lost packet, a correctly delivered packet, a lost packet, and a correctly delivered packet. Thus, a measured connection break does not necessarily mean a protocol-level connection break.

Name	Description	Unit
Connection break count	The number of connection breaks during the last averaging interval.	Count
Connection break duration	Connection break duration (time-domain)	Seconds
Connection break length	Connection break length (packet domain), often as mean loss burst size.	Packets
Connection break length, total	As previous, but averaged over the whole measurement.	Packets
Connection break maximum duration	The maximum connection break duration in the last averaging interval.	Seconds
Delay	Delay (latency) – one-way – as all the QoS statistics, calculated from network interface to another.	Seconds
Delay threshold crossings	The number of packets whose delay exceeded the set threshold.	Packets
Delay threshold crossings, total	The number of packets whose delay exceeded the set threshold (total, measurement).	Packets
Delay, max	The maximum observed sample delay during the last averaging interval	Seconds
Delay, min	The minimum observed sample delay during the last averaging interval	Seconds
Delay, estimated	This is an estimation, a drift-corrected value, of the delay. It is based on the raw measured delay, but the possible drift of the clocks between the measurement points is strived to be compensated by Kaitotek's intelligent estimation algorithm.	Seconds
Delay samples	The number of Delay samples, i.e., how many values have been used to calculate the QoS statistics	Samples
Jitter	The pure delay variation	Seconds
Jitter threshold crossings	The number of packets whose jitter exceeded the set threshold	Packets

Name	Description	Unit
Jitter threshold crossings, total	The number of packets whose jitter exceeded the set threshold (total, measurement)	Packets
Jitter, max	The maximum observed sample jitter during the last averaging interval	Seconds
Jitter, MA	Moving average jitter (similar as in RTP)	Seconds
Lost Packets	The number of lost packets included in the QoS calculation	Packets
Lost Packets, total	The number of lost packets included in the QoS calculation (total, measurement)	Packets
Packet loss ratio	The ratio of those packets that were lost in the measured network path against all packets	Percentage
Packet loss ratio, total	Packet loss ratio (total, measurement)	Percentage
QoS samples	The number of packets that were used to calculate the QoS statistics.	Packets
Sent info not found	This special statistic is, in a way, *negative packet loss*, meaning that a packet was received but was not indicated sent on the other end. As understood, this is not a normal situation, so having non-zero values here is an indicator of events that should be investigated. See this to understand the possibilities of what is happening.	Packets
Sent info not found, total	The summed value of the previous over the measurement.	Packets
Successful packets	The number of successfully transmitted packets included in the QoS calculation	Packets
Successful packets, total	The number of successfully transmitted packets included in the QoS calculation (total, measurement)	Packets

5.4.6. QoE Statistics

This category includes higher-level statistics refined typically from the QoS statistics. The Quality of Experience (QoE) is closer to the user's observations than the pure QoS. These statistics are available only for certain applications.

While QoE refers to real user tests, the QoE statistics provided here are QoE approximations based on the measured QoS metrics and correlation with measurement data gathered with real user tests.

Name	Description	Unit
GQoSM	QoE evaluation by using GQoSM algorithm	MOS (1-5)
PSQA	QoE evaluation by using PSQA algorithm	MOS (1-5)

5.5. Packet Results

5.5.1. Packet Statistics

These are single-point statistics, giving general information on captured packets. Why not always evaluate full Pcap traces instead, since Qosium also provides them? First, taking Pcap traces from a remote point generates a control load roughly equal to the amount of measured traffic. Getting Qosium's Packet statistics generates only a fraction of that. The same goes for the required storage space as well. Second, Qosium can be used in sensitive environments, where full packet capturing is not even allowed. In many cases, the general packet information is enough, and full captures are taken only when really needed.

Name	Description	Unit
Timestamp	The packet capture time	Microseconds
Direction	Indicates whether the packet received or sent from the perspective of the measurement point.	
Source Address	The source address (Ethernet, IPv4, and IPv6 addresses currently supported) of the packet	String
Destination Address	The destination address (Ethernet, IPv4, and IPv6 addresses currently supported) of the packet	String
Source Port	The source port at the transport layer, when available	
Destination Port	The destination port at the transport layer, when available	
ID	The Identification number of the packet given by the chosen method	
Protocol	This indicates the protocol number above the network layer. If the network layer is absent, this indicates the protocol number above the link layer. Standard protocol numbering is used added with some of Qosium's own definitions.	Protocol number
Fragment offset	The fragment offset of the packet, when fragmented at the IPv4 network layer	8-byte blocks
Size	The total size of the packet	Bytes

5.5.2. Packet QoS Statistics

Name	Description	Unit
Timestamp	The packet capture time at the sending end	Microseconds
Direction	Indicates whether the packet received or sent from the perspective of the primary measurement point.	
Source Address	The source address (Ethernet, IPv4, and IPv6 addresses currently supported) of the packet	String

Name	Description	Unit
Destination Address	The destination address (Ethernet, IPv4, and IPv6 addresses currently supported) of the packet	String
Source Port	The source port at the transport layer, when available	
Destination Port	The destination port at the transport layer, when available	
ID	The Identification number of the packet given by the chosen method	
Protocol	This indicates the protocol number above the network layer. If the network layer is absent, this indicates the protocol number above the link layer. Standard protocol numbering is used added with some of Qosium's own definitions.	Protocol number
Fragment offset	The fragment offset of the packet, when fragmented at the IPv4 network layer	8-byte blocks
Size	The total size of the packet	Bytes
Delay	The duration of time it took the packet to traverse from one measurement point to another.	Microseconds
Jitter	The absolute delay variation between consecutive packets	Microseconds
Jitter (MA)	Packet's delay variation calculated as moving average (similarly as in RTP)	Microseconds
Delivery information	Indicates whether the packet was delivered correctly, lost, etc. This information might not be directly available in all visualization tools, but it is present, e.g., in the packet QoS results file (see below).	

In the Qosium results file, the **Delivery information** is coded as follows:

- `0` - The packet was delivered correctly.
- `1` - The packet was lost.
  - The packet is calculated in the Packet loss ratio statistic.
  - Reception timestamp is missing (because the packet was never received).
  - QoS values are missing.
- `2` - The packet's sent information is not found.
  - The packet is calculated in the Sent info not found statistic.
  - The sent timestamp is missing (because the packet was never marked as sent).
  - QoS values are missing.
  - Read this for more information.

5.6. Flow Results

The flow results present captured traffic as *conversations* between source and destination hosts. What constitutes a conversation is determined by 3 - 5 identifiers:

- Source address (and port, when available)
- Destination address (and port, when available)
- Protocol

When a packet is captured, these identifiers are compared to each existing flow's identifiers. If none of the flows in the current flow results have the exact same identifiers, then a new flow is created instead. If no packets are detected in the given flow for the duration of the Flow timeout parameter, then Qosium determines that the flow has ended and it is removed.

Name	Description	Unit
Source address	The source address (Ethernet, IPv4, and IPv6 addresses currently supported)	string;
Source port	The source port (transport layer). 0 is used if the transport layer does not have ports	string;
Destination address	The destination address (Ethernet, IPv4, and IPv6 addresses currently supported)	Number;
Destination port	The destination port (transport layer). 0 is used if the transport layer does not have ports	Number;
Protocol	This indicates the protocol number above the network layer. If the network layer is absent, this indicates the protocol number above the link layer. Standard protocol numbering is used added with some of Qosium's own definitions.	Protocol number
Flow start time	The time when the flow was first registered.	Microseconds
Flow end time	The time when the flow was terminated. In results files, `N/A` is used if the flow had not ended at the end of the measurement.	Microseconds
Flow age	The age of the flow.	Seconds
Sent traffic	The amount of sent traffic, reported in different ways in different places. For example, Qosium Scope reports this in real time as bits/s, but in results files, this is the total amount of data (bytes).	Bytes
Received traffic	The same as previous but for received direction	Bytes
Inactivity	Indicates the duration, how long the flow has been idle (no packets detected).	Seconds

5.7. Pcap Results

Qosium can capture and save full packet captures as Pcap files. While Qosium itself doesn't have a way to view these files' contents, they can be opened with any packet analyzer tool that supports a well-known Pcap file format. For more information, see How to Take Packet Capture?.

Qosium Without Packet Capture

Being able to capture the full data of packets can be a security hazard in certain environments. By request, we can deliver a version of Qosium where the packet capture feature is completely excluded from the build. This makes Qosium a safe choice even with strict security requirements.

6. Qosium Editions and Versions

This section introduces the different editions of Qosium and the most recent version numbers of each Qosium family component.

6.1. Qosium Editions

There are four different editions of Qosium available:

- **Trial Edition**
- **Evaluation Edition**
- **Academic Edition**
- **Commercial Edition**

Trial and Evaluations Editions are free of charge and meant solely for evaluating Qosium's suitability for your intended purposes.

Trial Edition is downloadable directly after the free registration, but it has functional limitations:

- Qosium Scope: When writing results to disk, only the first 10 seconds are written. In Qosium Scope Lite, results file writing is not allowed at all.
- Qosium Scope holds the statistics in the memory only from the last two minutes.
- Qosium Scope & Qosium Scope Lite: Direct distribution of results is not available.
- Qosium Scope: Measurements stop automatically after 10 minutes.

Evaluation Edition, like Academic and Commercial Editions, is a full version without functional limitations. To get that, contact our support. The Evaluation Edition is available for various environments, unlike *Trial Edition*, which is available only for Windows and Ubuntu.

Academic and Commercial Editions do not have technical differences but differ only in their intended use. *Academic Edition* is meant for research and educational organizations for work without direct commercial target. *Commercial Edition* is for companies and research organisations that, besides research and education, put effort also on commercial customer work with Qosium.

The following table summarizes the main differences between the editions.

Feature	Trial	Evaluation	Academic	Commercial
Unlimited measurements	Yes	Yes	Yes	Yes
Unlimited Probe instances	Yes	Yes	Yes	License dependent
Typical License Period	2 weeks	2-4 weeks	1 year	1 year
Measurement toolset	Yes	Yes	Yes	Yes

Feature	Trial	Evaluation	Academic	Commercial
Monitoring toolset	-	Yes	Yes	Yes
Relieved from functional limits	-	Yes	Yes	Yes
Commercial use allowed	-	-	-	Yes

6.2. Qosium Family Component Versions

The following version numbers per Qosium component are the most recent currently available:

- **Qosium Probe**: 1.7.0.2 (with Core: 2.30.1.2 and QMCP 1.11.27.6)
- **Qosium Scope**: 2.6.0.1
- **Qosium Scopemon**: 1.5.11.1
- **Qosium Scope Lite**: 1.8.0.2
- **Qosium Storage**: 1.9.0

7. Glossary

Quality of Service

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

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

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.

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.

Comma-Separated Values

Originally CSV was only about values that were separated by commas, but nowadays the separation mark can also be something else like a semicolon.

Qosium, for example, uses tabs as separation marks in its CSV-formatted results files.

5G Non-Standalone

A mobile network technology providing a pathway for 5G New Radio workloads to connect to a 4G/LTE core.