eSIP Evolution Series
QueueMetrics Reports and Wallboard Manual

The purpose of this document is to provide instruction on the reports and wallboard functions of QueueMetrics contact center management software (developed by Loway) for eSIP.

Chapter 1. Running a report

To successfully run a report, your system administrator must have configured the correct queues in use on your system. You will find them in the drop-down menu on top of the page. See the section called "Configuring queues" [211] for details on how to do it.

Quick activity reports

The quickest way to obtain an analysis is by selecting the queue and the report you want to analyze and then click on the appropriate time frame below the "Quick activity reports" title on the home page.
The defined time frames are the following:

- Today, Yesterday, The day before yesterday The day in question, starting from midnight to midnight
- Last day, Last 7 days, Last 30 days, Last 90 days The exact time period, starting from the current hour backwards.

QueueMetrics implements an extended dropdown that allows easy access to queues and reports. Extended options could be reached clicking on the magnifier glass located on the right side of the dropdown.

The system will then show the "Answered calls" page, like here below.

On top of the page, you can see a multi-tab menu; by clicking on it you can select which part of the report you are going to see. To go back to the home page, click on the "Home" tab. You can also see all the analyses at once by clicking on the "All" label (this is mostly useful when printing the results to paper).

Agent report
If the user has the appropriate grants, s/he can restrict the analysis to a single agent. This way one can see exactly what one agent did.

To use this feature, select the agent you want to filter by and click on the desired time period in the "Agent report" section of the Home Page.

If you are running an agent-restricted report, you should know that:

- *For inbound traffic, no calls are shown in the "Lost calls" page. This is because an inbound call that has been lost has never been connected to any agent, so there is no way to attribute them to one single agent.*
- *All outbound calls placed by the agent (answered or not) are shown*
- *Agent statistics are shown only for the given agent*
- *As always, activity is restricted by queue - no activity but the one happening on selected queues is shown.*

Custom reports
Custom reports are available by clicking on "Run custom report" from the Home Page. A new menu will appear, asking for custom report parameters:

The meaning is as follows:
- Queue is the queue or composite queue you want to analyze;
- Call filtering criteria can be specified by clicking on the title to open it (see below);

- Start and end date let you select the period you want to analyze, with five-minute resolution or, optionally
- Time period a fixed time period in the past you want to analyze.
- File is the queue_log file you want to analyze. You may want to change it to run reports on a different Asterisk server or on an older archived version of your queue_log. If you run QM on the same machine as Asterisk, the file name should be already correct. Make sure the file is readable to your servlet container. If you use MySQL storage or clusters, the file will look something like "sql:P001" or "cluster:*"
- Time zone offset is to be set if the Asterisk server that created the queue_log file was in a different time zone from the one you are using.
- Join multi-stint calls lets you join together the pieces of the same call if it has been processed by more than one Asterisk queue (see the section called "Multi-stint calls" *[49])*.

By clicking on the "Run custom report" button, you can run the analysis, which output is the same as the "Quick activity report" and will be explained below.

Call search criteria
A number of criteria can be specified to better zoom in on a given set of calls.

- Agent is a specific agent code
- Location is a given location
- Supervision lets you search only for agents that have the current user as their supervisor
- Outcome lets you select a call outcome
- Call features lets you select a call feature code
- Call tags lets you filter by call tag
- Variables lets you filter by a generic variable associated to a call
- Asterisk call-id search by substring on the Asterisk's UniqueID of the call
- Caller search by substring on the Caller-ID
- Wait duration lets you specify the call waiting duration
- Call duration lets you specify a minimum and maximum for the call duration
- Disconnection cause lets you pick a disconnenction reason for the call
- Enter position lets you enter a minimum and maximum enter position (note: this is not tracked for every call)
- Number of attempts lets you select a given number of attempts
- DNIS lets you select calls that have a specific DNIS (if tracked)
- IVR choice lets you select calls that have a specific IVR choice (if tracked)
- Server for clustered systems, lets you select only calls that were processed on a given server
- *Non-contiguous reports let you choose the day(s) of the week and the time periods that you want to include in the reports.*

Note that:
- Criteria involving a full-text search (e.g. Caller) can optionally support full Regular Expressions; if they start by "^" they will be processed as Regular Expressions. E.g. entering "^\d+$" means "find all fields which value is made up only of numeric characters, having at least one character".
- Criteria involving a time-range require you to enter both time values as HH:MM:SS or HH:MM. Invalid values cause the time-range to be ignored.
- Criteria involving an integer range can optionally be left blank; leaving the miminum value blank is the same as entering "0", while leaving the maximum value blank equals to "any number".
- Invalid criteria are discarded and are not used as rectriction. The list of applied criteria is shown on the "Common header" that is available on every page.
- If multiple criteria are input at the same time, they are AND-ed together - that is only calls that suit all given criteria will be shown.
- Running criteria with multi-stint calls may or may not lead to the results you are expecting. See the section called "Multi-stint calls" [49] for more information on this issue.

Search criteria are ignored for real-time reports.

When a report is run with criteria set, all statistics are computed "as if" those were the only calls available; so e.g. agent sessions may yeld different results from what you would get with no criteria.

Persistent user properties
Per-user persistence settings allow user search configuration to be stored and kept even after log off and can be changed and re-stored at any stage.

Custom Reports maintain the latest query parameters entered, even when a user logs off and logs back in, in order to facilitate the work-flow of a user requiring the same reports on a daily basis. The Refresh button allows to clear the input query parameters at any stage.

Preferences
The value set in *default.hourly_slot* acts as a default for a drop down box that is available within the Preferences of the Custom Reports page. User that edit this value basically override the default
and can change it to a set of predefined durations: 1, 2, 3, 5, 10, 15, 20, 30, 60, 90, 120, 240 and 480 minutes. Once this value is edited, it stays the same until the user logs off or changes it again. This option adds persistence (per user) to the options set in Custom Reports. This makes the user-accessible customisation options of QM persistent.

Saving queries
Users holding SQUERY_ADD security keys are allowed to save queries for later usage. If holding also SQUERY_FORALL, saved queries will be visible for all users if required, otherwise query visibility is restricted to that specific user. Queries can be saved from the custom report page, where a bottom line is shown, as reported below.

Saved queries are listed on the home page, grouped between historical reports and realtime reports.

Users can run the saved queries by clicking on it. If holding the SQUERY_ADD security keys, users can edit already saved queries through the pencil icon shown when flying over the query link, as reported below:
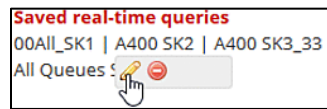
Users can delete their own keys by clicking on the proper key shown when flying over the query link:
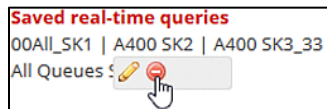
A popup will be shown asking for confirmation
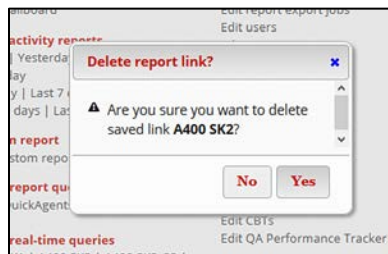
Understanding results: Common header

On the top of each report, a box will be shown showing:

- Which queue or queues were considered for the analysis
- The time period the analysis refers to
- Whether the report is about the whole of the queue or is filtered by some criteria
- The total number of calls processed for this analysis, divided into answered and unanswered ones.
- If running in multi-stint mode, the total number of calls that were joined together There is also a box showing a number of analyses you can export in CSV format.

When running in report mode, QM distinguishes between calls or agent sessions that are complete and calls or agent sessions that are "ongoing" at the moment the report was taken.
Ongoing calls or sessions are usually marked in red and counted separately, as data for them is not definitive and will appear differently if you run further reports.

You should also note that a call that has not been answered yet will be counted as "Ongoing unanswered", though it may well be answered in the nearest future by one of your agents.

In any case, if you need to see calls in progress or whether an agent is logged in, you should rely on the Real-time panels and not on the reports.

TIP: The number of decimals is usually set to 1 but can be changed to 2 via the property:
default.decimalDigits=1

Exporting data from reports

It is possible to export data in Microsoft Excel, Comma-Separated Values (CSV) or XML right from most QueueMetrics panels.

By clicking on the Excel, CSV or XML icons below each report, it is possible to save exactly the same report as seen on screen and then edit it using your favourite number-crunching software.

You must be logged in to download the reports, as you see them on screen.

Users holding the key USR_REPORTS_EXPORT could export the whole analysis to a PDF or XLS files by mean of the two buttons located on the report header, as shown below:

If you are looking for an automatic way to export a full analysis to one file, you should probably have a look at the section called "Automating statistics download: the ROBOT profile" [203].

Understanding results: Answered calls

- The answered calls section deals with calls that were correctly handled by agents. The top panel shows:
- How many calls were handled;
- The average call length (i.e. time the caller spends talking to an operator);
- The maximum and minimum call lengths recorded for the given time period;
- The total call length (for all calls on all operators);
- The average call waiting time (i.e. the time a caller was waiting on a queue before being connected to an operator).
- The minimum and maximum call waiting times on record
- The total waiting time for all handled calls.
- The average initial position of the call in the queue
- The minimum and maximum initial queue positions that have been detected

The queue position coverage: as this information is not tracked for all calls, this ratio shows the average number of calls that had queue position record.

You can see that the information above is reported twice: on the left for all calls, including incomplete ones, and on the right for complete calls only, i.e. excluding calls that were started before or terminated after the given time frame.

Agents on queue
This report shows which agents have been available for the given queue, how many calls each one handled and the percentage of all calls that each one handled.

If calls are connected directly to a phone terminal, QM tries its best to show the corresponding terminal, usually in the format used by Asterisk, like "SIP/303" to signify a SIP phone whose number is 303.

If you connect to H.323 telephones via the OH323 module, the recorded channel names have no meaning and do not refer to a specific terminal; that's why all OH323 calls are grouped together under the label "OH323/-".

Service level agreement
This report shows the distribution of call waiting times. It shows how many calls were answered within a given time frame, usually 120 seconds in 10 second increments (the time frame and increment can be modified by the administrator, if needed - see the section called "Configuring system preferences" [257]).

It is also possible to have two time frames in order to have a higher granularity for shorter time periods -

e.g. having SLA computed in intervals of 5 seconds up to 20 seconds, and in intervals of 10 seconds up to 120 seconds.

You get a percentage of how many calls were answered within X seconds; the percentage includes calls answered in a shorter time frame and therefore grows with time.

The "delta" value you see is the absolute increment, expressed in number of calls, between each time frame.

This metric is computed only on answered calls, i.e. ignoring lost calls . If your SLA is defined in terms of taken and lost calls, see the corresponding metrics "Inclusive SLA" on the section called "Inclusive Service Level Agreement" [36].

Disconnection causes
This report shows the reason why calls were terminated; this means that:
- The agent hung up, or
- The caller hung up, or
- The call was transferred outside the queue and the agent was freed again, or
- The call was ongoing at the time the report was run.

Transfers
This graph shows how many calls were transferred to each extension in the given time frame. This lets you know who is handling exception calls.

When a call is transferred outside the queue system, its length is no more recorded by the queue subsystem; therefore you only get to see the length of the call while the agent was on line.

Answered calls by queue
If more than one queue is in use for the report, this graph shows the relative magnitude of each queue.
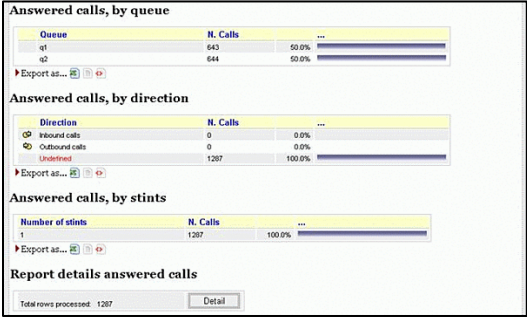
Inbound queues are marked with the symbol while outbound queues use the symbol .

Answered calls by direction
If more than one queue is in use for the report, this graph shows the relative magnitude of inbound versus outbound calls made.

Answered calls, by stints
This graph counts the distribution of multi-stint calls on selected queues. If multi-stint mode is not enable, all calls will have only one stint.

Queue position
This graph shows the initial queue position that the calls had when they joined the queue. For example, a queue position of 1 means that a call was first in line, of 5 means that a call had four other calls in

line before being answered. As the logging of queue positions is a bit inconsistent, some calls might be missing it ("Untracked")

IVR selection

This graph shows the distribution of IVR selections available in the calls processed.

This must be tracked manually in Asterisk - See "Configuring Asterisk for QueueMetrics" below.

DNIS used
This graph shows the distribution of DNIS lines available in the calls processed.

This must be tracked manually in Asterisk - See "Configuring Asterisk for QueueMetrics" below.

Detail of answered calls
This page shows the detail of answered calls. See Chapter 5, *Showing call details* [59].

Understanding results: Unanswered calls

Unanswered calls are calls that were lost, i.e. the caller could not connect to an agent. This usually means that either the caller hung up, fed up with waiting, or the queue system decided to discharge the caller, maybe sending him to voicemail or another queue.

if you run a report with an agent filter, or a supervisor filter, or a location filter, the number of lost calls in the report is usually zero, showing just outgoing calls, if any. This is because the agent is specified only for taken calls and not lost ones, so not deleting them all would show, e.g., the taken call data for one single agent and the lost calls for all of the queue. You can override this behaviour through a configuration switch if you feel this is not correct for you.

The Unanswered calls page looks like the following picture:

Unanswered calls

N.of unanswered calls:	696	N.of unanswered calls:	696
Average wait time before disconnection:	10.0 s.	Average wait time before disconnection:	10.0 s.
Min wait time before disconnection:	10 s.	Min wait time before disconnection:	10 s.
Max wait time before disconnection:	10 s.	Max wait time before disconnection:	10 s.
Total wait time before disconnection:	6,960 s.	Total wait time before disconnection:	6,960 s.
Average queue position at disconnection:	1.0 s.	Average queue position at disconnection:	1.0 s.
Min queue position at disconnection:	1 s.	Min queue position at disconnection:	1 s.
Max queue position at disconnection:	1 s.	Max queue position at disconnection:	1 s.

Disconnection causes

Cause	N. Calls	...	
User abandon	696	100.0%	

Unanswered calls, by queue

Queue	N. Calls	...	
q1	348	50.0%	
q2	348	50.0%	

Unanswered calls - distribution by length

Hangup	N. Calls	Delta	...	
Within 10 seconds:	696		100.0%	
Within 20 seconds:	696	0	100.0%	
Within 30 seconds:	696	0	100.0%	
Within 40 seconds:	696	0	100.0%	
Within 50 seconds:	696	0	100.0%	
Within 60 seconds:	696	0	100.0%	
Within 70 seconds:	696	0	100.0%	
Within 80 seconds:	696	0	100.0%	
Within 90 seconds:	696	0	100.0%	
Within 100 seconds:	696	0	100.0%	
Within 110 seconds:	696	0	100.0%	
Within 120 seconds:	696	0	100.0%	

Inclusive SLA (computed on both answered and unanswered calls)

Hangup	N. Calls	Delta	...	
Within 10 seconds:	2088		75.0%	
Within 20 seconds:	2784	+ 696	100.0%	

The report shows:
- How many calls were lost;
- The average waiting time before disconnection;
- The average queue position at disconnection (i.e. how many calls the queue had to dispatch before connecting the caller to an operator).
- The minimum and maximum wait times
- The minimum and maximum queue position at disconnect.
- The average, minimum and maximum initial queue position, and the coverage given for this computation

As with answered calls, this report is computed twice; the version on the left is for all calls monitored, while the version on the right only holds data for calls that were complete at the moment the analysis was run.

Disconnection causes

This report shows the relative magnitude of disconnection causes, that are:
- The caller hung up, or

- The queue timed out and discharged the caller (if this feature is enabled by the queue configuration - see Chapter 18, Configuring Asterisk for QueueMetrics [345]), or
- The caller exited the queue by pressing a key (if this feature is enabled by the queue configuration).

Unanswered calls, by queue
If more than one queue is in use for the report, this graph shows the relative magnitude of each queue.

Unanswered calls - distribution by length
This report is functionally equivalent to "Service level agreement" in the Answered calls section (see the section called "Service level agreement" [32]), but is computed on lost calls. It shows how many calls were hung up within a given time frame, usually 120 seconds in 10 second increments (the time frame and increment can be modified by the administrator, if needed - see the section called "Configuring system preferences" [257]).

You get a percentage of how many calls were lost within X seconds; the percentage includes calls lost in a shorter time frame and therefore grows with time.

The "delta" value you see is the absolute increment, expressed in number of calls, between each time frame.

Inclusive Service Level Agreement
The inclusive SLA corresponds to the Service Level Agreement metrics shown on the section called "Service level agreement" [32], with the difference that it is computed taking into consideration both answered and unanswered calls.

Unanswered calls by key press
If there are any calls that are were set unanswered because the caller pressed a key to exit the queue, this graph shows which keys were pressed and how many calls were terminated for that reason.

Unanswered calls, by stints
This graph tells the stint distribution of unanswered calls. It corresponds to the graph called "Answered calls, by stints".

All calls, by stints
This graph tells the stint distribution of all processed calls. It corresponds to the sum of the graphs called "Answered calls, by stints" and "Unanswered calls, by stints"

Enter queue positions
This graph shows the initial queue position that the calls had when they joined the queue. For example, a queue position of 1 means that a call was first in line, of 5 means that a call had four other calls in line before being answered. As the logging of queue positions is a bit inconsistent, some calls might be missing it ("Untracked")

Enter queue positions for all calls
This graph shows the initial queue positions for both answered and unanswered calls.

IVR selection
This graph shows the distribution of IVR selections available for lost calls.

This must be tracked manually in Asterisk - See "Configuring Asterisk for QueueMetrics" below.
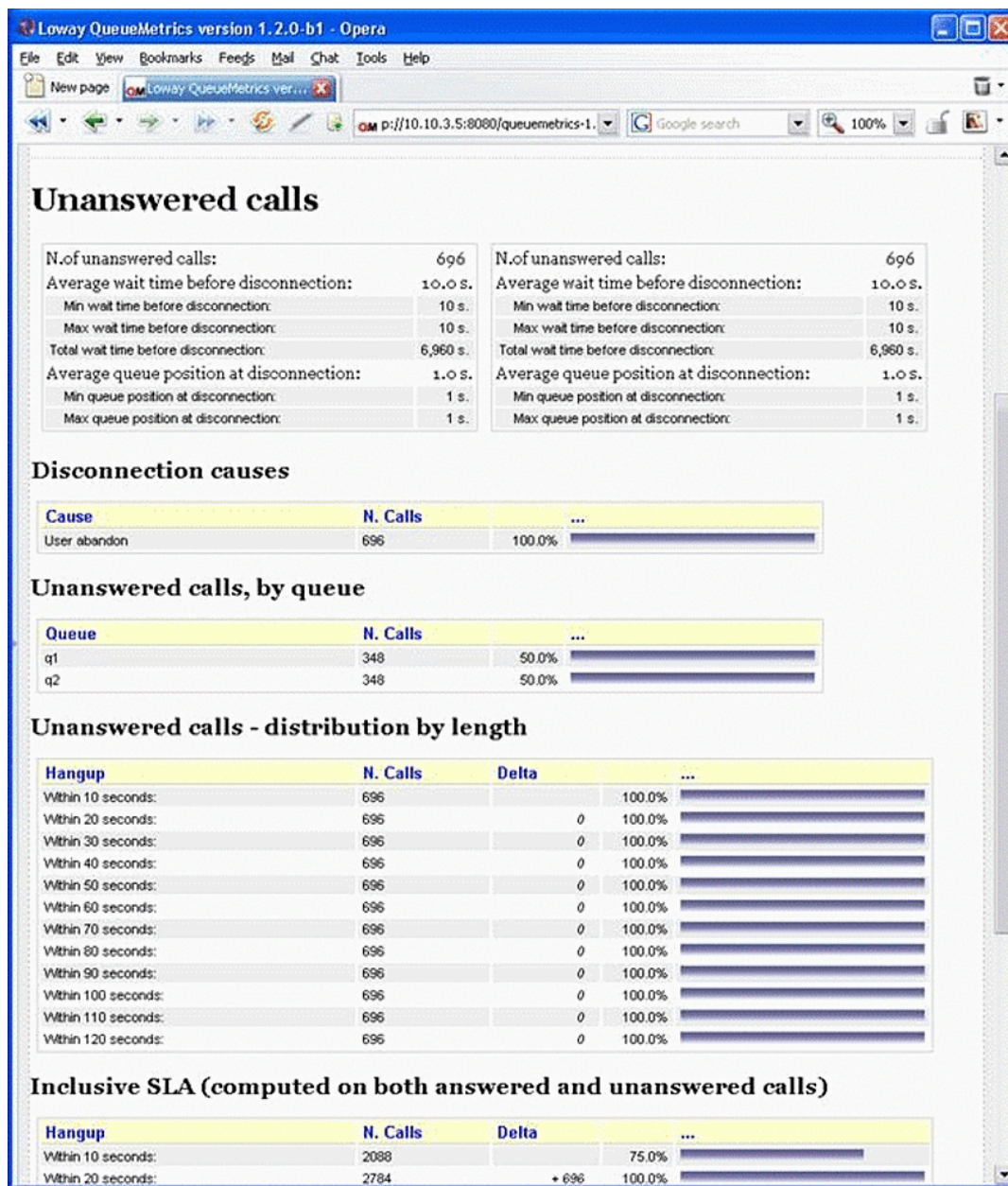
IVR selection for all calls
This graph shows the distribution of IVR selections for all calls (taken and lost) available in the report.

DNIS used
This graph shows the distribution of DNIS lines that lead to lost calls.

This must be tracked manually in Asterisk - See "Configuring Asterisk for QueueMetrics" below.

DNIS used, for all calls
This graph shows the distribution of DNIS lines for all calls (taken and lost) available in the report.

Details of unanswered calls
This page shows full details of unanswered calls the section called "Detail of unanswered calls" [63].

Understanding results: Area code report

If the Caller*ID is present, it is possible to break down both answered and unanswered calls to specific area codes by clicking on the "Area code analysis" button.

By selecting a number of caller id digits to search upon and a starting digit position, you get a number of statistics grouped by area codes.

This report gives an immediate check of the geographical origin of calls handled by your call center. It is possible to export all the reports as needed.
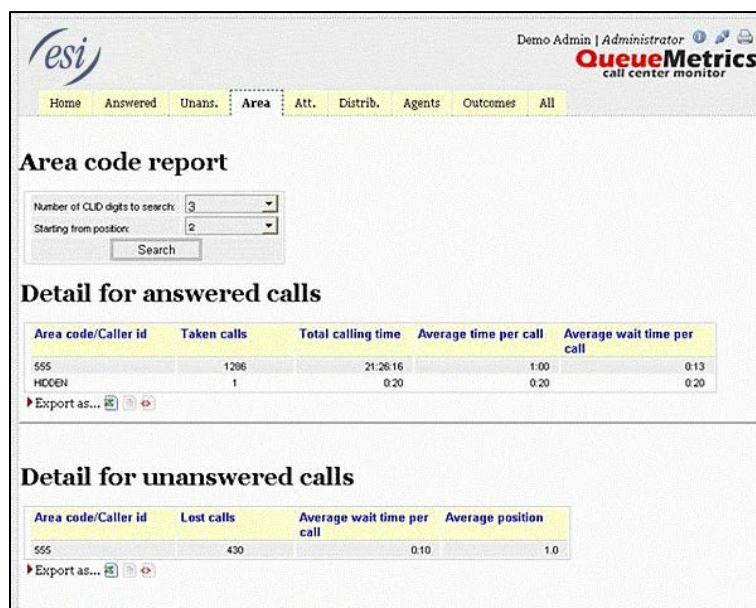
Understanding results: Inbound ACD call attempts

When running an inbound call center, it is very important to determine the reason why a call is delayed: are your clients refusing to answer? Did they forget to log off before leaving their workplace? The inbound ACD call attempts metrics try to answer to these questions.

As these metrics are not usually recorded by Asterisk, you'll have to patch and recompile your Asterisk system in order to gather them - see the section the section called "Enabling ACD call attempts recording on Asterisk 1.0 and 1.2" [349]. If you do not do so, the metrics presented here will always appear zeroed out. With Asterisk 1.4, this feature should be automatically enabled with no need to patch the source code.

This page shows the following pieces of information:
- How many agent attempts were made, i.e. how many times the agent's telephones were rung in total
- The average number of attempts that were necessary for a taken call; the minimum, maximum and total attempts made that resulted in a taken call

- The average number of attempts that were necessary for a lost call; the minimum, maximum and total attempts made that resulted in a lost call

ACD attempts by terminal
This graph breaks down agent attempts by the agent that was called. The following pieces of information are extracted for each agent:

- N. of lost agent attempts (i.e. the agent was called but not responding)
- The average ring time for lost attempts
- The total ringing time for lost calls
- The number of taken agent attempts (i.e. calls answered)
- The average ring duration for taken calls
- The total ring time for taken calls

ACD attempts by queue

The following metrics are extracted and broken down by queue:

- N. of lost agent attempts (i.e. the agent was called but not responding)
- The average ring time for lost attempts
- The total ringing time for lost calls
- The number of taken agent attempts (i.e. calls answered)
- The average ring duration for taken calls
- The total ring time for taken calls

Understanding results: Call distribution

The call distribution report shows when calls were handled, when calls were lost and the average wait times broken down by period.

All percentages are calculated on the call class they belong to, i.e. a 50% of "Unanswered calls" on one day means that 50% of all unanswered calls for the period happened during that day, not that 50% of calls were lost.

For each metrics, the total number of calls is shown, together with average, minimum and maximum times. Graphs are plotted on the total number of calls broken down and on the averages.

It is possible to change the interval in the Hourly graphs, so that you can have reports break down calls.
e.g. by half-hours or hour quarters, by changing a value in the QueueMetrics master configuration file.

Call distribution per day

Calls, both taken and lost, are shown per specific day. Days with no events are not shown. The total numbers of call lengths, wait time for answered calls and wait time for unanswered calls are plotted for each day. Sales and contacts are also shown on a daily basis.

The Schedule Adherence report shows the number of distinct agents that were detected during the given period. This makes it possible to detect the number of different people that had been working on a given moment.

The Queue Length report shows the average length of the queue for each period, giving minimums and maximums. The Steps computation shows how fast each queue progresses during the period, expresses in steps per hour.

Note: giving a meaning to the Queue Length reports may be hard in the case of composite queues.
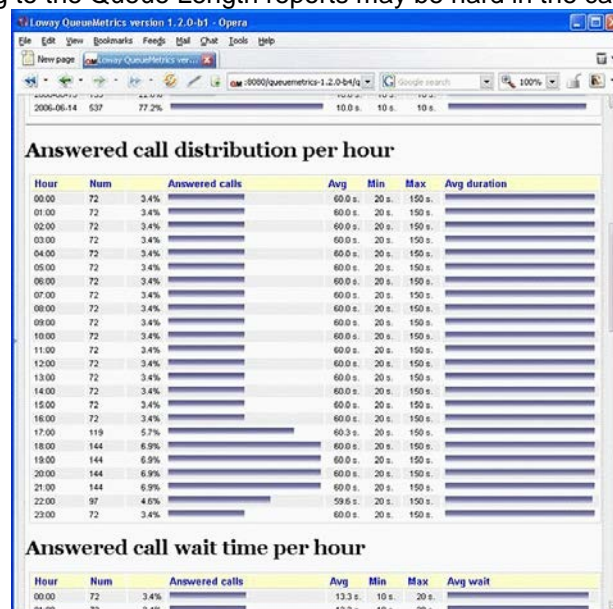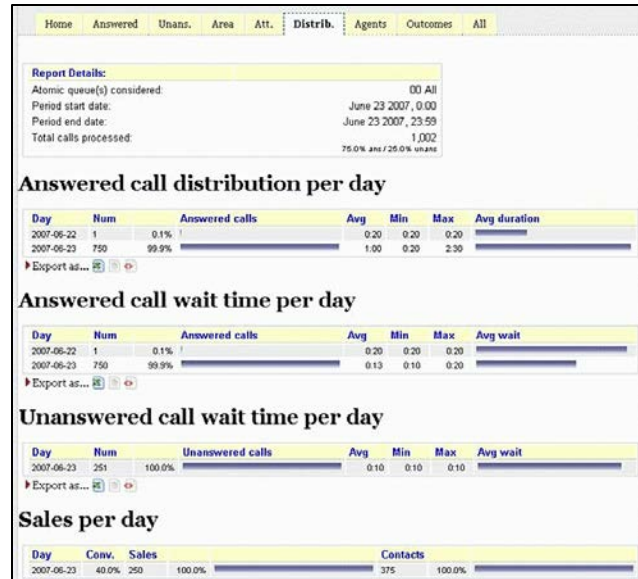
Call distribution per hour
Events are shown on a 24-hour distribution. If this graph appears to be incorrect, you have to run a "Custom report" setting the time zone accordingly (see the section called "Custom reports" [24]).

The total numbers of call lengths, wait time for answered calls and wait time for unanswered calls, together with sales and contacts, are plotted for each hourly interval. The size of hourly intervals can be controlled by the *default.hourly_slot* configuration property, making it possible to run this reports based on 30-minute, 20-minute or 15-minute intervals.

Call distribution per day of week
This report shows the weekly behaviour of your queues. The longer the analysis period, the more significant its results will be.

The total numbers of call lengths, wait time for answered calls and wait time for unanswered calls are plotted for each day of the week.
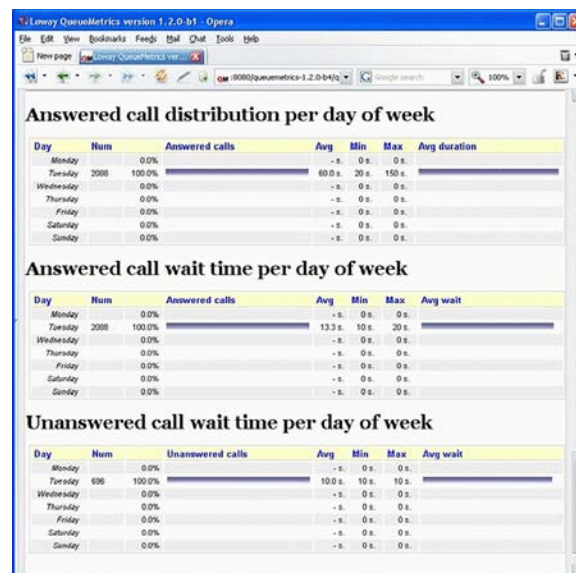
Understanding results: Agent activity

Agent activity refers to the behaviour of Asterisk defined agents. If you connect you queues straight to telephone terminals, this section will always be empty.

Each agent may be flagged as being a member of four priority groups:
- Main: the agents usually answering the queue
- Spill: the agents answering the queue if all "Main" agents are busy or unavailable
- Wrap: the agents answering the queue if all "Main" and "Spill" agents are busy or unavailable
- Undefined: this agent is not a member of any priority group for this queue

This feature is useful if priority groups are used in the queue configuration. If they are not used, just assign all agents to "Main" for each queue.

If an unknown agent appears on a queue, it will be marked as "Undefined", written in red.

Agent names are written in blue and are clickable, if you click on them in any of the graphs, you will be lead to a popup that detail the logon and pause history for that agent.

As a default, QueueMetrics will show and count an agent session if and only if the agent handled at least one call during this session. This may not be what you want when you use pause codes - an agent may log on and immediately go on pause to do some back-end activities. If this is the case, you should set the configuration option *default.useRawAgentSessions* to *true* to see all agent sessions.

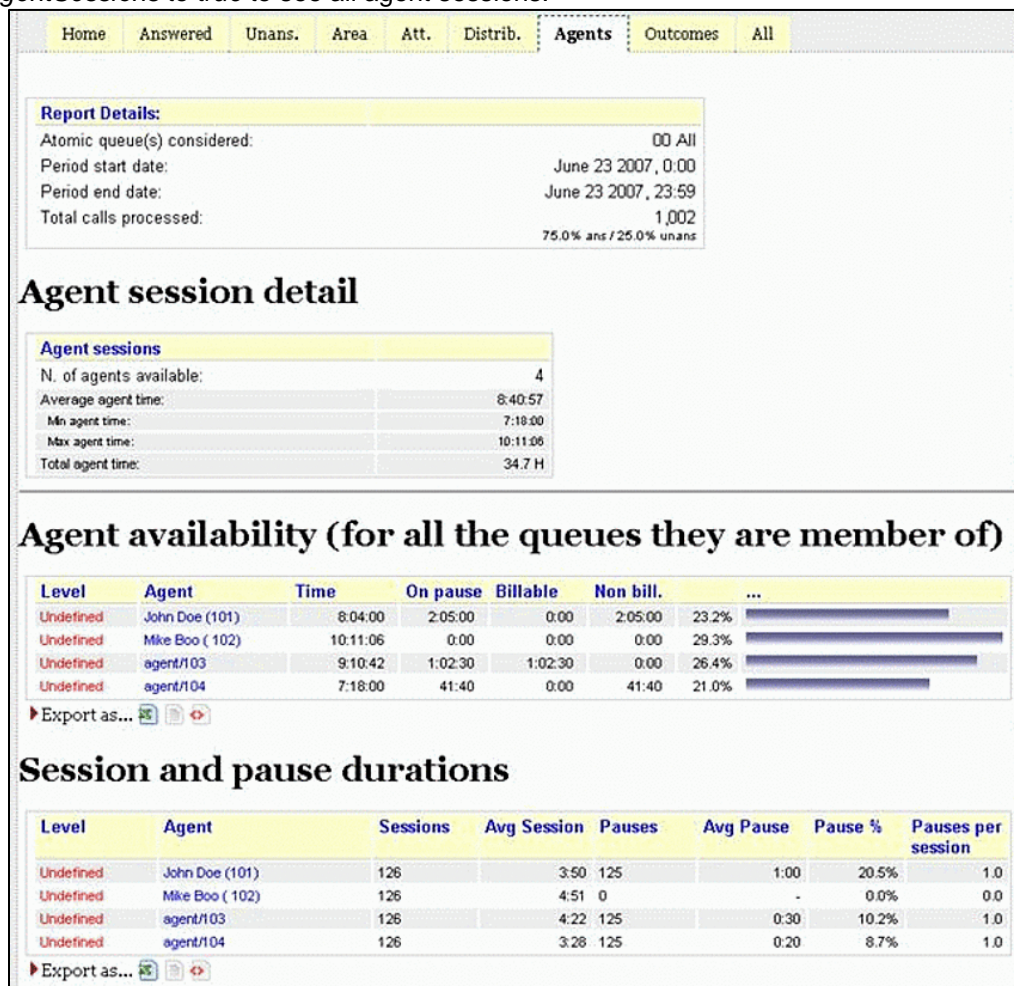| Home	Answered	Unans.	Area	Att.	Distrib.	Agents	Outcomes	All

Report Details:

Atomic queue(s) considered:	00 All
Period start date:	June 23 2007, 0:00
Period end date:	June 23 2007, 23:59
Total calls processed:	1,002
	75.0% ans / 25.0% unans

Agent session detail

Agent sessions

N. of agents available:	4
Average agent time:	8:40:57
Min agent time:	7:18:00
Max agent time:	10:11:06
Total agent time:	34.7 H

Agent availability (for all the queues they are member of)

Level	Agent	Time	On pause	Billable	Non bill.	...
Undefined	John Doe (101)	8:04:00	2:05:00	0:00	2:05:00	23.2%
Undefined	Mike Boo (102)	10:11:06	0:00	0:00	0:00	29.3%
Undefined	agent/103	9:10:42	1:02:30	1:02:30	0:00	26.4%
Undefined	agent/104	7:18:00	41:40	0:00	41:40	21.0%

▶ Export as...

Session and pause durations

Level	Agent	Sessions	Avg Session	Pauses	Avg Pause	Pause %	Pauses per session
Undefined	John Doe (101)	126	3:50	125	1:00	20.5%	1.0
Undefined	Mike Boo (102)	126	4:51	0	-	0.0%	0.0
Undefined	agent/103	126	4:22	125	0:30	10.2%	1.0
Undefined	agent/104	126	3:28	125	0:20	8.7%	1.0

▶ Export as...

The report shows:
- How many agents were available for the queue. To be considered available, an agent must have logged in and taken at least one call.
- How much time all agents have been available
- The average agent available time
- The minimum and maximum agent session durations
- The total billable and not billable pause times

Agent availability
This graph shows which agents were available during the specified time frame and the percentage of agents' available time each one cumulated.

This time is calculated per all queues any agent is a member of, as the act of logging on is in general for the whole system and not specific to one single queue.

For each agent, the total time on pause - if any - is computed and broken down as "Billable" or "Not billable" - see the section on Pause Codes.

The "overlapping" is the amount of time that an agent was paused BUT taking calls. As you can pause in the middle of a call (maybe your system doesn't allow this, but it is conceivable) you could have a period where you are paused BUT on conversation as well.

The % written next to the graph is the percentage of the pause time.

Session and pause duration
For each agent, the total number of sessions and pauses is computed (session time is already deducted of pause time). For both sessions and pauses, an average length is computed.

The "Pause percentage" is how much time an agent was on pause compared to available time.

The "Pauses per session" computes how many pauses - on average - each agent makes for each log-in session.

These metrics should be considered according to your call center rules on pauses and time-out.

Answered calls (for selected queues)

Level	Agent	N. calls	Total call time	Avg call time	Avg wait time	% of queue
Undefined	John Doe (101)	125	5:12:30	2:30	0:10	16.6%
Undefined	Mike Boo (102)	250	2:25:50	0:35	0:15	33.3%
Undefined	agent/103	125	2:05:00	1:00	0:10	16.6%
Undefined	agent/104	251	2:47:00	0:39	0:15	33.4%

▶ Export as...

Answered calls by service group

Level	N. calls	Total call time	Avg call time	% of queue
Main	0	0:00	-	0.0%
Wrap	0	0:00	-	0.0%
Spill	0	0:00	-	0.0%
Undefined	751	12:30:20	0:59	100.0%

▶ Export as...

Answered calls by location

Location	N. calls	Total call time	Avg call time	Avg wait time	% of queue
Main Location	125	5:12:30	2:30	0:10	16.6%
Secondary location	250	2:25:50	0:35	0:15	33.3%
-	376	4:52:00	0:46	0:13	50.1%

▶ Export as...

Session details

Total sessions: 504	Detail
Total pause activity detail	Detail

Answered calls for selected queues
This graph shows who of your agents answered calls for the queues you selected. The number of calls, together with total and average call durations is computed accordingly.

Answered calls by service groups
This graph show which priority levels handled calls for your queue. This shows whether your main line is staffed enough to handle the load of incoming calls.

Session details
By clicking on the "Detail" button, a new page is shown, detailing each agent session that was recorded.

For each agent session, the start and end times are recorded, together with the total duration in seconds. If the agent logs on via the call back function, the designated call back extension is shown.
The number of pauses and the total pause time in seconds is shown.

The "Srv" column tells you on which server an agent was working in case you set up a cluster of Asterisk servers.

It is possible to sort the table for each title, in either descending or ascending order. To do this, click once on the desired title for descending sort, and twice for ascending sort. Once the table is sorted, an arrow symbol will appear close to the title, so you know on which column it was sorted last. As the sorting is done on the client machine, it may take a while with very large tables.

Pause activity details
This table shows the specific pauses that each agent took and the pause code that was entered for each pause. It also shows whether the pause taken was considered to be billable or non-billable.

Agent history popup
If you click on an agent's name, a new popup will appear with full history for that agent. You can scroll in it as needed by using arrow keys or the wheel of your mouse.

A complete description of the popup is available in the Report Details: the section called "Popup of agent activity" [127] .

Understanding results: Call outcomes

If your agents are entering Pause codes or Call outcomes, the "Outcomes" tab will let you report on the information they just entered.

The top panel will display an overview of the situation, showing:
- How much billable time there has been on this system, broken down by ACD/call time ("agent available time") and billable activities (agent on pause)
- The total non billable time (e.g. lunch, breaks)
- The total number of Contacts, Qualified Contacts and Sales, as defined by call outcome codes
- The Sales per Hour (SPH), Qualified Contacts per hour (QCPH) and Contacts per Hour (CPH) ratios
- The Conversion index that is the percentage of sales over the total number of sales and contacts.

Further down the page, you can find details explaining Billable and Non-billable activities, with average, minimum and maximum session durations, and a percentage on all activities of the same kind.

Call results, by outcomes

Call outcome	Type	Total	Taken	Lost	%	
21	Unknown	1677	1677	0	37.5%	
20	Unknown	1120	1120	0	25.0%	
-	None	1678	559	1119	37.5%	

► Export as...

Billable activities

Activity	N. Times	Tot. Time	Avg.	Min.	Max.	%

► Export as...

Non billable activities

Activity	N. Times		Tot. Time	Avg.	Min.	Max.	%	
30		1119	12:25:40	0:39	0:20	1:00	72.7%	
31		559	4:39:30	0:30	0:30	0:30	27.3%	

► Export as...

Detailed agent report

Agent name	Avail.		Bill.		NonB.		Sales	QC.	Cont.	SPH	QCPH	CPH	CONV	QCONV
Mike Boo (102)	45:12:58	100.0%	0:00	0.0%	0:00	0.0%	0	0	0	0.0	0.0	0.0	-	-
A 103	40:42:47	100.0%	0:00	0.0%	4:39:30	11.4%	0	0	0	0.0	0.0	0.0	-	-
John Doe (101)	35:44:39	100.0%	0:00	0.0%	9:19:00	26.1%	0	0	0	0.0	0.0	0.0	-	-
A 104	32:37:19	100.0%	0:00	0.0%	3:06:40	9.5%	0	0	0	0.0	0.0	0.0	-	-

► Export as...

The Detailed Agent Report will show, for each agent:
- The Available (ACD) time, as an absolute value and a percentage of its total time logged on
- The Billable time, as an absolute value and a percentage of its total time logged on
- The Non-Billable time, as an absolute value and a percentage of its total time logged on
- The number of Sales And Contacts the agent had (if a sale is counted a s both a Sale and a Contact, it's counted only once as a Sale)
- The Sales per Hour (SPH) and Contacts per Hour (CPH) ratios for this agent
- The Conversion ratio, that is the percentage of sales over the total number of sales and contacts.

How are Call Outcomes calculated?
The idea is that a call can be a Contact, or a specialized contact that is a Qualified Contact, or a specialized Qualified Contact that is a Sale

This is needed because all the SPH, CPH and QCPH are calculated not on the totals of each class, but on sums of that class and generic types, like:
> S = Number of Sales
> C = Number of Contacts
> Q = Number of Qualified Contacts

CPH = C / (logon time - not billable pause time) QCPH = Q / (logon time - not billable pause time) SPH = S / (logon time - not billable pause time)

Conversion indexes are calculated as:
CO% = S / C QC% = S / Q

Multi-stint calls
In QueueMetrics, we define a multi-stint call as a call that was processed on more than one queue, with one or more queue terminating it for timeout, transfers or key exits.

In the standard QueueMetrics reporting mode, this call would be seen as a series of "lost calls" on one or more queues, possibly followed by a taken call if the call was answered at all; the system does not notice that those events happened on the same call.

Running QueueMetrics in multi-stint mode, calls will be grouped together based on the call's Unique- ID, and a single call will be rebuilt as a multi-stint call so that:
- The call looks like it was handled on the first queue it was presented on; the queue enter time and queue position are those of the first queue.
- The call will be considered "answered" if the last stint is an answered call, "lost" in all other cases

- The wait time will be the sum of the wait times on different queues (if there are intermediate wait times, like those for IVR menus, they will not be counted)
- The talking time and agent taken the call will be taken from the last taken call
- All stats (number of call, call distribution, etc) will be counted on the new multi-stint calls.

Limitations and side-effects
Multi-stint calls aren't for everyone. There are a number of limitations and side effects you should be aware of before attempting to run QueueMetrics in this mode:
- Calls are grouped by the Asterisk Unique-ID code; this means that if different call stints happen on different servers in a cluster, they will not be grouped together
- All queues the call passed on must be included in the report. If you include only the master queue, stints on other queues will not be seen.
- Because of the previous bullet, it is usually better to configure separate "wrap up" or "timeout" queues, that is, instead of having both a Sales and a Support queue that will send people to the

General queue on timeout, it would be better to have "Sales", "General-sales" and "Support", "General-support", even if "General-sales" and "General-support" are actually aliases of the same queue.
- All stints of all calls must be included between the starting and ending report times. Stints that start before the start data or end after it will be ignored.
- Run time and memory will be comparatively more than a standard analysis, as the grouping and additional data stored take their toll on the system
- Stint-grouping does not work for real-time analysis.

On versions of QueueMetrics up to 1.6.3, calls are filtered by search criteria before being aggregated in multi-stint mode. This may lead to problems when you want to use filtering criteria in multi-stint mode, where only some stints match the critera while others do not. T To avoid this issue, on newer versions of QueueMetrics calls are joined together in multi-stint mode before criteria are applied to the aggregated results.

Multi-stint calls in QueueMetrics
If you run calls with multi-stint mode enabled, the string "Multi-stint calls joined together" will appear on the top panel, and the number of joined together calls will be shown.

The distribution of taken calls by stints will be shown in the "Answered calls" tab:

The distribution of lost calls by stints will be shown in the "Lost calls" tab; aggregate calls by stints will also be shown in the "Lost Calls" page:
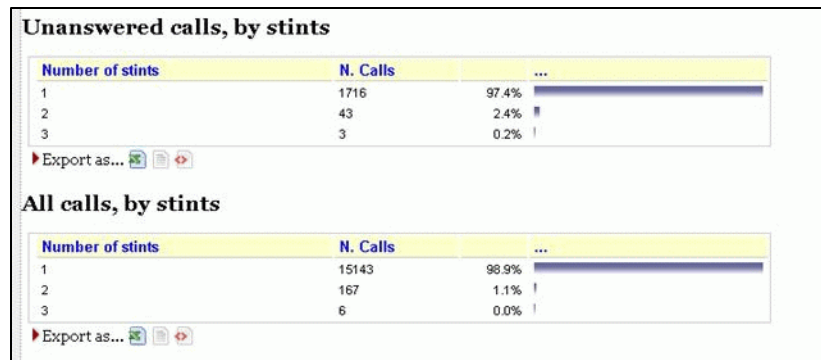
Details of multi-stints calls are displayed in the call details pop-up, in a separate tab. See Chapter 5,
Showing call details [59].

Exporting call sets from QueueMetrics

There is a need to make it possible for external parties to review the call processing as done on QueueMetrics or do an external QA monitoring on them; and similarly, there is an opportunity for a QueueMetrics to do the same thing for external third-parties.

In order to make this possible, we need to have a way for QueueMetrics to import/export both call records (with associated audio/video recordings where present) and related QA data.

Exporting calls - an overview
The main problems that arise from exporting calls are:
- Selecting the set of calls that have to be exported, and
- Retrieving all audio calls for export

We need to be able to select freely a number of calls for export, given one or more export criteria; we want to be able to review the results before they are final, and we want to exclude specific calls when reviewing.

The retrieval of audio causes a similar problem; first, audio retrieval was not really made for batch access, so seek times for individual files may be in the order of one per seconds; secondarily, those files may well require a large disk space when preparing the batch.

The call export feature works in batches, that is, at any given time there are a set of batches that an administrator creates and that are to be exported. Each of them has a name and a status.

This is the life-cycle of a batch:
- An administrator creates a batch for a given data export needed and gives it a meaningful name, e.g. "Client X week 02/10". The batch is now in state Open.
- Users holding the correct key will be able to add calls to the batch, that is, when they run a report in QM, on the "Taken calls" page they see a button that invites them to add the selected set of calls to the open batches. They can repeat this process as many times as it is needed. If the same call is added multiple times to the batch, it only appears once.
- When the batch is ready, the administrator closes it. When the state is Closed, it is not possible anymore to add calls to that batch.
- When the batch is Open or Closed, it is possible for the administrators to see the list of calls in the batch and to listen to their audio/video attachments. Individual calls may be flagged as "Do not send"- those calls will appear in the batch but will not be show in the outgoing records.
- When the batch is finalized, the administrator will flag it as "Ready to send". Batches that are "Ready to send" are actually being built by QM - it may take a while to create them and download the audio files required.
- When the batch is finished processing, it will appear as "Sent".
- A batch can be deleted at any time by the administrator, unless it is in status "Ready To Send".

The graph provides a visual representation of the whole process:
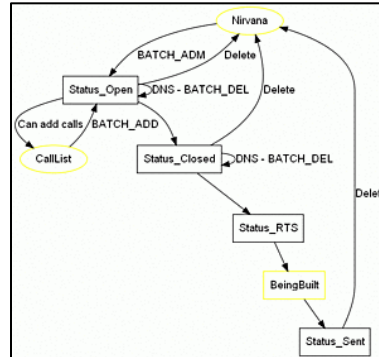
The process of building a batch may take a while - therefore there is a transaction that simply keeps displaying a page in a browser that shows a progress bar while the project is being exported.

Exporting calls in practice
In order to export calls, an administrator must first create a batch. To do this, they go to the Home Page / Import/Export calls:

From here you select List Export Jobs.
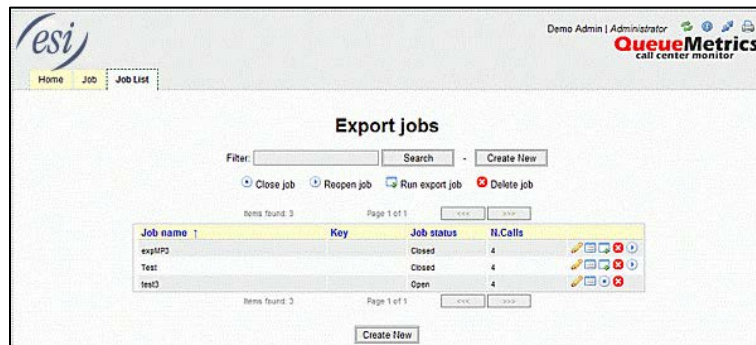
You can then create a new job.

When configuring the job, you have to specify:
- A **name** for the job. This will be the name of the folder that will be created including the call details and the audio files.
- A **security key** so that only some agents can add calls to this job.
- An **export folder** - it can be the same for all jobs, and be on the server QM is on. It should be writable by the java process that runs QM,.
- An **implementor**, that is, a data format for Manifest file of the job. See below for more details.
- If the implementor requires them, you can specify a set of **parameters**.

These properties can be changed through the lifetime of the job.

From now on, users accessing the Detail of Taken Calls will see a gadget by the end of the page like the one shown here:

This basically lets you add all the calls above to the export job you select. If the export gadget should not be visible, you have to add it manually to the current report (its code is OD02 - see Block OD02 the section called "OD02 - Add to export job" [75]).

After you add sone calls to a job, the import transaction will confirm the add through a popup screen that displays the number of calls imported and the number of calls rejected (because they were already a member of this job). Only jobs in state Open will be availble for adding calls.

You can now see a list of calls from the Export Jobs page:

You can search calls within the current job and toggle their inclusion in the exported list by clicking on the reload icon next to each call.

When the job is ready for shipment, the administrator first has to Close it and then to Export it. This will take a while. Audio files will be retrieved through the curremtly configured Pluggable Module for audio records and will be saved in the job folder.

Output format
Thought the actual attributes used are based on the Implementor module used, the following are common attributes.

Batch attributes
- Name
- Disk path (must be accessible to Java)
- Created by, on date
- Closed by, on date
- Sent by, on date
- Video (yes/no)
- QA (yes/no)

Each call in the batch has the following attributes:

- Type: "T" taken "L" lost (initially we will only have Taken calls)
- Cluster-ID
- Server-ID
- Entered at
- Wait time
- Talk time
- Caller
- Agent
- Queue
- Call status
- Call status type (e.g Sale)
- N. of audio pieces
- Names of the audio pieces, comma-separated (a single call may have multiple recordings)

Batch disk format
A call batch appears on disk as a folder under the system call batch folder. The folder is created if not present (it must be in a Java-writable location). When the batch is in status "Ready to send", the audio files are copied to this folder; at the end of it all, a manifest file that includes the details is written. Each downloaded audio file is renamed in order to be unique and coherent, typical file names may be:
0000123-1.wav
0000123-2.mov
Both files are about the 123rd call, the first one being an audio recording and the second one a video recording.

When the batch is in status "Sent" QM no longer cares about the disk representation - it can be moved, sent as FTP, compressed and encrypted, whatever.

Manifest file format
The manifest file format should be chosen by the sysadmin - it is implemented as an abstract class for ease of change.

- It will be called Manifest.xml
- It will include a set of <call> entries including all data as per the previous section "Batch attributes"

Available implementors

HTTP file transfer

This is the basic implementor and produces an XML file.

HTTP MP3 file transfer

This implementor does two things:
- produces a basic manifest
- retrieves QA valuation data if present

If audio files are in MP3, will insert or set ID3v2 tags so that the title of the MP3 contains information about the call.

The following ID3 tags are created:
- Call details
- caller id
- queue
- dnis
- call lenght
- start time

- lenght
- agent
- Call status
- Date of Grading
- Time of Grading
- Grader ID and/or Name
- Overall Grade
- Graders Comment

This implementor DOES NOT transcode files to MP3 - they must already be in MP3 format or you should provide an external batch script to do the conversion. More details are provided in the following paragraph.

MP3 conversions on the fly
QueueMetrics lets you able to perform an external batch script call in order to convert call files to different format like, for example, wav files in mp3s. This feature is applied only for export jobs implemented with the "HTTP MP3 File Transfer" object. No external calls will be made for "HTTP File Transfer" enabled export jobs.

Assuming to have this type of export job parameters: Job name: MyJob Export folder (on server): /var/ spool Implementor: HTTP MP3 File Transfer

The work flow followed by QueueMetrics, for HTTP MP3 File Transfer enabled jobs, will be like depicted below.

For each call in the job and for each file associated to a specific call:
1. The file will be stored on server folder /var/spool/MyJob folder
2. A new temporary folder will be created on /var/spool/MyJob folder
3. An external bash script will be called. The script will receive, as parameters:
   - The full file name of the file to be converted
   - The full name of the temporary folder created
   - The name of the job as defined in the job definition page
   - The parameters string as defined in the job definition page
4. The script should convert the file in the preferred format and should place the result in the provided temporary folder. QueueMetrics will wait for the conversion termination
5. QueueMetrics will move (not copy) the full conversion result it will find in the temporary folder to the original destination folder (in this case /var/spool/MyJob). Please note that there should be more than one file resulting in the conversion (like, for example, a preview quality and a hi-res quality .mp3 files) and QueueMetrics will copy all of that.
6. The temporary directory will be removed by QueueMetrics
7. QueueMetrics will publish in the manifest all the files found in the temporary folder. These files will be associated to the specific call and the number of chunks published in the manifest will reflect this.

When all files in the job are properly downloaded and converted, QueueMetrics will add all relevant QA information only to files with name ending with .mp3
Please note that:
1. The external script to be called by QueueMetrics should be specified in the configuration.properties files through the key "export.conversionCommand". It should be executable by TomCat. If no key was defined, the HTTP MP3 File Transfer will skip all actions specified in the 2, 3, 4, 5, 6 steps above described (i.e. it will simply download the files and apply QA informations to eventually present .mp3 files)
2. If the external script is not present in the server and/or QueueMetrics is not able to run it, this will be signaled in the <errors> field present in the manifest (one for each call). In this case QueueMetrics will publish in the manifest the original file name.
3. QueueMetrics will not delete the original file from the /var/spool/MyJob folder. If you need to have it deleted, your conversion script should do it.
4. QueueMetrics will not publish the original file name in the manifest, unless if exceptions were raised in the conversion/move process. If you need to have the original file published in the manifest, your script should move (not copy) it to the temporary folder.
5. If some exception is raised when moving files or calling the external bash script, QueueMetrics will publish the original

file name in the manifest.
6. If the conversion script generates a (set) of filename with name(s) already present in the /var/spool/ MyJob folder, QueueMetrics will rename it (them) prepending the name(s) with a random 5 digit number followed by an underscore sign. The new name(s) will be published in the manifest

To summarize the overall process, a simple example of working environment is provided: In the configuration.properties is the key: export.conversionCommand=/usr/local/apache-tomcat-5.5.25/webapps/ROOT/testbatch and in the /usr/local/apache-tomcat-5.5.25 is an executable testbatch script: [root@qmmachine ROOT]# ls -la testbatch
-rwxr-xr-x 1 root root 87 Mar 10 05:17 testbatch

In this simple example, the batch script copies the original file in the temporary folder (with a not unique name in this example)

```
#!/bin/bash

        # $1   is   the  source file
        # $2   is   the  output directory
        # $3   is   the  job name
        # $4   is   the  parameter field defined in the job definition page


cp $1 $2/testresult
```
The resulting working folder will contain something like:

```
        ....
        -rw-r--r--      1   root   root      408516  Mar  10   05:18   17619_testresult
        -rw-r--r--      1   root   root       50110  Mar  10   05:18   18542_testresult
        -rw-r--r--      1   root   root      884372  Mar  10   05:18   18795_testresult
        -rw-r--r--      1   root   root        4740  Mar  10   05:19   20110304_084640-99.wav
        -rw-r--r--      1   root   root      419784  Mar  10   05:19   20110304_084640.wav-99-1.avi
        ....
```

And the manifest something like:

```
....
<call>
<files>testresult,2071_testresult</files>
<errors></errors>
<chunks>2</chunks>
<uniqueid>3033212900899824</uniqueid>
<videocall>true</videocall>
<callid>4006</callid>
....
```

Chapter 2. Showing call details

As shown above, QM lets you see the very detail of calls handled by Asterisk.

Detail of answered calls

For each answered call, the following information is shown:

- Date and time for the call;
- The Caller-ID, if available (the Caller-ID format may differ according to your local Telco - in some countries it include the full name of the caller, in others it might be a number and in others it may be unavailable at all);
- The queue that handled the call;
- The total waiting time before the agent was connected;
- The duration of the call, talking to an agent;
- The initial position of the call
- The cause of disconnection;
- Which agent or terminal handled the call.
- How many agent attempts were made before this call was answered
- The call completion code your agents entered

- How many stints make up this call
- The server that handled this call (in the case of clusters) Optionally other information could be shown:
- The asterisk unique ID associated to each call
- An icon that opens a new web page with an URL user customizable (useful for proprietary CRM integrations). To enable these two columns, the keys default.crmapp and default.showAstClid have to be correctly set. Please read the section called "Configuring system preferences" [257] and Appendix C, System preferences [383] for further details.

It is possible to sort the table for each title, in either descending or ascending order. To do this, click once on the desired title for descending sort, and twice for ascending sort. Once the table is sorted, an arrow symbol will appear close to the title, so you know on which column it was sorted last. As the sorting is done on the client machine, it may take a while with very large tables.

If you click on the small icon on the right, it will be possible to see the details of the call, including:
- Asterisk's internal Call-ID code
- The call date and time
- The caller-id (if any)
- The agent handling the call
- The call duration
- The wait time
- The disconnection cause
- The extension the call was transferred to
- The URL that was linked to this call through the Queue() command, if any
- The call status code
- The server that handled this call
- The sound files (one or more) that were recorded for this call (see below).

If the call is ongoing and you have the special grants to do so, a red scissor icon might appear next to the call status to allow for brute-force call closure. See the section the section called "Closing ongoing calls" [357] for further details.

The call may also display:
- If the call has more than one stint - that is was rebuilt through multi-stint mode - then a list of stints is visible. If a call has only one stint, then the Stints tab is not needed and appears disabled. See the section called "Multi-stint calls" [49]
- If a call has events on it - that is specific items happening at points in time that go beyond what is displayed for the main call - they are shown in the Events tab. If not, they are disabled. The ones currently tracked are IVR traversal events, failed call connection attempts and music-on-hold events.
- A set of markers (Note: QueueMetrics till release 14.10.4 shows "tags" instead of markers).
- A link to open the QA form(s) available for that call

Listening to answered calls

If there are audio recordings, they will be shown by the bottom of the call details page. If the call was monitored, i.e. recorded to disk, a number of sound files may be shown. By clicking on a sound file you can listen to it straight from your browser.

You can optionally have other types of media linked to a specific call, for example chat transcripts, free text or documents. They will be opened in the browser.

Please note that:
- The recorded file name must contain the Asterisk Call ID for QM to relate it to the call - see the section called "Listening to recorded calls using QM" [346] for tips on how to configure Asterisk correctly to implement this feature;
- The audio storage on the Asterisk server must be readable by the servlet container;
- You must have the correct sound codecs to listen to the sound file on your PC. WAV files usually work out of the box but are comparatively quite big, while GSM files require an additional codec pack on most Windows machines

but consume disk storage much more efficiently. The best compromise is usually to use the WAV49 format on Asterisk, that is played natively by Windows machines but has a compression and sound quality comparable to the GSM format. The HTML5 player requires WAV or MP3 files.

- Asterisk will usually record two different sound files - one for the caller and the other for the agent and will then mix them together at the end of the call. If this does not happen automatically, you might find two different sound files, named "-in" and "-out", each of which contains the voice of one of the parties. If your call is a multi-stint call, you may find a number of different sound files for it.
- It is possible to use different PMs to handle different audio needs - see the section called "Listening to calls using Pluggable Modules (PM)" [276].
- If generated by Asterisk, QueueMetrics can display a variety of other file type call attachments, ie. calls that are shown with a file extension that is not necessarily a sound file (image, video, audio, text or application files).

It is also possible to add markers in order to keep notes about "points of interest" for the call at hand, as shown in the example below:
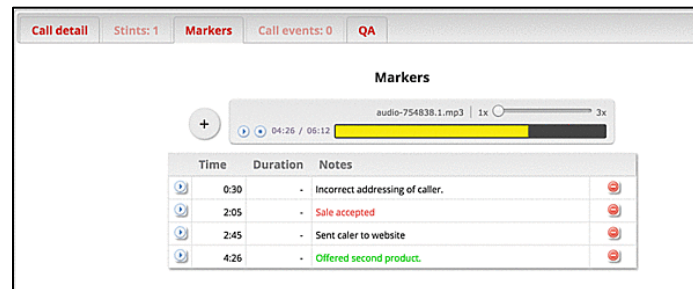
By clicking on the Play icon next to a marker, the correct audio recording is loaded in the embedded player and it is started at the point in time specified in the marker.

Until release 14.10.4, QueueMetrics used the name "tags" instead of "markers".

The HTML5 Audio Player and Markers
Since QueueMetrics 15.02, it is possible to play audio straight in the browser using an HTML5-enabled audio player.

This has many advantages:

- You can play the audio straight in the browser, jumping back and forth as needed
- You can speed-up the audio playing, in order to take less time to listen to calls
- It integrates with the Markers editor, so you can add a comment at a specific point in time and see it later. When you click on a comment, the audio editor starts playing at that very point in time.

This is extermely powerful, and can be done either on the Call Details panel (under the Markers tab) or on the QA form.

In order for this to work, you need:

- A modern browser supporting HTML5 audio playback
- The system property 'audio.html5player' must be set to true.
- If audio files are hosted outside of QueueMetrics, your HTTP server must support Http-Range requests. QueueMetrics supports Http-Range file streaming natively.
- Files must be stored in a format that is compatible with the browser. Presenty this means that your audio files should either be saved as WAV or MP3. GSM and WAV49 files will likely not work.
- Streaming is not supported for encoded audio recordings.

Users having the capacity of adding and removing markers should hold the keys CALLMONITOR_ADDTAGS and CALLMONITOR_DELTAGS respectively.

The following pieces of information are tracked through the Markers page:

- System ID and Asterisk unique-id; marker author and time
- Time offset and duration (in seconds). You can leave the duration field blank. Values can be input as an integer number of seconds or as MM:SS
- A title (max 200 characters). Newline characters are allowed.
- A color for display (this may be used to set, say, all sales in red, all issues in blue, etc).

Detail of unanswered calls

The unanswered calls detail is quite similar to that of answered calls.

The following data are shown:
- Date and time of the lost call;
- The Agent that placed the call (if it's outbound) or blank if inbound;
- Caller-ID;
- Queue that handled the call;
- Disconnection cause;
- Position at disconnection, if available;
- Waiting time before disconnection, if available;
- The initial position of the call when it joined the queue, if available;
- The number of Agent attempts made before disconnection;
- The call code, if entered (this might be added automatically, e.g. by outbound diallers marking unsuccessful attempts as "unanswered" versus "fax" or "voicemail")
- The key pressed on disconnection (if any)
- The number of stints this call has
- The server that handled the call

Optionally other information could be shown:
- The asterisk unique ID associated to each call
- An icon that opens a new web page with an URL user customizable (useful for proprietary CRM integrations). To enable these two columns, the keys default.crmapp and default.showAstClid have to be correctly set. Please read the section called "Configuring system preferences" [257] and Appendix C, System preferences [383] for further details.

Please note that on a queue timeout, Asterisk will not report the waiting time, as it is fixed and same as the queue timeout.

It is possible to sort the table for each column, in either descending and ascending order. To do this, click once on the desired title for descending sort, and twice for ascending sort. Once the table is sorted, an arrow symbol will appear close to the title, so you know on which column it was sorted last. As the sorting is done on the client machine, it may take a while with very large tables.

If the call is ongoing and you have the special grants to do so, a red scissor icon might appear next to the call status to allow for brute-force call closure. See the section the section called "Closing ongoing calls" [357] for further details.

Detail of IVR calls
IVR calls, that is calls that were handled without being queued, can be displayed through the pop-up linked to the the section called "OD04 - IVR details (paged)" [76] data block. The contents are very similar to the information displayed for taken or lost calls, though the only information currently tracked are call IVR events, DID and caller-id.

Chapter 3. Report Details

Reports can be fully configured by deciding which of the following blocks shall be included in each - see Configuring reports the section called "Configuring reports" [239]. The default report already includes all common blocks.

Historical reports - Answered calls

OK01 - All calls

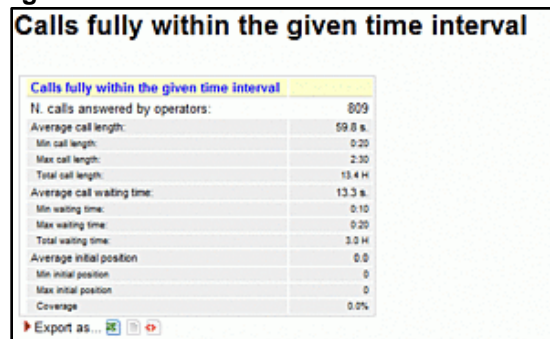The answered calls section deals with calls that were correctly handled by agents. The top panel shows:

- How many calls were handled;
- The average call length (i.e. time the caller spends talking to an operator);
- The maximum and minimum call lengths recorded for the given time period;
- The total call length (for all calls on all operators);
- The average call waiting time (i.e. the time a caller was waiting on a queue before being connected to an operator).
- The minimum and maximum call waiting times on record
- The total waiting time for all handled calls.
- The average initial position of the call in the queue
- The minimum and maximum initial queue positions that have been detected
- The queue position coverage: as this information is not tracked for all calls, this ratio shows the average number of calls that had queue position record.

| Available since	1.6.0
Default page	Answered calls
Shortcut code	OK01
API block name	OkDO.RiassAllCalls
Parameters	-
See also	

OK02 - Calls fully within the given time interval

The answered completed calls section deals with calls that were correctly handled by agents. This is similar to what's reported on previous panel but may exclude calls that were started before or terminated after the given time frame.

Available since	1.6.0
Default page	Answered calls
Shortcut code	OK02
API block name	OkDO.RiassFullyWithin
Parameters	-
See also	OK01 - All calls

OK03 - Agents on queue
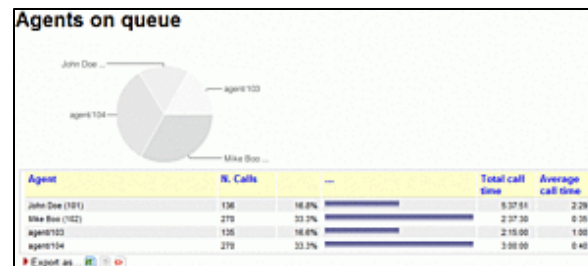
This report shows which agents have been available for the given queue, how many calls each one handled and the percentage of all calls that each one handled.

If calls are connected directly to a phone terminal, QM tries its best to show the corresponding terminal, usually in the format used by Asterisk, like "SIP/303" to signify a SIP phone whose number is 303.

If you connect to H.323 telephones via the OH323 module, the recorded channel names have no meaning and do not refer to a specific terminal; that's why all OH323 calls are grouped together under the label "OH323/-".

The pie graph shows which agents have been available for the given queue representing the percentage of all calls that each one handled.

Available since	1.6.0
Default page	Answered calls
Shortcut code	OK03
API block name	OkDO.AgentsOnQueue
Parameters	-
See also	

OK04 - Service level agreement
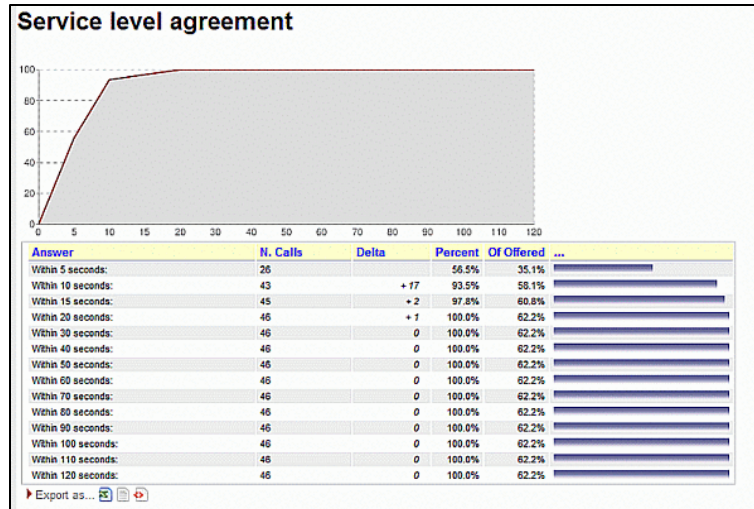
Service level agreement

Answer	N. Calls	Delta	Percent	Of Offered	...
Within 5 seconds:	26		56.5%	35.1%	
Within 10 seconds:	43	+ 17	93.5%	58.1%	
Within 15 seconds:	45	+ 2	97.8%	60.8%	
Within 20 seconds:	46	+ 1	100.0%	62.2%	
Within 30 seconds:	46	0	100.0%	62.2%	
Within 40 seconds:	46	0	100.0%	62.2%	
Within 50 seconds:	46	0	100.0%	62.2%	
Within 60 seconds:	46	0	100.0%	62.2%	
Within 70 seconds:	46	0	100.0%	62.2%	
Within 80 seconds:	46	0	100.0%	62.2%	
Within 90 seconds:	46	0	100.0%	62.2%	
Within 100 seconds:	46	0	100.0%	62.2%	
Within 110 seconds:	46	0	100.0%	62.2%	
Within 120 seconds:	46	0	100.0%	62.2%	

▶ Export as...

This report shows the distribution of call waiting times. It shows how many calls were answered within a given time frame, usually 120 seconds in 10 second increments (the time frame and increment can be modified by the administrator, if needed - see below).

You get a percentage of how many calls were answered within X seconds; the percentage includes calls answered in a shorter time frame and therefore grows with time.

The "delta" value you see is the absolute increment, expressed in number of calls, between each time frame, while the "Offered" column displays the result of the taken calls divided by the total taken plus the total lost.

This metric is computed only on answered calls, i.e. ignoring lost calls . If your SLA is defined in terms of taken and lost calls, see the corresponding metrics "Inclusive SLA" on the section called "Inclusive Service Level Agreement" [36].

The graph reports the percentage of how many calls were answered within X seconds, as reported in the table.

Since 1.6.2, it is possible to configure the time frame and increment separately for an initial period and the rest of the interesting period; in this way it is possible to have different breakdowns, e.g. every 5 seconds up to 20 seconds and every 10 seconds up to 120 seconds. See the section called "Configuring system preferences" [257].

For example, by setting:
- initial_interval=5 and max_initial_interval=20
- interval=10 and max_monitored_delay=60

You get the following cutoff points: 5, 10, 15, 20, 30, 40, 50, 60 seconds
By setting:
- initial_interval=3 and max_initial_interval=3
- interval=5 and max_monitored_delay=60

You get the following cutoff points: 3, 5, 10, 15, 20, 25,… seconds By setting
- initial_interval=0 and max_initial_interval=0
- interval=10 and max_monitored_delay=120

You get the default cutoff points: 10, 20, 30, 40, 50 , 60, 70, 80, 90, 100, 110 and 120 seconds

| Available since	1.6.0
Default page	Answered calls
Shortcut code	OK04
API block name	OkDO.ServiceLevelAgreement
Parameters	-
See also	UN18 UN07 UN06

OK05 - Disconnection causes
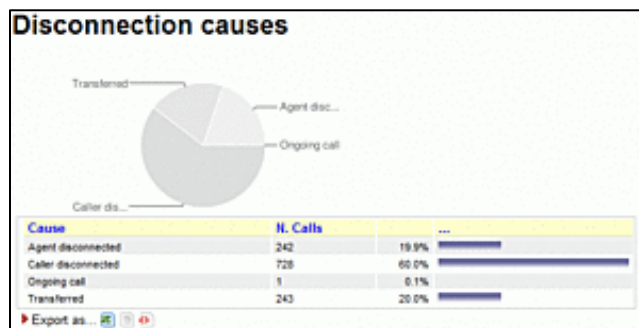
This report shows the reason why calls were terminated; this means that:
- The agent hung up, or
- The caller hung up, or
- The call was transferred outside the queue and the agent was freed again, or
- The call was ongoing at the time the report was run.

The graph reports the percentage values associated to the reason of why calls were terminated, as calculated in the table.

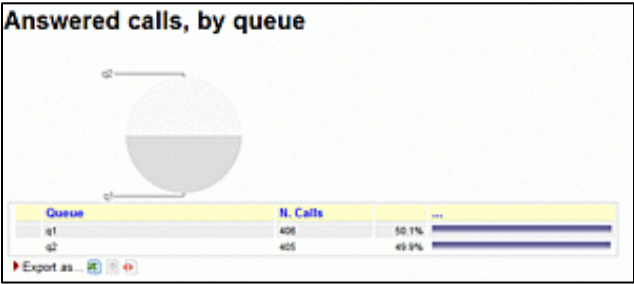| Available since	1.6.0
Default page	Answered calls
Shortcut code	OK05
API block name	OkDO.DisconnectionCauses
Parameters	-
See also	

OK06 - Transfers
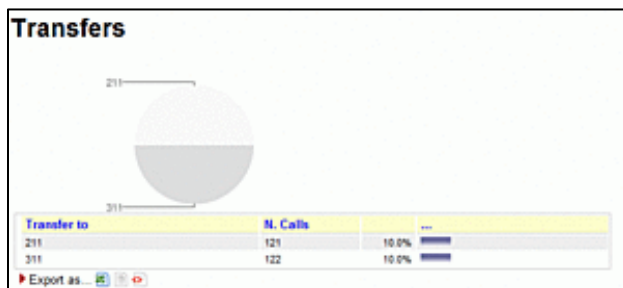
This graph shows how many calls were transferred to each extension in the given time frame. This lets you know who is handling exception calls.

When a call is transferred outside the queue system, its length is no more recorded by the queue subsystem; therefore you only get to see the length of the call while the agent was on line.

Available since	1.6.0
Default page	Answered calls
Shortcut code	OK06
API block name	OkDO.Transfers
Parameters	-
See also	

OK07 - Answered calls, by queue

If more than one queue is in use for the report, this table shows the relative magnitude of each queue. The graph reports the percentage associated to each queue in the table.

Available since	1.6.0
Default page	Answered calls
Shortcut code	OK07
API block name	OkDO.AnsweredcallsByQueue
Parameters	-
See also	

OK08 - Answered calls, by direction

If more than one queue is in use for the report, this table shows the relative magnitude of each queue.

Inbound queues are marked with the symbol [symbol] while outbound queuesuse the symbol [symbol] .
The graph reports the percentage associated to each queue in the table.

Available since	1.6.0
Default page	Answered calls
Shortcut code	OK08
API block name	OkDO.AnsweredcallsByDirection
Parameters	-
See also	

OK09 - Answered calls, by stints

This graph counts the distribution of multi-stint calls on selected queues. If multi-stint mode is not enable, all calls will have only one stint.

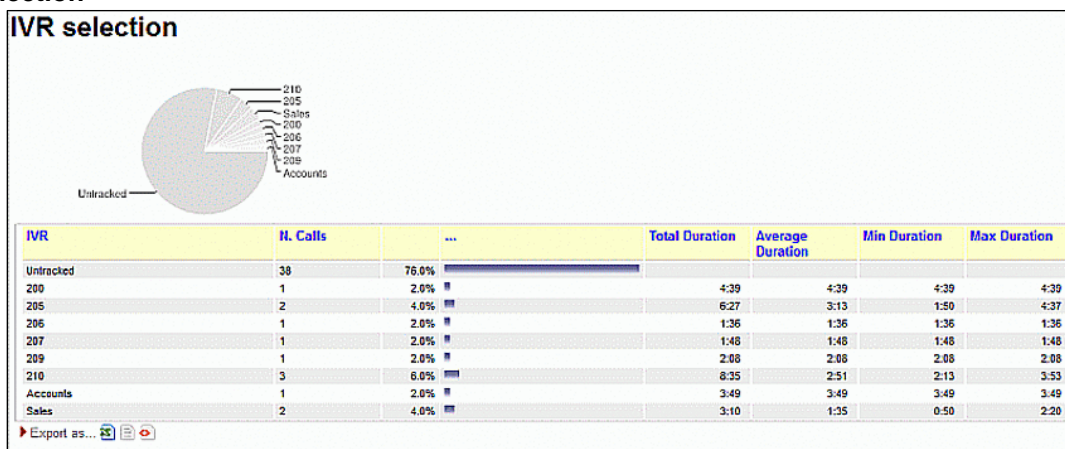| Available since	1.6.0
Default page	Answered calls
Shortcut code	OK09
API block name	OkDO.StintsOk
Parameters	-
See also	

OK10 - Queue position

This graph shows the initial queue position that the calls had when they joined the queue. For example, a queue position of 1 means that a call was first in line, of 5 means that a call had four other calls in line before being answered. As the logging of queue positions is a bit inconsistent, some calls might be missing it ("Untracked")

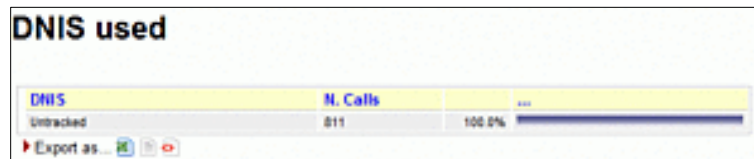| Available since	1.6.0
Default page	Answered calls
Shortcut code	OK10
API block name	OkDO.QPosOk
Parameters	-
See also	

OK11 - IVR selection

This graph shows the distribution of IVR selections available in the calls processed. We can also see the IVR duration values, which is related to the time that the call was within the IVR before entering the queue.

This must be tracked manually in Asterisk - See "Configuring Asterisk for QueueMetrics" below.

| Available since	1.6.0
Default page	Answered calls
Shortcut code	OK11
API block name	OkDO.IvrOk
Parameters	-
See also	

OK12 - DNIS used

This graph shows the distribution of DNIS lines available in the calls processed.

This must be tracked manually in Asterisk - See "Configuring Asterisk for QueueMetrics" below.

Available since	1.6.0
Default page	Answered calls
Shortcut code	OK12
API block name	OkDO.DnisOk
Parameters	-
See also	

OK13 - Music On Hold by Agent
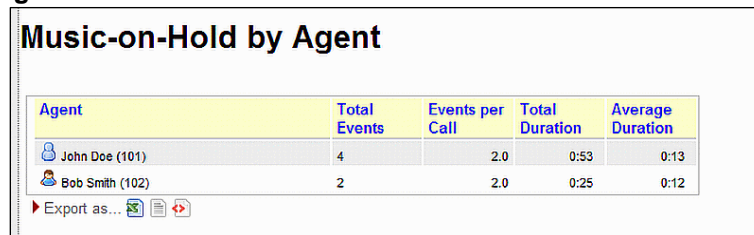
This allows you to see the total number of Music on Hold (MOH) events per agent, how many MOH instances took place throughout a call, the average and total duration of MOH events.

This must be tracked manually in Asterisk - See "Configuring Asterisk for QueueMetrics" below.

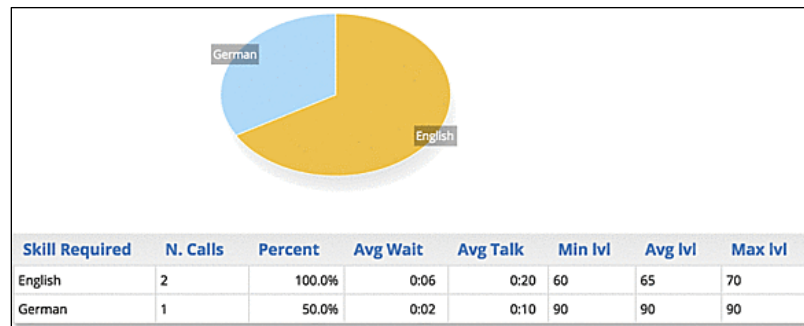|	
Available since	12.2.0
Default page	Answered calls
Shortcut code	OK13
API block name	OkDO.MOHOk
Parameters	-
See also	

OK14 - Report Header

This is the report header and will be added automatically on each tab on the screen and at the beginning of each session in the PDF and Excel report export.

Available since	13.04.9
Default page	Beginning of each page
Shortcut code	OK14
API block name	OkDO.HDRRpt
Parameters	-
See also	HDRRpt

OK15 - Required Skills for Taken Calls

Skill Required	N. Calls	Percent	Avg Wait	Avg Talk	Min lvl	Avg lvl	Max lvl
English	2	100.0%	0:06	0:20	60	65	70
German	1	50.0%	0:02	0:10	90	90	90

This data block displays the skills that were required for taken calls in the current reports.

As multiple skills may be required for the same call, or some calls may have no required skill, the sum of all skills may be significantly different from the number of calls.

For each skill, the average wait and talk duration of all calls having that specific skill is computed. Plus, the minimum, average and maximum level at which this skill was requested for the specific set of calls is displayed.

Available since	15.10
Default page	Answered calls
Shortcut code	OK15
API block name	OkDO.SkillsOk
Parameters	-
See also	UN19, UN20

Historical reports - Details of answered calls

OD01 - Queue details

This page shows the detail of answered calls. See Chapter 5, *Showing call details* [59].

When running a report, it is advisable to use block the section called "OD03 - Queue details (paged)" [75] instead.

Available since	1.6.0
Default page	Details of answered calls
Shortcut code	OD01
API block name	DetailsDO.CallsOK
Parameters	-
See also	the section called "OD03 - Queue details (paged)" [75]

OD02 - Add to export job

This is a pseudo-block that is used to display a button for call export. The button may not be displayed if the user does not have the correct grants. As it does not actually contain data, it cannot be queried over XML-RPC.

Available since	1.7.0
Default page	Details of answered calls
Shortcut code	OD02
API block name	-
Parameters	-
See also	

OD03 - Queue details (paged)

This page shows the paged detail of answered calls. See Chapter 5, *Showing call details* [59].

The details of calls are paged - so you can move back and forth - and you can control which columns will appear on screen. The paged mode uses way less memory and is quicker to use than the full list as in the section called "OD01 - Queue details" [74].

When accessing the data source though XML-RPC or printing a report, it is mandatory to use the OD01 block, as this block is interactive and will not render correctly.

Available since	13.03
Default page	Details of answered calls
Shortcut code	OD03
API block name	-
Parameters	-
See also	the section called "OD01 - Queue details" [74] and the section called "Configuring paged call lists" [247]

OD04 - IVR details (paged)

This block shows the paged detail of IVR calls. These are calls that did not hit a queue in the current report and thererfore are reported as neither answered nor unanswered.

The details of calls are paged - so you can move back and forth - and you can control which columns will appear on screen. The paged mode uses way less memory and is quicker to use than the full list as in the section called "OD05 - IVR details (full list)" [76].

When accessing the data source though XML-RPC or printing a report, it is mandatory to use the OD05 block, as this block is interactive and will not render correctly.

Available since	13.03
Default page	Details of answered calls
Shortcut code	OD04
API block name	-
Parameters	-
See also	the section called "OD05 - IVR details (full list)" [76]

OD05 - IVR details (full list)
This block shows the paged detail of IVR calls. These are calls that did not hit a queue in the current report and thererfore are reported as neither answered nor unanswered.

This block prints a complete list and should be used only for printing and XML-RPC querying.

Available since	13.03
Default page	Details of answered calls
Shortcut code	OD05
API block name	DetailsDO.CallsIVR
Parameters	-
See also	the section called "OD04 - IVR details (paged)" [76]

OD06 - Executive Summary

This report provides a wealth of information regarding the situation in general. With a final focus on lost calls and disconnection reasons. The information is the following:

General Statistics
- Taken Calls
- Avg. Elapsed Time
- Lost Calls
- Total Calls
- Total Elapsed Time Caller Statistics
- Short Calls
- Unique Callers
- Unique First Taken
- Unique First Lost
- Unique Final Lost

Lost Calls Statistics A list of disconnection reasons and the number of lost calls with each disconnection reason
Notes:

- The number of seconds needed for a call to be classified as long is dictated by the following configuration property: default.shortcall_treshold = 30 If the property is not specified, it will deafault at 30.

Available since	17.06.16
Default page	Executive Summary
Shortcut code	OD06
API block name	DetailsDO.ExecutiveSummary
Parameters	-
See also	the section called "OD06 - Executive Summary" [77]

Historical reports - Unanswered calls

Unanswered calls are calls that were lost, i.e. the caller could not connect to an agent. This usually means that either the caller hung up, fed up with waiting, or the queue system decided to discharge the caller, maybe sending him to voicemail or another queue.

If you run a report with an agent filter, or a supervisor filter, or a location filter, the number of lost calls in the report is usually zero, showing just outgoing calls, if any. This is because the agent is specified only for taken calls and not lost ones, so not deleting them all would show, e.g., the taken call data for one single agent and the lost calls for all of the queue. You can override this behaviour through a configuration switch if you feel this is not correct for you.

UN01 - All calls

The report shows:
- How many calls were lost;
- The average waiting time before disconnection;
- The average queue position at disconnection (i.e. how many calls the queue had to dispatch before connecting the caller to an operator).
- The minimum and maximum wait times
- The minimum and maximum queue position at disconnect.
- The average, minimum and maximum initial queue position, and the coverage given for this computation

Available since	1.6.0
Default page	Unans.
Shortcut code	UN01
API block name	KoDO.ReportKoAll
Parameters	-
See also	

UN02 - Calls fully within the given time interval

The unanswered completed calls section deals with calls that were lost. This is similar to what's reported on previous panel but may exclude calls that were started before or terminated after the given time frame.

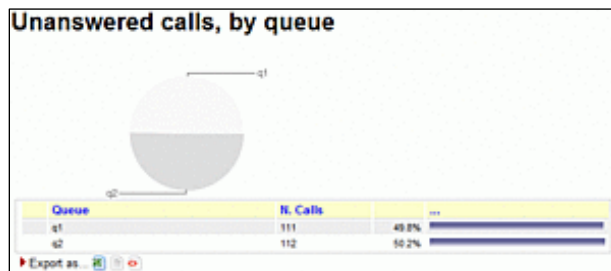|	
Available since	1.6.0
Default page	Unans.
Shortcut code	UN02
API block name	KoDO.ReportKoFully
Parameters	-
See also	

UN03 - Disconnection causes
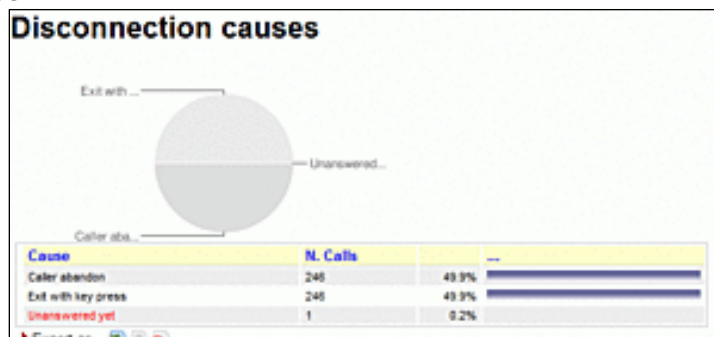
This report shows the relative magnitude of disconnection causes that are:
- The caller hung up, or
- The queue timed out and discharged the caller (if this feature is enabled by the queue configuration - see Chapter 18, Configuring Asterisk for QueueMetrics [345]), or
- The caller exited the queue by pressing a key (if this feature is enabled by the queue configuration).

|	
Available since	1.6.0
Default page	Unans.
Shortcut code	UN03
API block name	KoDO.DiscCauses
Parameters	-
See also	

UN04 - Unanswered calls, by queue

If more than one queue is in use for the report, this graph shows the relative magnitude of each queue, either in a numerical than in a graphical form.

| Available since	1.6.0
Default page	Unans.
Shortcut code	UN04
API block name	KoDO.UnansByQueue
Parameters	-
See also	

UN05 - Unanswered outbound calls, by agent
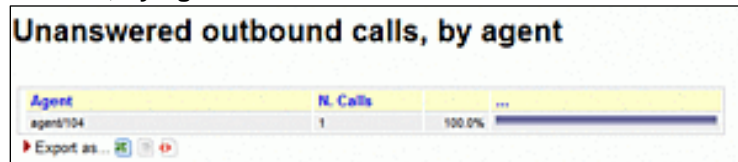
This graph shows the relative magnitude of unanswered outbound calls, grouped by agent.

Available since	1.6.0
Default page	Unans.
Shortcut code	UN05
API block name	KoDO.OutboundKo
Parameters	-
See also	

UN06 - Unanswered calls - distribution by length
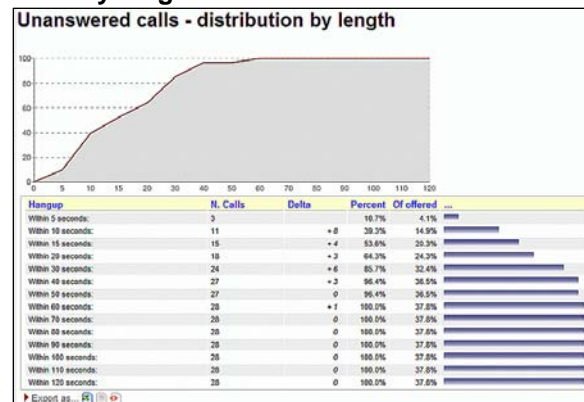
This report is functionally equivalent to "Service level agreement" in the Answered calls section (see the section called "Service level agreement" [32]), but is computed on lost calls. It shows how many calls were hung up within a given time frame, usually 120 seconds in 10 second increments (the time frame and increment can be modified by the administrator, if needed - see the section called "OK04 - Service level agreement" [68]).

You get a percentage of how many calls were lost within X seconds; the percentage includes calls lost in a shorter time frame and therefore grows with time.

The "delta" value you see is the absolute increment, expressed in number of calls, between each time frame, while the "Offered" column displays the result of the taken calls divided by the total taken plus the total lost.

The graph reports the percentage of how many calls were not answered within X seconds, as reported in the table.

Available since	1.6.0
Default page	Unans.
Shortcut code	UN06
API block name	KoDO.UnansByLen
Parameters	-
See also	UN07 UN18 OK04

UN07 - Inclusive SLA (computed on both answered and unanswered calls)
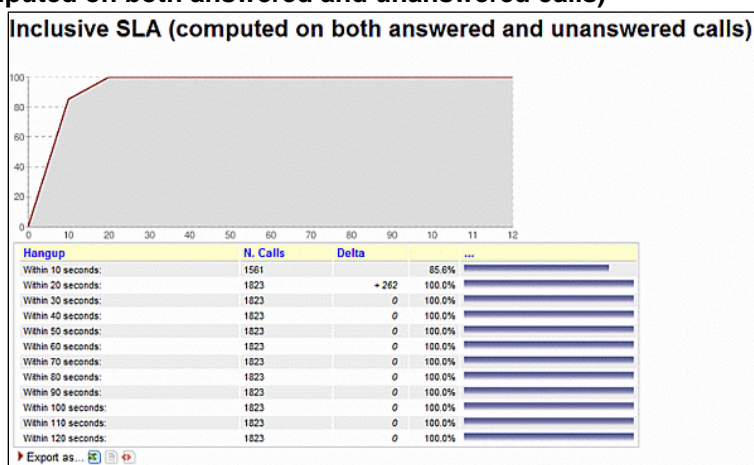
The inclusive SLA corresponds to the Service Level Agreement metrics shown on the section called "Service level agreement" [32], with the difference that it is computed taking into consideration both answered and unanswered calls.

The difference between UN07 and UN18 is that the number of calls in UN18 is only the number of taken calls within the given answer period, while in UN07 it is the total number of taken and lost calls within the time period.

The graph reports the same information found in the table, but in a graphical way.

The time frame and increment can be modified by the administrator, if needed - see the section called "OK04 - Service level agreement" [68].

Available since	1.6.0
Default page	Unans.
Shortcut code	UN07
API block name	KoDO.InclusiveSLA
Parameters	-
See also	OK04 UN06 UN18 (Inclusive Answered SLA)

UN08 - Unanswered calls by key press
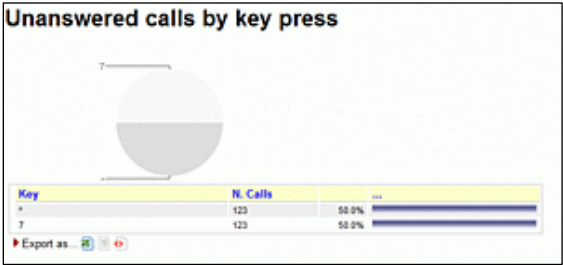
If there are any calls that are were set unanswered because the caller pressed a key to exit the queue, this graph shows which keys were pressed and how many calls were terminated for that reason.

Available since	1.6.0
Default page	Unans.
Shortcut code	UN08
API block name	KoDO.ReportKoKeyPress
Parameters	-
See also	

UN09 - Unanswered calls, by stints

This graph tells the stint distribution of unanswered calls. It corresponds to the graph called "Answered calls, by stints".

|	
Available since	1.6.0
Default page	Unans.
Shortcut code	UN09
API block name	KoDO.StintsKo
Parameters	-
See also	

UN10 - All calls, by stints
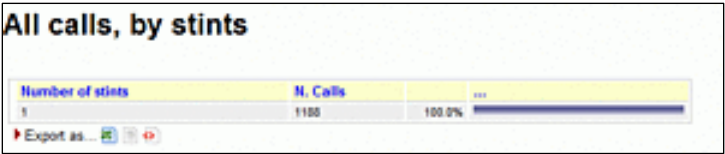
This graph tells the stint distribution of all processed calls. It corresponds to the sum of the graphs called "Answered calls, by stints" and "Unanswered calls, by stints"

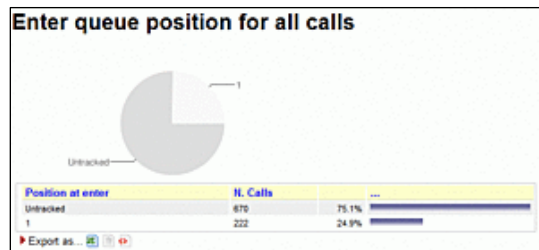|	
Available since	1.6.0
Default page	Unans.
Shortcut code	UN10
API block name	KoDO.StintsOkKo
Parameters	-
See also	

UN11 - Enter queue position
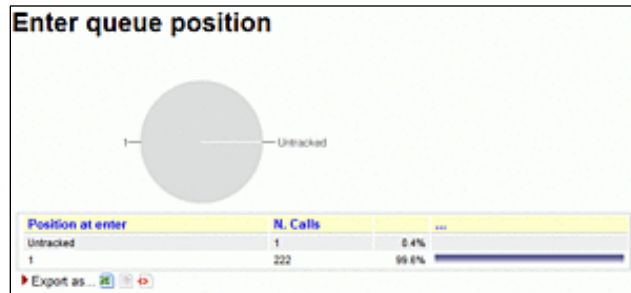
This graph shows the initial queue position that the calls had when they joined the queue. For example, a queue position of 1 means that a call was first in line, of 5 means that a call had four other calls in line before being answered. As the logging of queue positions is a bit inconsistent, some calls might be missing it ("Untracked")

Available since	1.6.0
Default page	Unans.
Shortcut code	UN11
API block name	KoDO.QPosKo
Parameters	-
See also	

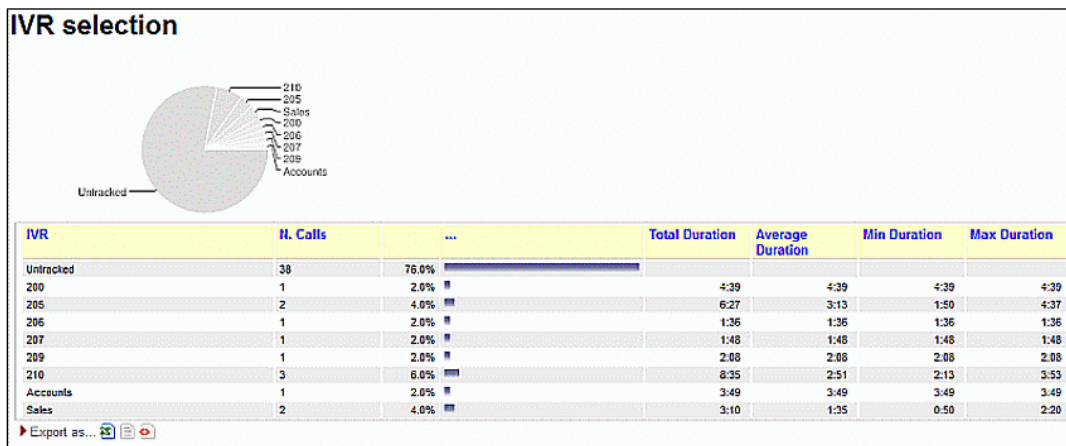**UN12 - Enter queue position for all calls**

This graph shows the initial queue positions for both answered and unanswered calls.

Available since	1.6.0
Default page	Unans.
Shortcut code	UN12
API block name	KoDO.QPosOkKo
Parameters	-
See also	

UN13 - IVR selection

This graph shows the distribution of IVR selections available for lost calls.

We can also see the IVR duration values, which are related to the time that the call was within the IVR before entering the queue.

This must be tracked manually in Asterisk - See "Configuring Asterisk for QueueMetrics" below.

Available since	1.6.0
Default page	Unans.
Shortcut code	UN13
API block name	KoDO.IvrKo
Parameters	-
See also	

UN14 - IVR selection, for all calls

This graph shows the distribution of IVR selections for all calls (taken and lost) available in the report.

Available since	1.6.0
Default page	Unans.
Shortcut code	UN14
API block name	KoDO.IvrOkKo
Parameters	-

UN15 - DNIS used
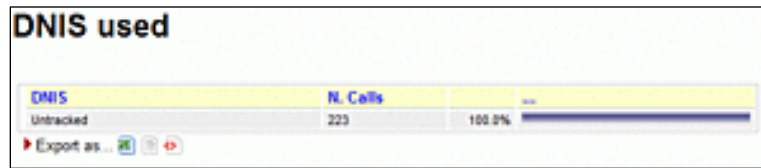
This graph shows the distribution of DNIS lines that lead to lost calls.

This must be tracked manually in Asterisk - See "Configuring Asterisk for QueueMetrics" below.

Available since	1.6.0
Default page	Unans.
Shortcut code	UN15
API block name	KoDO.DnisKo
Parameters	-
See also	

UN16 - DNIS used, for all calls

This graph shows the distribution of DNIS lines for all calls (taken and lost) available in the report.

Available since	1.6.0
Default page	Unans.
Shortcut code	UN16
API block name	KoDO.DnisOkKo
Parameters	-

UN17 - Call Overview

The answered calls section deals with calls that were correctly handled by agents.

For each queue included in the report, the following data will be shown:

- Calls offered: total number of calls in the period to hit that queue
- Calls answered: total number of answered calls
- Lost calls: total number of calls that have not been answered
- Average call length: it computed only on the talk time of answered calls, expressed as MM:SS
- Total call length: the total cumulate speak time for each queue, expressed as decimalized hours
- Average wait Q: the average wait time for both answered and lost calls

- Total wait Q: total cumulate wait time for answered and unanswered calls, expressed as decimalized hours

In the first line, there is a "Total" line, that sums up the values shown in previous lines. The following lines are sorted according to the number of offered calls.

Available since	1.6.1.1
Default page	Unanswered calls
Shortcut code	UN17
API block name	KoDO.OverviewOkKo
Parameters	-
See also	

UN18 - Inclusive Answered SLA
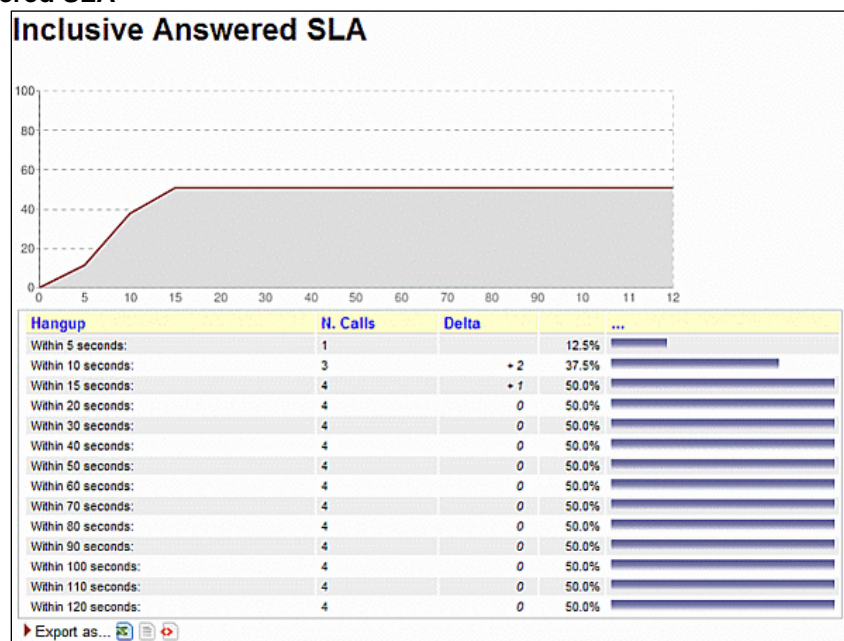
The inclusive SLA corresponds to the Service Level Agreement metrics shown on the section called "Service level agreement" [32], with the difference that it is computed taking into consideration both answered and unanswered calls.
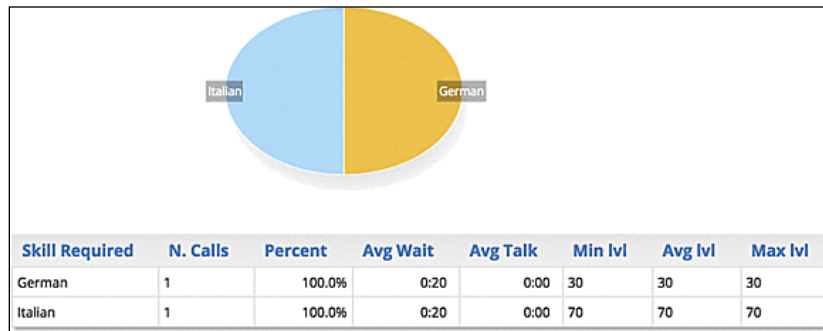
The difference between UN07 and UN18 is that the number of calls in UN18 is only the number of taken calls within the given answer period, while in UN07 it is the total number of taken and lost calls within the time period.

The graph reports the same information found in the table, but in a way that is easier to read.

The time frame and increment can be modified by the administrator, if needed - see the section called "OK04 - Service level agreement" [68].

Available since	1.6.1.2
Default page	Unans.
Shortcut code	UN18
API block name	KoDO.InclusiveAnswSLA
Parameters	-
See also	UN07 (Inclusive SLA) UN06 OK04

UN19 - Required Skills for Lost Calls

Skill Required	N. Calls	Percent	Avg Wait	Avg Talk	Min lvl	Avg lvl	Max lvl
German	1	100.0%	0:20	0:00	30	30	30
Italian	1	100.0%	0:20	0:00	70	70	70

This data block displays the skills that were required for lost calls in the current reports.

As multiple skills may be required for the same call, or some calls may have no required skill, the sum of all skills may be significantly different from the number of calls.

For each skill, the average wait duration of all calls having that specific skill is computed. As lost calls never have any talk time, it always appears as 0:00 and is displayed for symmetry with blocks OK15 and UN20.

The minimum, average and maximum level at which this skill was requested for the specific set of calls is displayed.

Available since	15.10
Default page	Unans.
Shortcut code	UN19
API block name	KoDO.SkillsKo
Parameters	-
See also	UN20, OK15

UN20 - Required Skills for All Calls

Skill Required	N. Calls	Percent	Avg Wait	Avg Talk	Min lvl	Avg lvl	Max lvl
English	2	66.7%	0:06	0:20	60	65	70
German	2	66.7%	0:11	0:05	30	60	90
Italian	1	33.3%	0:20	0:00	70	70	70
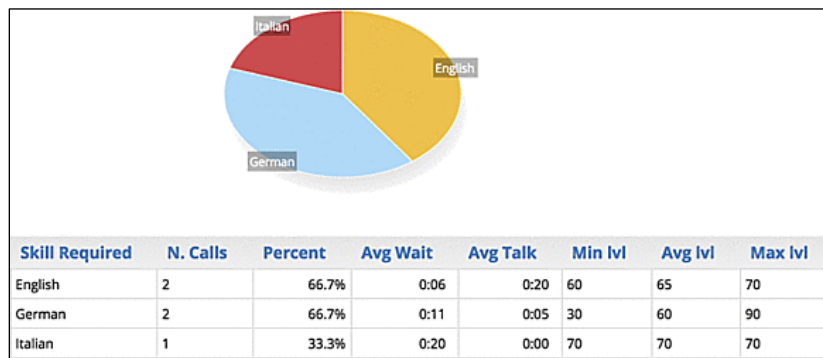
This data block displays the skills that were required for all calls in the current reports.

As multiple skills may be required for the same call, or some calls may have no required skill, the sum of all skills may be significantly different from the number of calls.

For each skill, the average wait and talk duration of all calls having that specific skill is computed. Plus, the minimum, average and maximum level at which this skill was requested for the specific set of calls is displayed.

Available since	15.10
Default page	Unans.
Shortcut code	UN20
API block name	KoDO.SkillsOkKo
Parameters	-
See also	UN19, OK15

Historical reports - Details of unanswered calls

UD01 - Detail of unanswered calls

This page shows full details of unanswered calls the section called "Detail of unanswered calls" [63].

In general this data block should not be used in reports but block the section called "UD02 - Lost calls (paged)" [90] should be preferred.

Available since	1.6.0
Default page	Details of unanswered calls
Shortcut code	UD01
API block name	DetailsDO.CallsKO
Parameters	-
See also	the section called "UD02 - Lost calls (paged)" [90]

UD02 - Lost calls (paged)

This page shows the paged detail of unanswered calls. See Chapter 5, *Showing call details* [59].

The details of calls are paged - so you can move back and forth - and you can control which columns will appear on

screen. The paged mode uses way less memory and is quicker to use than the full list as in the section called "UD01 - Detail of unanswered calls" [89].

When accessing the data source though XML-RPC or printing a report, it is mandatory to use the UD01 block, as this block is interactive and will not render correctly.

Available since	13.03
Default page	Details of unanswered calls
Shortcut code	UD02
API block name	-
Parameters	-
See also	the section called "UD01 - Detail of unanswered calls" [89] and the section called "Configuring paged call lists" [247]

Historical reports - Area code analysis

AC01 - Area code report

If the Caller*ID is present, it is possible to break down both answered and unanswered calls to specific area codes by clicking on the "Area code analysis" button.

By selecting a number of caller id digits to search upon and a starting digit position, you get a number of statistics grouped by area codes.

This report gives an immediate check of the geographical origin of calls handled by your call center.

Available since	1.6.0
Default page	Area code analysis
Shortcut code	AC01
API block name	AreaAnDO.Setup
Parameters	-
See also	

AC02 - Detail for answered calls

This report shows the answered calls grouped following the rules defined in AC01 - Area code report.

Available since	1.6.0
Default page	Area code analysis
Shortcut code	AC02
API block name	AreaAnDO.CallsOK
Parameters	-
See also	AC01 - Area code report

AC03 - Detail for unanswered calls

This report shows the unanswered calls grouped following the rules defined in AC01 - Area code report.

Available since	1.6.0
Default page	Area code analysis
Shortcut code	AC03
API block name	AreaAnDO.CallsKO
Parameters	-
See also	AC01 - Area code report

AC04 - Frequent Area Codes

This report shows detailed information regarding different Area Codes. We take into account the quantity and quality of calls related to different areas, by calculating the following information:
- Prefix
- N. of Callers
- Totale Calls
- Taken Calls
- % of Taken Calls
- Lost Calls

- Total Duration
- Avg. Duration

Available since	17.06.16
Default page	Area code analysis
Shortcut code	AC04
API block name	AreaAnDO.FrequentAreaCodes
Parameters	-
See also	AC01 - Area code report

Historical reports - Distributions

When running an inbound call center, it is very important to determine the reason why a call is delayed: are your clients refusing to answer? Did they forget to log off before leaving their workplace? The inbound ACD call attempts metrics try to answer to these questions.

As these metrics are not usually recorded by Asterisk, you'll have to patch and recompile your Asterisk system in order to gather them - see the section the section called "Enabling ACD call attempts recording on Asterisk 1.0 and 1.2" [349]. If you do not do so, the metrics presented here will always appear zeroed out. With Asterisk 1.4, this feature should be automatically enabled with no need to patch the source code.

AT01 - Inbound ACD call attempts

This page shows the following pieces of information:
- How many agent attempts were made, i.e. how many times the agent's telephones were rung in total
- The average number of attempts that were necessary for a taken call; the minimum, maximum and total attempts made that resulted in a taken call
- The average number of attempts that were necessary for a lost call; the minimum, maximum and total attempts made that resulted in a lost call

Available since	1.6.0
Default page	Distrib.
Shortcut code	AT01
API block name	DistrDO.ReportAcd
Parameters	-
See also	

AT02 - ACD attempts by queue

The following metrics are extracted and broken down by queue:
- N. of lost agent attempts (i.e. the agent was called but not responding)
- The average ring time for lost attempts
- The total ringing time for lost calls
- The number of taken agent attempts (i.e. calls answered)
- The average ring duration for taken calls
- The total ring time for taken calls

Available since	1.6.0
Default page	Distrib.
Shortcut code	AT02
API block name	DistrDO.AcdByQueue
Parameters	-
See also	

AT03 - ACD attempts by terminal

This graph breaks down agent attempts by the agent that was called. The following pieces of information are extracted for each agent:
- N. of lost agent attempts (i.e. the agent was called but not responding)
- The average ring time for lost attempts
- The total ringing time for lost calls
- The number of taken agent attempts (i.e. calls answered)
- The average ring duration for taken calls
- The total ring time for taken calls

Available since	1.6.0
Default page	Distrib.
Shortcut code	AT03
API block name	DistrDO.AcdByTerminals
Parameters	-

## Historical reports - Call distribution by day

The call distribution report shows when calls were handled, when calls were lost and the average wait times broken down by period.

All percentages are calculated on the call class they belong to, i.e. a 50% of "Unanswered calls" on one day means that 50% of all unanswered calls for the period happened during that day, not that 50% of calls were lost.
For each metrics, the total number of calls is shown, together with average, minimum and maximum times. Graphs are plotted on the total number of calls broken down and on the averages.

It is possible to change the interval in the Hourly graphs, so that you can have reports break down calls
e.g. by half-hours or hour quarters, by changing a value in the QueueMetrics master configuration file.

DD01 - Answered call distribution per day
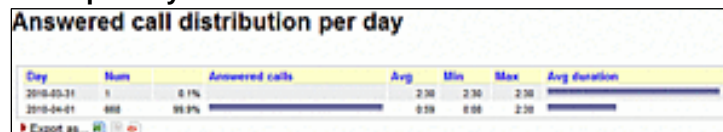
Taken calls are shown per specific day. Days with no events are not shown.

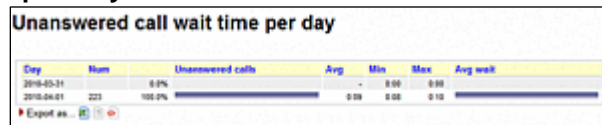|	
Available since	1.6.0
Default page	Call distribution, by day
Shortcut code	DD01
API block name	CallDistrDO.AnsDistrPerDay
Parameters	-

DD02 - Answered call wait time per day
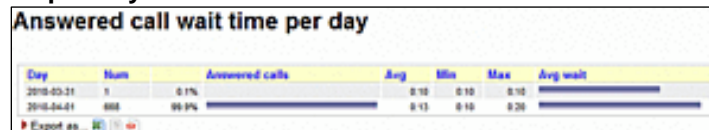
The total numbers of call wait time for answered calls are plotted for each day.

Available since	1.6.0
Default page	Call distribution, by day
Shortcut code	DD02
API block name	CallDistrDO.AnsWaitPerDay
Parameters	-
See also	

DD03 - Unanswered call wait time per day

The total numbers of lost wait time for lost calls are plotted for each day.

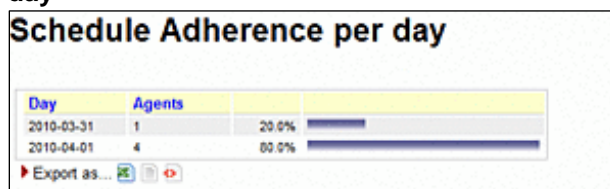| Available since	1.6.0
Default page	Call distribution, by day
Shortcut code	DD03
API block name	CallDistrDO.UnansWaitPerDay
Parameters	-

DD04 - Sales per day

Sales and contacts are shown on a daily basis.

Available since	1.6.0
Default page	Call distribution, by day
Shortcut code	DD04
API block name	CallDistrDO.SalesPerDay
Parameters	-

DD05 - Schedule Adherence per day

The Schedule Adherence report shows the number of distinct agents that were detected during the given period. This makes it possible to detect the number of different people that had been working on a given moment.

Available since	1.6.0
Default page	Call distribution, by day
Shortcut code	DD05
API block name	CallDistrDO.StaffPerDay
Parameters	-

DD06 - Queue length per day

The Queue Length report shows the average length of the queue for each period, giving minimums and maximums. The Steps computation shows how fast each queue progresses during the period, expresses in steps per hour. Note: giving a meaning to the Queue Length reports may be hard in the case of composite queues.

Available since	1.6.0
Default page	Call distribution, by day
Shortcut code	DD06
API block name	CallDistrDO.QPosPerDay
Parameters	-

DD07 - Inclusive SLA per day

The inclusive SLA corresponds to the Service Level Agreement metrics shown on the section called "Service level agreement" [32], grouped by day.

Available since	1.6.0
Default page	Call distribution, by day
Shortcut code	DD07
API block name	CallDistrDO.InclSlaPerDay
Parameters	-

DD08 - Traffic Analysis by period - per day

This report shows aggregate inbound/outbound activity per day.
- Date: the date, hour or day of week used for the report.
- Avg agents: the average number of agents, as calculated by agents logged in for each period versus the total period. E.g. if an agent logs in at 10:00 and logs off at 11:30, and a second agent logs in at 10:15 and logs off at 11:00, the period for 10:00 to 10:30 will show 1.5 available agents.
- Avg calls/agent: number of INCOMING calls (answered+unanswered) per period versus average available agents
- Service level: the SLA (see below), computed on INCOMING calls only
- Unans: Lost calls (INCOMING only)
- Unans short: Lost calls below X seconds (INCOMING only, as a percentage of all calls)
- Avg out: number of OUTGOING calls (completed and lost) per period versus number of available agents
- Out/in: ratio of outbound to inbound
- Avg ans. Average answer time (for INCOMING only)
- Avg talk time: Average talk time (for INCOMING only)
- Max wait ans: Maximum answer time in period (for INCOMING only)
- Max wait lost: Maximum wait time for lost calls in period (for INCOMING only)
- Max duration: Maximum talk time in period (for INCOMING only)
- Max duration OUT: Maximum talk time in period (for OUTGOING only)

- N. offered: Total number of INCOMING calls
- N Answered: Total number of answered INCOMING calls
- N Answered out: Total number of answered OUTGOING calls
- Min Agents: minimum number of agents logged on for the period
- Max Agents: maximum number of agents logged on in the period

The Service Level is measured against a time period that is specidfied in the *default.secondsServiceLevel* parameter (default is 20, as to say "percentage of calls answered within 20 seconds").

Short calls are defined as being shorter than the *default.shortCallsLimit* parameter - default is 5 seconds.

Available since	1.6.0.4
Default page	Call distribution, by day
Shortcut code	DD08
API block name	CallDistrDO.TrafficAnPerDay
Parameters	-
See also	DH08, DW08

DD09 - Skill sets per day
This report shows the unique sets of skills during a specified time period.

Skills per Day

Day	Ans.	Unans.	Num	Within SLA: 10s.	Product1 english internet technical	Product1 english invoice land	Product1 english invoice mobile	Product1 english land others	Product1 english mobile others	Product1 english mobile technical
2016-12-04	0	0	0	-	0	0	0	0	0	0
2016-12-05	2,505	223	2,728	45.3%	0	2	8	7	4	8
2016-12-06	1,737	133	1,870	50.9%	0	0	4	2	4	5
2016-12-07	1,656	55	1,711	58.1%	1	0	3	0	4	3
2016-12-08	1,488	50	1,538	56.7%	1	2	5	0	2	1
2016-12-09	1,745	40	1,785	56.8%	3	0	9	0	6	0
2016-12-10	711	26	737	53.9%	0	0	0	0	4	0

The report is computed out of all calls in the current analysis. Time bands are set as for the current report (system property that can be overridden and is "remembered" across sessions; can be reset through the Custom Reports page). The SLA is set for a queue (or set of queues to be reported on at once) though the parameter "Service Level Agreement" on the Queue configuration page; if missing, the default value is taken from the property sla.max_initial_delay.
Each call is grouped into a "bucket" based on skills found; skills are sorted by **Skill Sort Weight** in order to create unique buckets for each permutation. Only buckets with at least one call are added to the report. Buckets are displayed with codes correctly decoded. To change the order of how the skills appear inside each bucket, make sure to adjust the **Skill Sort Weight** field in the **Agents Skill** editor. Skills with a lower sort weight are positioned before skills with higher sort weight. If two skills have the same sort weight, they are sorted alphabetically based on their description.

For example, a call with skills "inv en fix" is set to a bucket "en fix inv", that is a separate bucket from

> *e.g. "en fix" but is is the same as "en inv fix" . Such a call is displayed in a column named "English Fix Invoice", where the names are based on the descriptions used for skills, sorted by the skills' sort weight.*

The report shows the following columns:
- Time Slot: a call is banded in a time slot based on the time it is queued
- Answered calls: Total number of answered calls for this slot
- Unanswered calls: total number of lost calls for this slot
- Num: Total Answered + Unanswered

- Within SLA Xsec: the number of calls answered within X seconds divided the total number of calls offered (answered+lost)
- "bucket": the total number of calls (taken+lost) that belong to this bucket

As the report may be very large, if you export it to PDF it may be cut on the right side. XLS report works without problems instead.

Available since	16.10.11
Default page	Call distribution, by day
Shortcut code	DD09
API block name	CallDistrDO.SkillsPerDay
Parameters	-
See also	DH09, DW09

DD10 - Detail Summary
This reports shows the number of taken and lost call by day.

This report comes in three versions that show the number of calls in different time slices. The available period of times are:
- Hourly [DH10]
- Days of the week [DW10]
- Days (with date) [DD10]

The information is the following
- Taken Calls
- Lost Calls
- N. of Calls

Available since	17.06.16
Default page	Detail Summary, by day
Shortcut code	DD10
API block name	CallDistrDO.DetailSummaryDom
Parameters	-
See also	DH10, DW10

Historical reports - Call distribution by hour
Events are shown on a 24-hour distribution. If this graph appears to be incorrect, you have to run a "Custom report" setting the time zone accordingly (see the section called "Custom reports" [24]).
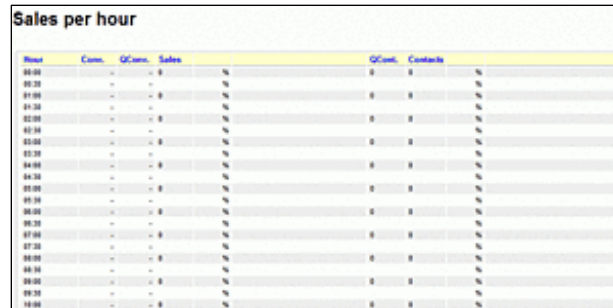
DH01 - Answered call distribution per hour

The total numbers of call lengths for answered calls are plotted for each hourly interval. The size of hourly intervals can be

controlled by the *default.hourly_slot* configuration property, making it possible to run this reports based on 30-minute, 20-minute or 15-minute intervals.

Available since	1.6.0
Default page	Call distribution, by hour
Shortcut code	DH01
API block name	CallDistrDO.AnsDistrPerHr
Parameters	-

DH02 - Answered call wait time per hour
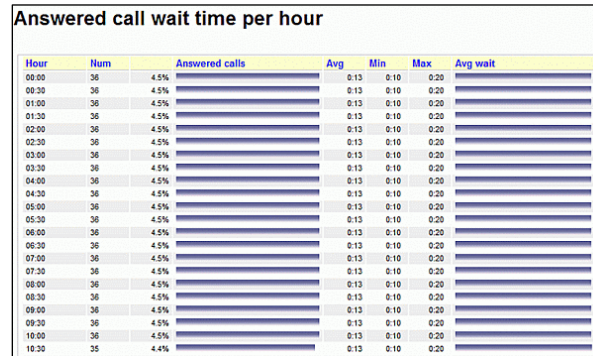
The total numbers of call wait time for answered calls are plotted for each hourly interval. The size of hourly intervals can be controlled by the *default.hourly_slot* configuration property, making it possible to run this reports based on 30-minute, 20-minute or 15-minute intervals.

Available since	1.6.0
Default page	Call distribution, by hour
Shortcut code	DH02
API block name	CallDistrDO.AnsWaitPerHr
Parameters	-

DH03 - Unanswered call wait time per hour
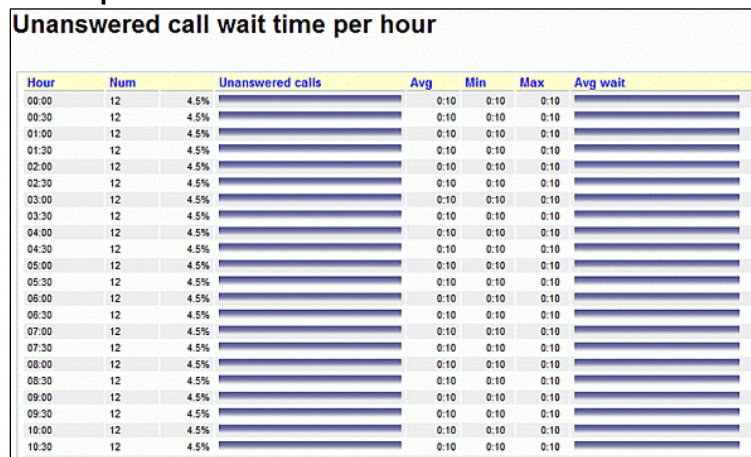
The total numbers of call lengths for unanswered calls are plotted for each hourly interval. The size of hourly intervals can be controlled by the *default.hourly_slot* configuration property, making it possible to run this reports based on 30-minute, 20-minute or 15-minute intervals.

Available since	1.6.0
Default page	Call distribution, by hour
Shortcut code	DH03
API block name	CallDistrDO.UnansWaitPerHr
Parameters	-

DH04 - Sales per hour

The total numbers sales and contacts are plotted for each hourly interval. The size of hourly intervals can be controlled by the *default.hourly_slot* configuration property, making it possible to run this reports based on 30-minute, 20-minute or 15-minute intervals.

Available since	1.6.0
Default page	Call distribution, by hour
Shortcut code	DH04
API block name	CallDistrDO.SalesPerHr
Parameters	-

DH05 - Schedule Adherence per hour
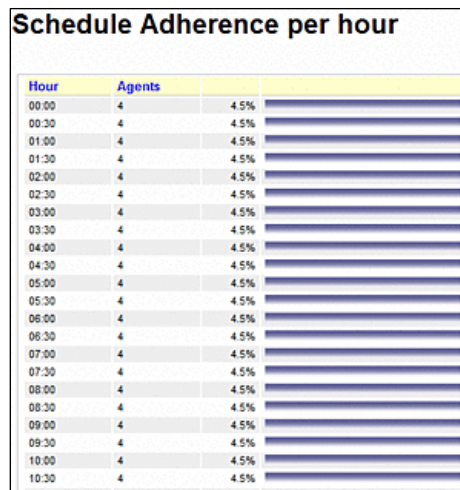
The Schedule Adherence report shows the number of distinct agents that were detected during the given period. This makes it possible to detect the number of different people that had been working on a given moment.

Available since	1.6.0
Default page	Call distribution, by hour
Shortcut code	DH05
API block name	CallDistrDO.StaffPerHr
Parameters	-

DH06 - Queue length per hour

The Queue Length report shows the average length of the queue for each period, giving minimums and maximums. The Steps computation shows how fast each queue progresses during the period, expresses in steps per hour. Note: giving a meaning to the Queue Length reports may be hard in the case of composite queues.

Available since	1.6.0
Default page	Call distribution, by hour
Shortcut code	DH06
API block name	CallDistrDO.QPosPerHr
Parameters	-

DH07 - Inclusive SLA per hour

The inclusive SLA corresponds to the Service Level Agreement metrics shown on the section called "Service level agreement" [32], grouped each hour.

Available since	1.6.0
Default page	Call distribution, by hour
Shortcut code	DH07
API block name	CallDistrDO.InclSlaPerHr
Parameters	-

DH08 - Traffic Analysis by period - per hour

Distribution of calls and agent availability per hour (or interval you specified). The whole 24h are mapped out for ease of comparison.

For a complete description of parameters, see DD08 - Traffic Analysis by period - per day the section called "DD08 - Traffic Analysis by period - per day" [97]

Available since	1.6.0.4
Default page	Call distribution, by hour
Shortcut code	DH08
API block name	CallDistrDO.TrafficAnPerHr
Parameters	-
See also	DD08, DW08

DH09 - Skill sets per hour
This reports shows the unique sets of skills during a specified time period.

Skills per Hour

Day	Ans.	Unans.	Num	Within SLA: 10s.	Product1 english internet technical	Product1 english invoice land	Product1 english invoice mobile	Product1 english land others	Product1 english mobile others	Product1 english mobile technical	Product1 english movie technical
01:00	0	0	0	-	0	0	0	0	0	0	
05:00	7,921	457	8,378	51.8%	13	1	10	5	17	15	
09:00	42,610	2,213	44,823	52.1%	39	10	103	42	160	78	
13:00	41,431	1,880	43,311	53.0%	40	17	83	49	129	64	
17:00	18,609	2,603	21,212	41.1%	27	10	39	17	67	38	
21:00	6	14	20	15.0%	0	0	0	0	0	0	

▶ Export as...

For a complete description of parameters and behavior, see DD09 - Skill sets per day the section called "DD09 - Skill sets per day" [98]

Available since	16.10.11
Default page	Call distribution, by day
Shortcut code	DH09
API block name	CallDistrDO.SkillsPerHour
Parameters	-
See also	DD09, DW09

DH10 - Detail Summary
This report shows the number of taken and lost call by hour.

Detail Summary

Detail Summary

Hour	Taken Calls	Lost Calls	N. of Calls
0:00	0	0	0
1:00	0	0	0
2:00	0	0	0
3:00	0	0	0
4:00	0	0	0
5:00	0	0	0
6:00	0	0	0
7:00	0	0	0
8:00	0	0	0
9:00	0	0	0
10:00	0	0	0
11:00	7	0	7
12:00	2	0	2
13:00	2	0	2
14:00	10	0	10
15:00	17	3	20
16:00	6	1	7
17:00	1	0	1
18:00	1	0	1
19:00	0	0	0
20:00	0	0	0
21:00	0	0	0
22:00	0	0	0
23:00	0	0	0

► Export as...

For a complete description of parameters and behavior, see DD10 - Detail Summary by day the section called "DD10 - Detail Summary" [99]

Available since	17.06.16
Default page	Detail Summary, by hour
Shortcut code	DH10
API block name	CallDistrDO.DetailSummary
Parameters	-
See also	DD10, DW10

Historical reports - Call distribution by day of week
This report shows the weekly behaviour of your queues. The longer the analysis period, the more significant its results will be.

DW01 - Answered call distribution per day of week

The total numbers of call lengths for answered calls are plotted for each day of the week.

Available since	1.6.0
Default page	Call distribution, by day of week
Shortcut code	DW01
API block name	CallDistrDO.AnsDistrPerDOW
Parameters	-

DW02 - Answered call wait time per day of week

The total numbers of call wait time for answered calls are plotted for each day of the week.

Available since	1.6.0
Default page	Call distribution, by day of week
Shortcut code	DW02
API block name	CallDistrDO.AnsWaitPerDOW
Parameters	-

DW03 - Unanswered call wait time per day of week

The total numbers of call wait time for unanswered calls are plotted for each day of the week.

Available since	1.6.0
Default page	Call distribution, by day of week
Shortcut code	DW03
API block name	CallDistrDO.UnansWaitPerDOW
Parameters	-

DW04 - Sales per day of week

The total numbers sales and contacts are plotted for each day of the week.

Available since	1.6.0
Default page	Call distribution, by day of week
Shortcut code	DW04
API block name	CallDistrDO.SalesPerDOW
Parameters	-

DW05 - Schedule Adherence per day of week

The Schedule Adherence report shows the number of distinct agents that were detected during the given period. This makes it possible to detect the number of different people that had been working on a given moment.

Available since	1.6.0
Default page	Call distribution, by day of week
Shortcut code	DW05
API block name	CallDistrDO.StaffPerDOW
Parameters	-

DW06 - Queue length per day of week

The Queue Length report shows the average length of the queue for each period, giving minimums and maximums. The Steps computation shows how fast each queue progresses during the period, expresses in steps per hour. Note: giving a meaning to the Queue Length reports may be hard in the case of composite queues.

Available since	1.6.0
Default page	Call distribution, by day of week
Shortcut code	DW06
API block name	CallDistrDO.QPosPerDOW
Parameters	-

DW07 - Inclusive SLA per day of week

The inclusive SLA corresponds to the Service Level Agreement metrics shown on the section called "Service level agreement" [32], grouped for each day of week.

Available since	1.6.0
Default page	Call distribution, by day of week
Shortcut code	DW07
API block name	CallDistrDO.InclSlaPerDOW
Parameters	-

DW08 - Traffic Analysis by period - per day of week

Distribution of calls and agent availability per day of week.

For a complete description of parameters, see DD08 - Traffic Analysis by period - per day the section called "DD08 - Traffic Analysis by period - per day" [97]

Available since	1.6.0.4
Default page	Call distribution, by day of week
Shortcut code	DW08
API block name	CallDistrDO.TrafficAnPerDOW
Parameters	-
See also	DD08, DH08

DW09 - Skill sets per day of week
This reports shows the unique sets of skills during a specified time period.

For a complete description of parameters and behavior, see DD09 - Skill sets per day the section called "DD09 - Skill sets per day" [98]

Available since	16.10.11
Default page	Call distribution, by day
Shortcut code	DW09
API block name	CallDistrDO.SkillsPerHour
Parameters	-
See also	DD09, DH09

DW10 - Detail Summary by Day of the week
This reports shows the number of taken and lost call by day of the week.

For a complete description of parameters and behavior, see DD10 - Detail Summary by day the section called "DD10 - Detail Summary" [99]

Available since	17.06.16
Default page	Detail Summary by day of the week
Shortcut code	DW10
API block name	CallDistrDO.DetailSummaryDow
Parameters	-
See also	DD10, DH10

Historical reports - Agents and Sessions

Agent activity refers to the behaviour of Asterisk defined agents. If you connect you queues straight to telephone terminals, this section will always be empty.

Each agent may be flagged as being a member of four priority groups:
- Main: the agents usually answering the queue
- Spill: the agents answering the queue if all "Main" agents are busy or unavailable
- Wrap: the agents answering the queue if all "Main" and "Spill" agents are busy or unavailable
- Undefined: this agent is not a member of any priority group for this queue

This feature is useful if priority groups are used in the queue configuration. If they are not used, just assign all agents to "Main" for each queue.

If an unknown agent appears on a queue, it will be marked as "Undefined", written in red.

Agent names are written in blue and are clickable, if you click on them in any of the graphs, you will be lead to a popup that detail the logon and pause history for that agent.

As a default, QueueMetrics will show and count an agent session if and only if the agent handled at least one call during this session. This may not be what you want when you use pause codes - an agent may log on and immediately go on pause to do some back-end activities. If this is the case, you should set the configuration option *default.useRawAgentSessions* to *true* to see all agent sessions.

AG01 - Agent session detail

This report shows:
- The number of available agents in the considered period
- The average agent time availability
- The minumum agent time availability
- The maximum agent time availability
- The cumulated agents time availability

Available since	1.6.0
Default page	Agents and Sessions
Shortcut code	AG01
API block name	AgentsDO.ReportAgents
Parameters	-
See also	

AG02 - Session and pause durations

Session and pause durations

This report shows:
- The number of sessions for each available agent
- The number of pauses for each available agent
- The average pause time
- The percentage between the pause time and the availability time for each agent
- The number of pauses for each session, for each available agent

Available since	1.6.0
Default page	Agents and Sessions
Shortcut code	AG02
API block name	AgentsDO.SessionPauseDur
Parameters	-
See also	

AG03 - Agent availability (for all the queues they are member of)

Agent availability (for all the queues they are member of)

This report shows:
- The agent available time
- The total billable and not billable pause times

In the latest release a new column has been added (Paused on Conversation), which displays occupancy values.

Available since	1.6.0
Default page	Agents and Sessions
Shortcut code	AG03
API block name	AgentsDO.AgentAvail
Parameters	-
See also	

AG04 - Answered calls (for selected queues)

This report shows:
- The number of answered calls for each agent
- The cumulated call time for each agent
- The average call time for each agent
- The average wait time for each call taken
- The percentage of taken calls, related to the total queue calls, for each agent

Available since	1.6.0
Default page	Agents and Sessions
Shortcut code	AG04
API block name	AgentsDO.AnsCallsQueues
Parameters	-
See also	

AG05 - Answered calls by custom group

This report shows:
- The number of answered calls for each defined agent group
- The cumulated call time for each defined agent group
- The average call time for each defined agent group
- The percentage of taken calls, related to the total queue calls, for each defined agent group The report displays separately the following cases:
- Agents that are known to QueueMetrics but have no custom group, are counted under the group "-"
- Agents that are not known to QueueMetrics but found in the analysis are counted under "Undefined".

Available since	1.6.0
Default page	Agents and Sessions
Shortcut code	AG05
API block name	AgentsDO.AnsCallsCG
Parameters	-

AG06 - Answered calls by location

This report shows:
- The number of answered calls for each defined location
- The cumulated call time for each defined location
- The average call time for each defined location
- The average wait time for each defined location
- The percentage of taken calls, related to the total queue calls, for each defined location

Available since	1.6.0
Default page	Agents and Sessions
Shortcut code	AG06
API block name	AgentsDO.AnsCallsLocation
Parameters	-

AG07 - Answered calls by service group

This report shows:
- The number of answered calls for each defined service group
- The cumulated call time for each defined service group
- The average call time for each defined service group
- The percentage of taken calls, related to the total queue calls, for each defined service group

Available since	1.6.0
Default page	Agents and Sessions
Shortcut code	AG07
API block name	AgentsDO.AnsCallsSG
Parameters	-
See also	

AG08 - Agent Performance by ACD Group

One entry is presented for each agent session. What makes this report different from most is that inbound and outbound activity for the agent is aggregated.

Items are computed as:
- Level: the agent level this agent belongs to
- Agent: agent name (if present) or Asterisk internal code if unknown. By clicking on it, the session details open.
- Login: Session login time
- Duration: Session duration
- N.calls in: Number of calls taken for this queue(s)
- N calls out: Number of calls outbound made for these campaign(s)
- Tot calls: calls in + calls out
- Average duration IN
- Average duration OUT
- Average duration for all (weighted)
- Available: precentage of idle time
- On call IN: percentage of time on call inbound
- On call OUT: percentage of time on call outbound
- Pause Billable: percentage of time on a pause code marked as billable
- Pause Unbillable: percentage of time on a pause code marked as unbillable, or not specified, or unknown. (a.k.a. Clerical time)

Available since	1.6.0.4
Default page	Agents and Sessions
Shortcut code	AG08
API block name	AgentsDO.PerformanceAcdGroups
Parameters	-

AG09 - Agent Occupancy Report

The report details the occupancy rate for each and all agents in the current analysis.
- The Agent column contains the decoded name, level and current group of the agent (if defined, expressed as an icon like elsewhere in QM). By clicking on the agent name, it is possible to open a popup with the session details for that agent.
- Total session time is the sum of the duration of all sessions in the current analysis, from start to end.
- Pause Billable is the total time on Billable pauses, for all sessions considered.
- Pause Non Billable is the total time on Non-Billable pauses, for all sessions considered.
- Total pause time is the sum of all pauses for all sessions considered.
- Total talk time is the sum of all time that the specified agent spent in conversation during the sessions considered.
- Occupancy is computed as: Talk time / (Total session - Total Pause) and expressed as a percentage, as per the numeric examples show in the table above.

In the latest release a new column has been added (Paused Conversation) which is subtracted from the total pauses time;

this way Occupancy will never be able to be over 100%.

The first line is labeled "All agents" and is computed according to the following rules:
- Total session time, Pause billable, Pause non billable, Pause wrap, Total pause, Total talk time are computed as sums of the rest of data in each column
- Occupancy is computed as: Sum talk time / (Sum Total session - Sum Total Pause)

The Occupancy report allows you to define custom calculations by entering one or more custom formulas in the Parameters field when editing your report.

For example, if you add the formula:
Productivity = (TALKTIME + PAUSEBILL) / (SESSION - ($PAUSETIME[AAA] + $PAUSETIME

A new column "Productivity" will be computed based on the formula supplied. At the moment, the formula will always be displayed as a percentage.

New columns can be programmed as explained in User-Programmable reports the section called "User- Programmable Reports" [142]

A version of this data block meant to be completely customized is available as AG18.

Available since	1.6.2.4 - Formulas available since 15.10.5
Default page	Agents and Sessions
Shortcut code	AG09
API block name	AgentsDO.AgentOccupancy
Parameters	Agent parameters - See Variables bound for Agent blocks the section called "Variables bound for Agent blocks" [143]
See also	AG18 the section called "AG18 - Programmable Agent Performance" [124]

AG10 - Agent Session Time by Hour

The report details the total presence time of each agent over the 24h for which you are running the current analysis.
- The Agent column contains the agent's decoded name, level and current group (if defined, expressed as an icon like elsewhere in QM).
- Total time is the sum of the duration of all sessions included in the current analysis, from start to end.
- One or more columns are present for each hour's timeframe where total session information is present

Available since	1.7.2
Default page	Agents and Sessions
Shortcut code	AG10
API block name	AgentsDO.AgentBillableTimeByHour
Parameters	-

AG11 - Agent Payable Time by Hour

The report details the payable presence time of each agent over the 24h for which you are running the current analysis.
- The Agent column contains the agent's decoded name, level and current group (if defined, expressed as an icon like elsewhere in QM).
- Total time is the sum of the duration of payable session time within the current analysis, from start to end.
- One or more columns are present for each hour's timeframe where payable session information is present

Available since	1.7.2
Default page	Agents and Sessions
Shortcut code	AG11
API block name	AgentsDO.AgentPayableTimeByHour
Parameters	-

AG12 - Agent Billable Time by Hour

The report details the billable presence time of each agent over the 24h for which you are running the current analysis.
- The Agent column contains the agent's decoded name, level and current group (if defined, expressed as an icon like elsewhere in QM).
- Total time is the sum of the duration of billable session time within the current analysis, from start to end.
- One or more columns are present for each hour's timeframe where billable session information is present

Available since	1.7.2
Default page	Agents and Sessions
Shortcut code	AG12
API block name	AgentsDO.AgentBillableTimeByHour
Parameters	-

AG13 - Agent Session View

This report shows a set of calls aggregated by agents. It reports session information and shows, for each agent, the main figures for aggregated calls related to that agent. The first row reports the overall aggregated values.

Each row in the report shows:
- The agent name (the first line contains all agents information)
- Presence, calculated as sum of all agent sessions (so including pauses time)
- Talk time
- Idle time, calculated by subtracting pauses and talk time from the presence figure
- Billable time
- Not billable time
- Sales: the number of taken calls flagged "sales"
- S PH (Sales per hour): the number of sales per hour ratio
- QC: the number of taken calls flagged as qualified contacts
- QC PH: Qualified contacts per hour
- Cont: number of taken calls flagged as "contacts"
- CPH: Contacts per hour
- Offered: Overall taken + lost calls
- Answered: taken calls
- Conv: convertion ratio
- QConv: qualified convertion ratio
- Goal Ftrs: number of goal flagged feature codes found in the call dataset
- Ftrs: number of feature codes found in the call dataset
- Avg: average talk time
- T: by clicking on that icon, a popup with the taken call list (used to count the aggregated) is shown
- L: by clicking on that icon, a popup with the lost call list (used to count the aggregated) is shown

Regarding inbound calls, when looking at a single agent, the concept of "offered call" will coincide with the concept of "answered call". This is because this report does not keep track of the various attempts for agent, but only registers the call as an answered call for the agent that actually answered the call. This will eventually result in the first row containing a number of offered calls that is higher than the number of answered calls, while on the single agents rows the offered and answered calls will be the same.

Available since	15.9
Default page	Agents and Sessions
Shortcut code	AG13
API block name	AgentsDO.AgentSessions
Parameters	-

AG14 - Queue Session View

This report shows a set of calls aggregated by queue. It reports queue information and shows, for each queue, the main figures for aggregated calls related to that queue. The first row reports the overall aggregated values.

Each row in the report shows:
- The queue name (the first line contains all queues information)
- Presence, calculated as sum of all agent sessions in the queue
- Talk time
- Idle time, calculated by subtracting pauses and talk time from the presence figure
- Billable time
- Not billable time
- Sales: the number of taken calls flagged "sales"
- S PH (Sales per hour): the number of sales per hour ratio
- QC: the number of taken calls flagged as qualified contacts
- QC PH: Qualified contacts per hour
- Cont: number of taken calls flagged as "contacts"
- CPH: Contacts per hour
- Offered: Overall taken + lost calls
- Answered: taken calls
- Conv: convertion ratio
- QConv: qualified convertion ratio
- Goal Ftrs: number of goal flagged feature codes found in the call dataset
- Ftrs: number of feature codes found in the call dataset
- Avg: average talk time
- T: by clicking on that icon, a popup with the taken call list (used to count the aggregated) is shown
- L: by clicking on that icon, a popup with the lost call list (used to count the aggregated) is shown

QueueMetrics uses an approach where it does not exactly track sessions per queue. If an agent joins and leaves multiple queues during the same session, QueueMetrics does not track where each session starts and ends. Therefore, while we may get accurate details of

the time an agent was available, if an agent may cherry pick queues or work on multiple queues at once, we will have to find a way to estimate when its time was spent. In order to do this QueueMetrics counts the local time an agent was available per session, then it multiplies this value by the percentage of talk time that happened on each queue.

Available since	15.9
Default page	Agents and Sessions
Shortcut code	AG14
API block name	AgentsDO.QueueSessionsView
Parameters	-

AG15 - Agents by Hour

The report details the number of agents available on the selected queues each hour for each day in the report.
- The first column (Day) reports the date who the line is referred to

- The next 24 columns report the number of agents found for at least one second in the associated hour. Numbers are reported black if the timeband has been covered completely by the agents, are reported red if at least one agent is not present for the whole hour. In this case, by flying over the red number the user can have an idea on what's the coverage percentage for the timeband. The coverage is counted as the sum of periods, in seconds, the agents cumulated in that timeband, divided by 3600 seconds.
- The last column reports the agents seen, for at least one second, during the day

Available since	14.10.5-938
Default page	Agents and Sessions
Shortcut code	AG15
API block name	AgentsDO.AgentByHour
Parameters	-
See also	

AG16 - Tag Session View

This report shows a set of calls aggregated by tag. It reports tags information and shows, for each tag, the main figures for aggregated calls related to that tag. The first row reports the overall aggregated values.

Each row in the report shows:

- The tag name (the first line contains all tags information)
- Presence, calculated as sum of all agent sessions for agents taking calls with related tag
- Talk time
- Idle time, calculated by subtracting pauses and talk time from the presence figure
- Billable time
- Not billable time
- Sales: the number of taken calls flagged "sales"
- S PH (Sales per hour): the number of sales per hour ratio
- QC: the number of taken calls flagged as qualified contacts
- QC PH: Qualified contacts per hour
- Cont: number of taken calls flagged as "contacts"
- CPH: Contacts per hour
- Offered: Overall taken + lost calls
- Answered: taken calls
- Conv: convertion ratio
- QConv: qualified convertion ratio
- Goal Ftrs: number of goal flagged feature codes found in the call dataset
- Ftrs: number of feature codes found in the call dataset
- Avg: average talk time
- T: by clicking on that icon, a popup with the taken call list (used to count the aggregated) is shown
- L: by clicking on that icon, a popup with the lost call list (used to count the aggregated) is shown

QueueMetrics uses an approach where it does not exactly track sessions per tag. If an agent joins and leaves multiple queues during the same session, QueueMetrics does not track where each session starts and ends. Therefore, while we may get accurate details of the time an agent was available, if an agent may cherry pick queues or work on multiple queues at once, we will have to find a way to estimate when its time was spent. In order to do this QueueMetrics counts the local time an agent was available per session, then it multiplies this value by the percentage of talk time that

happened on each queue.

Available since	15.09
Default page	Agents and Sessions
Shortcut code	AG16
API block name	AgentsDO.TagSessionsView
Parameters	-
See also	

AG17 - Cumulated sessions by Queue, Tag

This report shows a set of calls aggregated by queue and tag. It reports queue and tags information and shows, for each queue/tag pair, the main figures for aggregated calls related to that pair. The first row reports the overall aggregated values.

Each row in the report shows:

- The queue name (the first line contains all queues information)
- The tag name (the first line contains all tags information)
- Presence, calculated as sum of all agent sessions for agents taking calls with related tag
- Talk time
- Idle time, calculated by subtracting pauses and talk time from the presence figure
- Billable time
- Not billable time
- Sales: the number of taken calls flagged "sales"
- S PH (Sales per hour): the number of sales per hour ratio
- QC: the number of taken calls flagged as qualified contacts
- QC PH: Qualified contacts per hour
- Cont: number of taken calls flagged as "contacts"
- CPH: Contacts per hour
- Offered: Overall taken + lost calls
- Answered: taken calls
- Conv: convertion ratio
- QConv: qualified convertion ratio
- Goal Ftrs: number of goal flagged feature codes found in the call dataset
- Ftrs: number of feature codes found in the call dataset
- Avg: average talk time
- T: by clicking on that icon, a popup with the taken call list (used to count the aggregated) is shown
- L: by clicking on that icon, a popup with the lost call list (used to count the aggregated) is shown

QueueMetrics uses an approach where it does not exactly track sessions per queue/tag. If an agent joins and leaves multiple queues during the same session, QueueMetrics does not track where each session starts and ends. Therefore, while we may get accurate details of the time an agent was available, if an agent may cherry pick queues or work on multiple queues at once, we will have to find a way to estimate when its time was spent. In order to do this QueueMetrics counts the local time an agent was available per session, then it multiplies this value by the percentage of talk time that happened on each queue.

Available since	15.09
Default page	Agents and Sessions
Shortcut code	AG17
API block name	AgentsDO.SessionByQueueTagReport
Parameters	-
See also	

AG18 - Programmable Agent Performance

This block just includes the first two columns: Level and Agent. All other columns are to be programmed by the user as explained in User-Programmable reports the section called "User- Programmable Reports" [142] .

This block is meant as a completyl configurable version of block AG09.

Available since	15.10.8
Default page	Agents and Sessions
Shortcut code	AG18
API block name	AgentsDO.ProgAgPerf
Parameters	Agent parameters - See Variables bound for Agent blocks the section called "Variables bound for Agent blocks" [143]
See also	AG09 the section called "AG09 - Agent Occupancy Report" [115]

AG19 - Detailed Pause Tracking

This block shows detailed information about agent pauses, including total pause time, average duration, maximum duration, minimum duration and number of pauses. This information is aggregated by agent and pause. For each agent/pause pair the agent's name and the pause code are displayed, together with the information described above.

Available since	16.10.10
Default page	Agents and Sessions
Shortcut code	AG19
API block name	AgentsDO.DetailedPauses
Parameters	-
See also	

AG20 - Agent Talk Time Per Hour

This block breaks down all the agents' talk times in one-hour long timeslots. This information is aggregated by agent, so you can keep track of how many minutes or hours of conversation your agents are having during the course of the day, across a given period of time. The name of the agent and the agent's level are displayed as well as the information listed above.

Available since	16.10.13
Default page	Agents and Sessions
Shortcut code	AG20
API block name	AgentsDO.AgentTTPerHour
Parameters	-
See also	

Historical reports - Details of agent sessions and pauses

AD01 - Detail of agent sessions

For each agent session, the start and end times are recorded, together with the total duration in seconds. If the agent logs on via the call back function, the designated call back extension is shown.
The number of pauses and the total pause time in seconds is shown.

The "Srv" column tells you on which server an agent was working in case you set up a cluster of Asterisk servers.

It is possible to sort the table for each title, in either descending and ascending order. To do this, click once on the desired title for descending sort, and twice for ascending sort. Once the table is sorted, an arrow symbol will appear close to the title, so you know on which column it was sorted last. As the sorting is done on the client machine, it may take a while with

very large tables.

Available since	1.6.0
Default page	Details of Agent sessions and pauses
Shortcut code	AD01
API block name	DetailsDO.AgentSessions
Parameters	-
See also	

AD02 - Detail of agent pauses

This table shows the specific pauses that each agent took and the pause code that was entered for each pause. It also shows whether the pause taken was considered to be billable or non-billable.

Available since	1.6.0
Default page	Details of Agent sessions and pauses
Shortcut code	AD02
API block name	DetailsDO.AgentPauses
Parameters	-
See also	

Popup of agent activity

This is not a normal data block but it is the popup that is displayed whenever you click on an agent's name.
- The top table reports session, pause, billable and payable time
- The bottom table shows the details of all agent sessions, with a break-up of all pauses that were made during that session. For each pause with a known pause code, their billable-payable status is displayed:

- **BP**: Pause is Billable and Payable
- **BNP**: Pause is Billable but not payable (be careful!)
- **NBP**: Pause is not billable but Payable
- **NBNP**: Pause is neither billable nor payable

In the latest release a new "Conversation" column has been added on, to the right of the "On Pause" column, which displays the pause time there may have been while actually in conversation.
You can close the popup by clicking on the Close button.

Historical reports - Call outcomes

If your agents are entering Pause codes or Call outcomes, the "Outcomes" tab will let you report on the information they just entered.

OU01 - Outcomes

This report shows:
- How much billable time there has been on this system, broken down by ACD/call time ("agent available time") and billable activities (agent on pause)
- The total non billable time (e.g. lunch, breaks)
- The total number of Contacts, Qualified Contacts and Sales, as defined by call outcome codes
- The Sales per Hour (SPH), Qualified Contacts per hour (QCPH) and Contacts per Hour (CPH) ratios
- The Conversion index that is the percentage of sales over the total number of sales and contacts.

Available since	1.6.0
Default page	Call outcomes
Shortcut code	OU01
API block name	OutcomesDO.GeneralRep
Parameters	-
See also	

OU02 - Call results, by outcomes

This report shows:
- The number of total calls grouped by outcome
- The number of answered calls grouped by outcome
- The number of unanswered calls grouped by outcome
- The percentage of calls related to each outcome

- The percentage of taken calls related to each outcome (available since 16.10.13)

Available since	1.6.0
Default page	Call outcomes
Shortcut code	OU02
API block name	OutcomesDO.CallResByOutcome
Parameters	-

OU03 - Billable activities

In this report you can find details explaining Billable activities, with average, minimum and maximum session durations, and a percentage on all activities of the same kind.

Available since	1.6.0
Default page	Call outcomes
Shortcut code	OU03
API block name	OutcomesDO.ActivBillable
Parameters	-

OU04 - Non billable activities

In this report you can find details explaining Non billable activities, with average, minimum and maximum session durations, and a percentage on all activities of the same kind.

Available since	1.6.0
Default page	Call outcomes
Shortcut code	OU04
API block name	OutcomesDO.ActivNotBillable
Parameters	-

OU05 - Detailed agent report

The Detailed Agent Report will show, for each agent:
- The Available (ACD) time, as an absolute value and a percentage of its total time logged on
- The Billable time, as an absolute value and a percentage of its total time logged on
- The Non-Billable time, as an absolute value and a percentage of its total time logged on
- The number of Sales And Contacts the agent had (if a sale is counted a s both a Sale and a Contact, it's counted only once as a Sale)
- The Sales per Hour (SPH) and Contacts per Hour (CPH) ratios for this agent

- The Conversion ratio, which is the percentage of sales over the total number of sales and contacts.

Available since	1.6.0
Default page	Call outcomes
Shortcut code	OU05
API block name	OutcomesDO.AgentReportDetailed
Parameters	-
See also	

OU06 - Outcomes per agent

This data block displays all the status codes for each call handled by each agent within the current reporting selection.

If a call has no associated status code, then it is counted under a special status code called "-". The columns show:
- The agent name. As an agent might have multiple status codes, the agent name will appear multiple times. If the agent belongs to an agent group, the proper icon is displayed.
- The call outcome, specified both as a code and a description (if it is present)
- The type currently associated with the status code
- The total number of calls for that agent under that status code All columns are client sortable.

Available since	13.04.3
Default page	Call outcomes
Shortcut code	OU06
API block name	OutcomesDO.AgentOutcomes
Parameters	-

OU07 - Call results, by tag

This data block displays calls aggregated by tag code. The first line reports the figures for all aggregated call. The second row, marked with the *dash* code as tag, counts aggregates for all calls with no tag code assigned.

The columns show:
- The tag code name
- Total number of calls aggregated
- Taken calls
- Average talk time
- Average wait time on taken calls
- Number of calls flagged as sales
- Number of calls flagged as contact
- Number of calls flagged as qualified contact

- Conversion ratio
- Qualified conversion ratio
- Number of lost calls
- Average wait on lost calls
- Average wait on IVR calls
- Time spent on IVR
- Ratio and histogram for total calls the row is reporting, compared with the overall total call in the report

All columns are client-sortable. Clicking on the lens icons present in the report will open a popup with call details (taken, lost and IVR) counted in the aggregation.

Available since	15.09
Default page	Call outcomes
Shortcut code	OU07
API block name	CallTagDO.CallResByTag
Parameters	-

OU08 - Call results, by feature code

Call results, by feature code

This data block displays calls aggregated by feature code. The first line reports the figures for all aggregated call. The second row, marked with the *dash* code as feature, counts aggregates for all calls with no feature code assigned.

The columns show:
- The feature code name
- Goal/Not goal flag for feature code
- Total number of calls aggregated
- Feature conversion factor, counted as ratio between the total calls and the call flagged as contact
- Taken calls
- Average talk time
- Average wait time on taken calls
- Number of calls flagged as sales
- Number of calls flagged as contact
- Number of calls flagged as qualified contact
- Qualified conversion ratio
- Number of lost calls
- Average wait on lost calls
- Ratio and histogram for total calls the row is reporting, compared with the overall total call in the report

All columns are client-sortable.

Available since	15.09
Default page	Call outcomes
Shortcut code	OU08
API block name	OutcomesDO.CallResByFeature
Parameters	-
See also	

Historical reports - IVR tracking

The following reports let you trrack information about IVR calls and goals. A detailed explanation of IVR can be found in: the section called "Tracking IVR data" [366]

IV01 - IVR Traversals

This block shows:
- The IVR paths examined, in total (as the sum of all choices)
- For each IVR path, the choices made on that path (prepended by #) For each path, the following information is computed:
- The number of calls traversing that path
- The number of IVR goals reached immediately after that path (a goal can be a queue or an activity expressly earmarked as a goal)
- The success rate, as the percentage of goals on all calls that entered the menu
- The number of hang-ups made in that IVR menu
- The attrition rate, as the number of hang-ups versus total number of calls

Available since	13.03
Default page	IVR
Shortcut code	IV01
API block name	IvrDO.IvrReport
Parameters	-

IV02 - IVR Timing

This block shows:
- The IVR paths examined, in total (as the sum of all choices)
- For each IVR path, the choices made on that path (prepended by #) For each path, the following information is computed:
- The number of calls traversing that path
- The average traversal time for that IVR menu
- The total traversal time (as the sum of all times for each call)
- The maximum and minimum traversal times

Available since	13.03
Default page	IVR
Shortcut code	IV02
API block name	IvrDO.IvrTiming
Parameters	-

IV03 - IVR Goals

This block shows:
- For each goal, all the paths that lead to that goal and their selection (the digit that was pressed in order to reach the goal)
- The number of calls that reached each goal
- The average, minimum and maximum traversal time to to reach the goal. The time is computed since the call is first tracked, and not only on the final IVR menu.

Available since	13.03
Default page	IVR
Shortcut code	IV03
API block name	IvrDO.IvrGoals
Parameters	-

Historical reports - First Call Resolution (FCRs)

First Call Resolution lets you track cases that are not resolved in a single call and so the customer ends up calling multiple times in a specified time frame.

Repeated callers and clusters

At the basis of FCR, there is a definition of "repeated call". A call is considered "repeated" if:
- the caller-id is the same for multiple calls; e.g. if some user calls from a different number, we do not consider this a repeated call.
- the same caller-id is calling multiple times that are separated by less than a given time period, called "margin".
- the margin is controlled by a configuration property called default.groupingMargin, that can be changed in the system configuration and which default value is 168 hours (7 days).
- the caller-id is considered the same if after a normalization process (stripping of whitespaces and non- numeric characters) the same caller-ids match
- the caller-id is skipped (not clustered) if after the normalization it is blank – e.g. caller-ids like "" or "anonymous" will be skipped. Any calls with a blank caller-id will be counted as a single-call, as we do not have further information to cluster it.
- the normalized caller-id appears at least twice

For example, given the following set of calls and a marging of one week The data blocks are computed in three different flavors:
- Taken calls
- Lost calls
- All calls

FC01/2/3 - Break-down of recalls

Recalls on answered calls

	Clusters	Delta	%	Total calls	Total %
FCRs - 1	43		68.3%	43	44.3%
Up to attempts: 2	54	+11	85.7%	65	67.0%
Up to attempts: 3	58	+4	92.1%	77	79.4%
Up to attempts: 4	63	+5	100.0%	97	100.0%

▶ Export as...
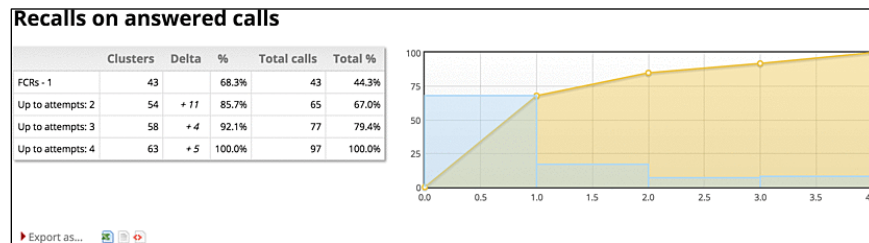
This report aims at letting you know how many calls were clustered.

This report is built to have a "shape" that is familiar for users used to SLAs, and is completed by a similar graph.
In detail:
- FCRs are calls that appear only once; while the other "classes" will be calls "up to" so many attempts
- "Up-to" lines will be present up to the maximum present in the report; considering all integers from 2 to the maximum included
- Clusters is the number of clusters that are "up to", so clusters that belong to the current class plus all previous ones
- Delta is the number of clusters that belong to the current class
- Percentage is the number of calls of the current class
- Total is the growing percentage of sorted classes, and always ends at 100%
- Total Calls is the number of calls that are "up to", so belong to the current class plus all previous ones

Available since	15.10.11
Default page	FCRs
Shortcut code	FC01
API block name	FcrDO.RecallsOk
Parameters	-
See also	Taken calls

Available since	15.10.11
Default page	FCRs
Shortcut code	FC02
API block name	FcrDO.RecallsKo
Parameters	-
See also	Lost calls

Available since	15.10.11
Default page	FCRs
Shortcut code	FC03
API block name	FcrDO.RecallsAll
Parameters	-
See also	All calls

FC04/5/6 - Recall details
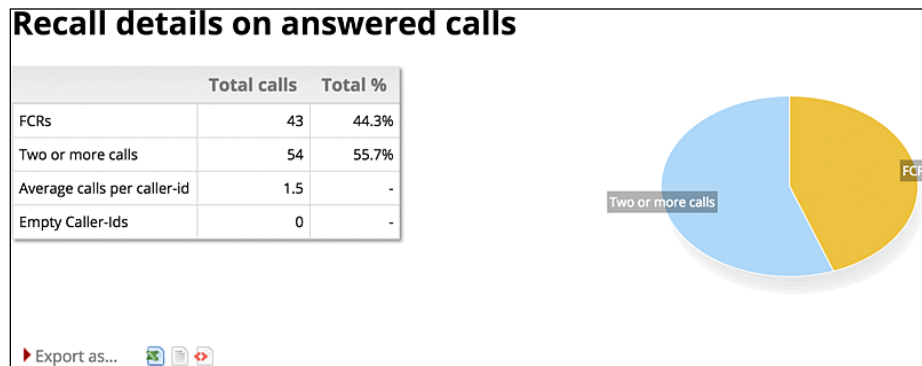
This report shows the total number of FCRs, clusters and calls with an empty IDs. It has a pie chart showing FCRs vs clusters.

Available since	15.10.11
Default page	FCRs
Shortcut code	FC04
API block name	FcrDO.ClidsOk
Parameters	-
See also	Taken calls

Available since	15.10.11
Default page	FCRs
Shortcut code	FC05
API block name	FcrDO.ClidsKo
Parameters	-
See also	Lost calls

Available since	15.10.11
Default page	FCRs
Shortcut code	FC06
API block name	FcrDO.ClidsAll
Parameters	-
See also	All calls

FC07/8/9 - Cluster details

This block is a list of all clusters, sorted by the number of attempts and showing the cluster's first and last call. Only clustered calls are included.

For all clusters including "taken" calls, the "Taken" column appears showing how many of those calls were counted and offering a link to a pop-up from which you can access the calls. The same goes for "Lost" calls. If both are counted, both columns are present.

Notes:

- When clicking on the icon to see calls, you will see all calls with the same caller-id belonging to the current report, whether they belong to the current cluster or not; so that you can have a general overview of all recalls in the current report

Available since	15.10.11
Default page	FCRs
Shortcut code	FC07
API block name	FcrDO.ClustersOk
Parameters	-
See also	Taken calls

Available since	15.10.11
Default page	FCRs
Shortcut code	FC08
API block name	FcrDO.ClustersKo
Parameters	-
See also	Lost calls

Available since	15.10.11
Default page	FCRs
Shortcut code	FC09
API block name	FcrDO.ClustersAll
Parameters	-
See also	All calls

FC10 - Retry Rate and Lost Caller Behaviour

This report shows information about customers retry behaviour. We can see how many calls are registered as being the n-th attempt, based on the Customer Attempt number in the first column. The information is the following:

- Customer Attempts
- Total Attempts
- Long Calls
- Short Calls
- Taken Calls
- Lost Calls Notes:
- the number of seconds needed for a call to be classified as long is dictated by the following configuration property: default.shortcall_treshold = 30 . If the property is not specified, it will deafault at 30.

Available since	17.06.16
Default page	FCRs
Shortcut code	FC10
API block name	FcrDO.RetryRate
Parameters	-
See also	NA

FC11 - Most Frequent Called-Calling Number

This report shows aggregated data by caller/called number. We can deduce by looking at this report, the most frequent callers and the duration of their calls. A special icon is also present to inspect the signular calls that make up the aggregated data.

The information is the following:
- Number
- N. of Calls
- Taken Calls
- Lost Calls
- % of Taken Calls
- Total Duration
- Avg. Duration Notes:
- Calls that show "anonymous" as caller number are calls which have an empty or undeterminable caller id. these calls are all evaluated together. Similarly, calls that show "~" as caller id, are calls coming from caller numbers that have only called once, we can refer to them as FCR (First Call Resolution).

Available since	17.06.16
Default page	FCRs
Shortcut code	FC11
API block name	FcrDO.MostFrequentNumber
Parameters	-
See also	NA

Externally generated reports

QueueMetrics is able to fetch and display data blocks generated by external systems. This may be used to enhance the standard QueueMetrics reports with data that is handled outside of QueueMetrics - for example, data from a CRM application or order tracking systems or anything else that makes sense to display together with queue data.

In order to use these blocks, you need to create external data providers that will generate the data that QueueMetrics is to display. Details on how to do this can be found in the *XML-RPC Manual* in the section called *Dynamic blocks in QueueMetrics*.

XS01 - Plain HTML
This data block includes a block of HTML right into the QueueMetrics report page.

For example, in order to include an HTML fragment generated by a server located at http://server/ page.php you would set: url=http://server/page.php in the data block's parameter field.

Available since	13.12
Default page	None
Shortcut code	XS01
API block name	ExtSourceDO.HTML
Parameters	*url*: The URL that points to the page generating HTML
See also	

XS02 - XML-RPC data source
This data block includes a formatted table of data into the QueueMetrics report page. The data must be generated in a specified format by an XML-RPC server, as described on the *XML-RPC Manual*.

Available since	13.12
Default page	None
Shortcut code	XS02
API block name	ExtSourceDO.XMLRPC
Parameters	*url*: The URL that points to the XML-RPC server
See also	XS01

XS03 - JSON data source

This data block includes a formatted table of data into the QueueMetrics report page. The data must be generated in a given format by a JSON server, as described on the *XML-RPC Manual*.

Available since	13.12
Default page	None
Shortcut code	XS03
API block name	ExtSourceDO.JSON
Parameters	*url*: The URL that points to the page generating a JSON document
See also	XS01

User-Programmable Reports

User Programmable Reports are meant to be completely customized by the end-user. As such, they usually include a fixed set of columns that will be displayed, plus a set of facilities for creating new columns.

Those columns are defined through the "Parameters" field of the data block, where new columns are to be configured. For example, in order to add a column that displays the time an agent was available, you could enter a definition like:

Tot Avail [hms]= SESSION - (PAUSETIME + (TALKTIME + OOSCALL))

This definition is made up of three parts:

- A column name: how the column's title will appear in the reports;
- A formatter: how the resulting value is to be displayed. Formatting is expressed as a token between square brackets just before the equal sign;
- A formula: how the value is to be computed.

The *column name* is a free string; you may put anything you want in it. The *formatter* must be one of the following values:

- hms - a number of seconds displayed as "[hours:]minutes:seconds"
- % - prints a float number into a percentage, es .365 is displayed as 36.5%
- n – prints an integer, right-aligned
- f – prints a float number

If no formatter is specified, the *percent* formatter is used. All numbers are printed according to the current rules set by your current language and locale (e.g. number of decimal digits, decimal separators, etc.)

The *formula* lets you insert an algebraic formula. This formula works on a set of bound variables (as specified below) that the block will replace with the correct values for each row.

The following operators are supported, with natural precedence rules:

- Addition: 2 + 2
- Subtraction: 2 - 2
- Multiplication: 2 * 2
- Division: 2 / 2
- Exponentation: 2 ^ 2

- Unary Minus and Plus: +2 - (-2)
- Modulo: 2 % 2
- Rounded brackets are supported to force the order of evaluation. Use them freely.
- The following functions are supported:
  - abs: absolute value
  - acos: arc cosine
  - asin: arc sine
  - atan: arc tangent
  - cbrt: cubic root
  - ceil: nearest upper integer
  - cos: cosine
  - cosh: hyperbolic cosine
  - exp: exponent (e^x)
  - floor: nearest lower integer
  - log: logarithmus naturalis (base e)
  - log10: logarithm (base 10)
  - log2: logarithm (base 2)
  - sin: sine
  - sinh: hyperbolic sine
  - sqrt: square root
  - tan: tangent
  - tanh: hyperbolic tangent
- Specific data blocks may implement custom functions. Those are prefixed by the dollar sign and depend on the agent block.
- Any error in the formula (e.g. division by zero, incorrect formula, missing parentheses, referencing an unbound variable) will cause the formula to return zero. An error will be printed on the system log.

You can have multiple columns defined; it is also valid to have empty lines in the parameters block and comments, prefixed by "#".

When creating new formulas, it is often useful to print all referenced items by themselves, so you can double-check them, and comment them out when you don't need them anymore.

Variables bound for Agent blocks
The following variables are automatically bound and therefore usable in formulas:
- TALKTIME: The total conversation time
- SESSION: The total session time
- OVERLAPPING: The total overlapping conversation time
- OOSCALL: Out-of-session call time
- PAUSETIME: The total pause time
- PAUSEBILL: The total time of billable pauses
- PAUSENBILL: The total time of non-billable pauses
- PAUSEPAY: The total time of payable pauses
- PAUSENPAY: The total time of non-payable pauses
- NCALLS: The total number of calls for this agent
- NCALLSIN: The total number of inbound calls for this agent
- NCALLSOUT: The total number of outbound calls for this agent The following custom functions are implemented:
- $PAUSETIME[X]: returns the total duration of all pauses with the pause code "X" for the current agent

All times are expressed in seconds.

JSON and XML-RPC access
As Programmable blocks often do not include much interesting information by themselves, when referenced though their own block code will only return fixed columns.
It is still possible to access them as they are configured in a report by referencing their direct code; this will be displayed by the Reports editor as soon as you enter parameters for an existing data block. The format will usually be "DRI.num", where *num* depends on your QueueMetrics install (different QueueMetrics systems will report different DRI codes for a block defined in the same way).

As accessing a data block that is a part of a report requires the accesing used to have the correct grants to access the report, screen and block, these security rules will be also

enforced on "robot" users doing data queries. Make sure that, if you key-protect reports or elements in them, the "robot" user has all needed grants.

Chapter 3. The real-time status panel

The real time status panel can be accessed by clicking the "Start real-time monitoring" label from the home page. It will show a page similar to the one below:
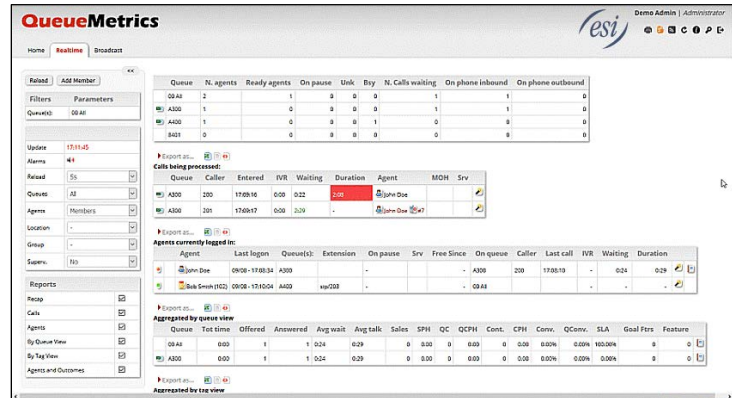
On the top of the page there is a control table showing the last update timestamp and other dropdown selectors as specified below:

- **Reload**: It defines the update period will be used to refresh the shown data
- **Agents**: Shows or hides the Agents currently logged in table
- **Queues**: Shows all queues or only active queues
- **Agents**: Toggle between all agents or members only agents
- **Location**: If granted by user permissions, defines which location is shown
- **Group**: Defines wich agent group is shown
- **Superv.**: Filter out agents not supervised by current user

The page is able to auto-refresh in background at the period specified in the first dropdown but you can anyway force a faster reload by clicking the "Reload" button.

Next to the control table, there is the data section. Depending on the status of previously mentioned dropdown, three sections could be shown.

The first is a table showing a summary of all calls flowing through queues. Following that summary, there is a table showing which calls are currently handled by the queue system, then the agents logged in at the moment.

This page is invaluable because can tell you in a glimpse what's happening in the call center; it is meant to stay open in a window on the CC manager's workstation to have the exact feeling of what is going on at the moment.

On the sample page above, you can see two calls and two connected agents. Just like in the main analysis, you can choose which queues you want to monitor to avoid being overwhelmed by data.

You can also see that the current call environment has triggered a number of yellow and red alarms, as specified in the queue definition. You can configure red an yellow alarms for most numeric values that appear on screen - see the chapter the section called "Setting attention levels (Red and yellow alarms)" [215]. You can also set sounds linked to yellow or red alarms that will be played if a red or yellow alarm is present.

Since the release of QueueMetrics 12.04, this panel has an added feature, which is the "Add Member" button, that allows an administrator or supervisor to add an agent to a queue, as required. Users holding the RT_ADDMEMEBER key will be able to add agents directly from the Realtime page.
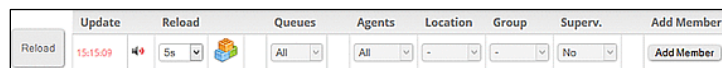
In order to change the default audio files, see "Appendix D: System Preferences". If you want to turn them off completely, just set them to blank.

The top menu bar shown in the above picture has been replaced by a left menu panel since QueueMetrics 15.10.

Since the release of QueueMetrics 15.09, the top panel contains a *three bricks* icon. Flying over with the mouse allows the user to select the report blocks will be shown in the realtime page. The number of available blocks is dependente by a set of configuration keys.
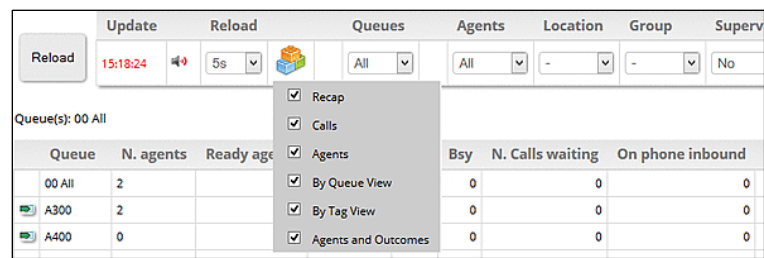
Thre *three bricks* icon has been replaced since QueueMetrics 15.10 by a checkbox list located on the new menu bar located on the left side, as reported below.
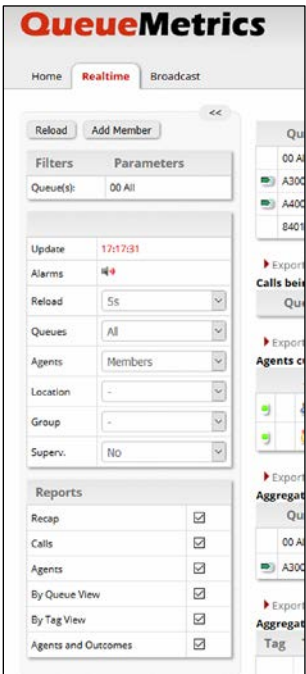
The menu bar can be optionally collapsed clicking on the << button to best fit realtime panel blocks.

Top status panel

The top status panel shows a quick status report for the current situation.

The first line shows information for all selected queues as a sum, while if there is relevant information for a specified queue it is displayed in a separate line. If an alarm is triggered for one of the numeric values displayed, the relevant cell turns either yellow or red.
The displayed fields have the following meanings:

- Queue: The name of the queue. Inbound queues are marked with the symbol while outbound queues use the symbol .
- N. agents: how many agents are logged on to the system, in total
- Ready agents: how many agents are ready to take calls, i.e. are logged on and are not in conversation or on pause
- On pause: how many agents are currently on pause
- Unk: how many agents are currently in conversation, but are not currently known as member of this queue
- Bsy: how many agents who are both members of the given queue and some other queue are currently busy because they are on call on the other queue.
- N. Calls Waiting: how many inbound calls are currently waiting in the selected queue. Outbound queues never have any call waiting.
- On phone inbound: how many agent are talking on the selected inbound queue
- On phone outbound: how many agents are talking on an outbound queue

Please note that, as agents are not linked to a specific queue save for the moment they are actually talking to a caller on the queue, the agent information is computed for all agents on the Asterisk server and not for specific queues, unless the "Members" option in the "Agents" dropdown is selected.

Calls being processed

A list of calls flowing through the selected queues is presented on the middle table. If no call is present the table is displayed empty.

When a call is processed, the following fields are shown: - Queue: the queue that is handling the call; - Caller: The Caller*ID, if available; - Entered: The date and time the call entered the queue system.

If the call is not answered yet, the "Waiting" field is displayed in red and is calculated according to the current date and time of the server. Depending on what type of information is present in the
database (ATTEMPTS or RINGNOANSWER), is possibile to have information about the last agent not picking up the call or the actual ringing agent's phone. The default configuration works with a standard Asterisk configuration and lets able to have RINGNOANSWER information. Please refer to Chapter 18, *Configuring Asterisk for QueueMetrics* [345] for the how to configure Asterisk generate ATTEMPTS information.

When a call is answered, the "Waiting" field tells the time that the caller had to answer; the "Agent" field shows the agent (or terminal) the caller is talking to and the "Duration", in red, is the current call duration.

If the call is ongoing and connected to an agent, moving the mouse on the *wizard* icon at the end of the line, some icons, like Call Monitor and VNC Monitor icons may be present. By clicking on one of these icons you activate the specified monitoring (see below).

If the call is ongoing and you have the special grants to do so, a red scissor icon might appear, moving the mouse on the *wizard* icon, to allow for brute-force call closure. See the section *Closing ongoing calls* for further details.

As soon as a call is completed or hung up, it exits the Calls panel.

This panel can be turned on or off through the "Calls" dropdown on the top of the page.

The "MOH" field shows the amount of time a customer is on hold with music, during a call. If multiple HOLD instances took place during the call, this filed will show the total "on hold" time.

To implement this feature, it is necessary to have an Asterisk patch, which can be obtained by following this link: https://reviewboard.asterisk.org/r/1778/

NOTE: The link to this Asterisk patch is provided by the Copyright holders and Contributors *as is* and any fitness for a particular purpose is Disclaimed. In no event shall Loway be liable for any direct, indirect, incidental, special, exemplary,
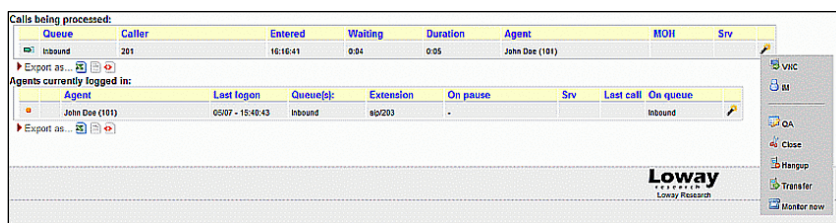
or consequential damages (including Loss of use, data, profits or business interruption) however caused (including negligence or otherwise), arising in any way from the use of this Asterisk patch.

The "Srv" column is used only in cluster-based environments to tell you on which server the call is being handled.

The last field contains a "wand" that on mouse-over displays a drop-down menu which allows to perform the following actions:

- VNC: Monitor agents via a VNC remote session
- IM: Begin an Instant Messaging session with an agent
- QA: Access the QA Form for the specific call (must have the QA_TRACK key enabled)
- Close: Close the call in the QM log, but not in Asterisk (must have the CLOSECALLS key enabled)
- Hangup: Close the live call in the PBX (must have the RT_HANGUPCALL key enabled)
- Transfer: Transfer the call to another extension (must have the RT_TRANSFERCALL key enabled).
- Monitor now: Start a listen-in chanspy Asterisk session (must have the MON_AUDIO key enabled)

To have precise realtime reporting, perfect clock synchronization is required. If your QM is on a different server, make sure the clocks are exactly aligned or you may see strange values in all fields. The NTP protocol offers excellent clock synchronization precision and is available on most operating systems.

Agents currently logged in

A list of available agents for all queues is displayed in this field. For each agent, the name, last log on and extension, if logged in via call-back, is provided.

A graphical indication of the status of each agent is shown using a coloured dot, where the following cases are possible:
- Green dot: the agent is ready to take calls
- Yellow dot: the agent is currently on a call
- Red dot: the agent is currently on pause

This panel can be turned on or off through the "Agents" dropdown on the top of the page.

The *Queue(s)* field shows the queues an agent is logged on to. This is meaningful only for agents who log-in on a queue-by-queue basis using the AddMember command in Asterisk. If an agent logs on to all queues he's enabled to work on, a small database logo may be shown, telling the viewer that the agent is linked to queues through the Asterisk's configuration.

As the queue_log file usually contains no information on what queue(s) an agent is a member of, usually all agents are shown when they log on, no matter to what queue they will work on. This might be a problem for larger call centres, so it is possible to see only calls and log-ons of agents that are a member of the current queue. Membership is set by clicking on the "Agents" button of the queue settings page. Make sure your queue membership data is up-to-date before clicking this button!

If the "All" option was selected in the "Agents" dropdown , all agents logged in on Asterisk will be shown, no matter to which queue they belong. When "Members" is selected, instead, only agents defined for the selected queue will be

displayed.

By turning on the property *"realtime.show_incorrect_queue_sets"*, the Queues field will display each queue with a different color code:
- in black: queues the agent is a known member of
- in brown: queues the agent is currently working on but he's not a known member of
- as a down-arrow: queues the agent is a known member on but he's not currently logged on to. In order to see the list of queues, fly over the down-arrow symbol with your cursor.

The *On Pause* field will contain the time the agent went on pause; if the agent is using a pause code to mark the reason for going on pause, the decoded pause code is shown as well. A switch can be used to show the pause start time either as an absolute hour or a time increment.

The *Srv* field will contain the server identifier for QueueMetrics running in cluster mode. The *Free Since* field will contain the length of period since the last taken call.
The *Last call* and *On queue* fields show the start time of the last call the agent handled (which is latest) and on which queue the last call was. This can be useful to diagnose queue strategy problems that lead to unfair call distribution, or agents having problems with their telephones and therefore not taking calls correctly.

The *IVR* field shows the call time spent on the IVR process before the call enter in the queue.

Waiting time and *Duration* fields are related to the last call taken.

The "wand" icon mouse-over displays a drop-down menu which allows you to perform the following actions:
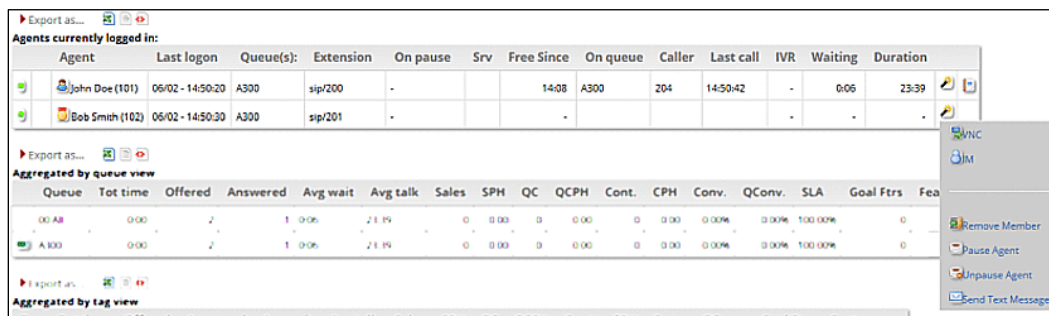
- VNC: Monitor agents via a VNC remote session
- IM: Begin an Instant Messaging session with an agent
- Remove Member: Remove an agent from a queue (must have the RT_REMOVEMEMBER key enabled)
- Pause Agent: Put an agent in a Pause state (must have the RT_PAUSEAGENT key enabled)
- Unpause Agent: Un-pause an agent so he/she is available to take calls (must have the RT_UNPAUSEAGENT key enabled)
- Send Text Message: Send a text message to an agent (must have the RT_SENDTEXTAGENT key enabled). The SMS functionality works only with asterisk 10+ versions.

The last icon is a *direct link to reports*. Clicking on this icon will run an historical report by populating the query with parameters similar to what generated the realtime panel results.

Since QueueMetrics 16.10.17, a new optional field is available for the Agents Currently logged in table. This field is called *Paused For* and it shows the elapsed time of ongoing agent's pauses. To enable this field one needs to enable the realtime.relativePauseTimes System Preference. See Appendix C, *System preferences* [383].

Aggregated by queue view

This block reports realtime figures related to each selected queue. All values are aggregated by queue except for the first row where calls for the whole selected queue set is reported. On each column a user could find:

- *An icon representing the queue direction (inbound/outbound)*
- *The queue name*
- *The total call time*
- *The number of calls feed by the queue*
- *The number of answered calls*
- *Average wait time for all calls in the queue*
- *Average talk time for answered calls*
- *Number of calls marked as sales*
- *SPH: Sales per hour figure*
- *Number of calls marked as qualified contact*
- *QCPH: Qualified contacts per hour figure*
- *Cont: number of calls marked as contact*
- *CPH: Contact per hour figure*
- *Conv: convertion ratio*
- *QConv: qualified conversion ratio*
- *SLA: percentage of calls matching the SLA time configured for the queue*
- *Goal Ftrs: number of goal marked feature codes found in the call set*
- *Feature: number of feature codes found in the call set*

By clicking on the last icon, a user can start an historical report with query parameters matching the realtime page parameters that generated the selected row.

Aggregated by tag view

This block reports realtime figures related to tags associated to processed calls. All values are aggregated by tag except for the first row, where calls for the whole tag set is reported, and the second row, where only calls with no tag are counted.

On each column a user could find:
- Tag name (empty for all call set, dash for calls with no associated tags)
- The total call time
- The number of calls with associated tag
- The number of answered calls with associated tag
- Average wait time for all calls with associated tag
- Average talk time for answered calls with associated tag
- Number of calls marked as sales with associated tag
- SPH: Sales per hour figure
- Number of calls marked as qualified contact with associated tag
- QCPH: Qualified contacts per hour figure
- Cont: number of calls marked as contact with associated tag
- CPH: Contact per hour figure

- Conv: convertion ratio
- QConv: qualified conversion ratio
- Goal Ftrs: number of goal marked feature codes found in the call set
- Feature: number of feature codes found in the call set

By clicking on the last icon, a user can start an historical report with query parameters matching the realtime page parameters that generated the selected row.

Agent and outcome view

This block reports realtime figures related agent sessions and calls. All values are aggregated agent except for the first row, where lost calls are shown.

On each column a user could find:
- Agent status icon
- Agent name
- Agent status explained (Logged on time, pause status, pause time, pause code)
- Offered call number
- Answered call number
- Total talk time
- Total idle time (counted as session time purged from talk and pause time)
- Billable pause time
- Non billable pause time
- Average talk time
- Sales: number of calls markes as sale
- SPH: Sales per hour figure
- Number of calls marked as qualified contact
- QCPH: Qualified contacts per hour figure
- Cont: number of calls marked as contact
- CPH: Contact per hour figure
- Conv: convertion ratio
- QConv: qualified conversion ratio
- Goal Ftrs: number of goal marked feature codes found in the call set
- Feature: number of feature codes found in the call set

By clicking on the last icon, a user can start an historical report with query parameters matching the realtime page parameters that generated the selected row.

Using Locations
You can also assign each agent to al Location, i.e. a group of agents working together that you want to monitor as a unit. They might be a supervisor's team, or people working in the same building, or maybe in the same location for big multinational call-centres. This way you can avoid being cluttered with information about all agents working on the selected queue(s) and only see those you are actually interested in.

Note that when you filter by agents or locations you may see calls being queued and then disappear. This happens because all calls that wait to be answered on a queue are shown, but answered calls are shown only if the answering agents is a member of your defined filter conditions.

Imagine you have two groups of agents, one working in NY and the other one in LA. You are the supervisor of the NY group, so you are filtering by location. You see a call entering your queue and then disappear. The reason why might be

that it was answered by someone in LA, so QueueMetrics filters it out for you.

It is also possible to have QM ask for a location to be selected in any case for Real-time reports. As locations can be key-protected, this is a handy way to determine which agents can be seen by which supervisors. In order to enable this feature, you must set the *realtime.assignedLocationsOnly* property. When this is set:
- When a user enters the real-time screen, the dash that means "Any location" is not shown anymore and the first location they have the grants to see is be selected.
- If a user has no grants to see at least one location, an error message is raised and they are not able to enter the real-time screen
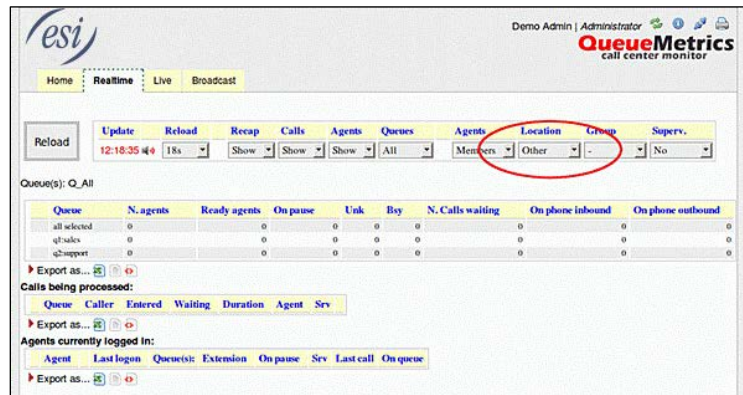
Unattended call and VNC monitoring
It is possible for you to set up both Unattended Call Monitoring and VNC Monitoring.

Unattended Call Monitoring makes it possible to listen to an ongoing call from an agent; by clicking on the small telephone-shaped icon, that will appear moving the mouse under the *wizard* icon, a popup will be shown where you may enter your current extension or PSTN telephone number.

As soon as you confirm the entered data, your telephone will ring and you will start listening to the ongoing call between the selected agent an the caller.

In order to set up this behaviour, please see the section the section called "Listening to live calls: Unattended Call Monitoring" [350]. It is possible to use different PMs to handle different audio needs
- see the section called "Listening to calls using Pluggable Modules (PM)" [276].

It is also possible to set up the system in order to allow the real-time monitoring of the agent's screen using VNC. If this feature is enabled, a small screen will appear close to the agent's name; by clicking on it, your selected VNC application will be launched and you will be monitoring the agent's screen.

In order to set up this behaviour, please see the section the section called "Enabling VNC Monitoring" [351] of this manual.

Real-time performance considerations

Running a large set of real-time pages / wallboards may impose a significant burden on a QueueMetrics system. But sometimes you have no choice. The best way to run them is to:

- Use SQL or CLUSTER storage types
- Set the beginning of the analysis to the minimum usable time. For example, QM usually runs the real-time report since the last midninght, but sometimes a sliding window of (say) 8 hours might be equally effective in your scenario and require a fraction of the load. You control this by setting the realtime.startHour property
- Turn on row caching - this allows QM to keep cached and pre-processed copies of objects it needs to run the Real-time and Agents Page. This may offer a significant speed up, ofter by an order of magnitude, and will reduce database load. You turn on this feature by setting realtime.useRowCache to true.
- Tune Java memory for maximum performance - see the **Advanced Configuration** manual.

The real-time live page

The real-time live page can be accessed by clicking on the "Live" tab next to the "Realtime" tab.
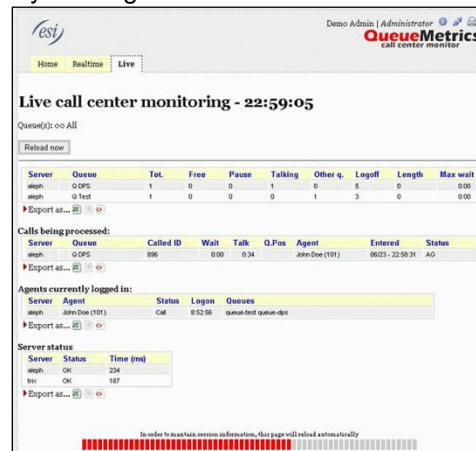
This page is not built from the queue_log data as all other information reported by QueueMetrics, but it' s read right from each Asterisk box's Manager interface. So what you see in this page is the status of each Asterisk box, as reported by itself.

This feature is still less developed than the Realtime page, but still can be pretty useful.

The top panel

On the top panel, for each queue on each server, you will see the following pieces of information:

- Tot: the total number of agents available for this queue
- Free: the number of free agents
- Pause: the total number of agents on pause
- Talking: the total numbers of agents who are in conversation at the moment
- Other q.: the number of agents that are logged in to this queue and some other queue, and are at the moment busy on another queue.
- Logoff: the number of possible agents that are defined for this queue but are not logged on at the moment.
- Length: the current queue length, i.e. how many calls are waiting in line before being connected to an agent

- Max wait: the current maximum wait time for this queue.

Calls being processed
In the calls panel you see the following pieces of information, sorted from oldest to newest by call start- time:

- Server: the server that is handling this call
- Queue: the name of the queue
- Caller-id: the caller-id of this call, if any
- Wait: the wait time (if the call is not connected)
- Talk: the total duration (if the call is connected)
- Q.Pos: the queue position (if the call is waiting)
- Agent: the agent handling this call, if connected
- Entered: the time this call was queue
- Status: the call status
- Info: contains all the Feature Codes, Variables, Call Tags, IVR choices if present. The Wait and Talk times cannot be distinguished at the moment.

Agents currently logged in
In the agents panel you can see the following pieces of information:
- Server: the server your agent is logged on to
- Agent: the agent
- Status: if the agent is free, paused or on call
- Logon: the time this agent logged on
- Queues: to which queues is this agent connected at the moment

Server status
The last panel details the status of each server making up the cluster. If a server is not correctly set up, it will appear as KO.
- Server: the server that QM is polling
- Status: OK - the server answered correctly; KO - it was impossible to retrieve information from this server
- Time: how much querying this server took. If this value goes up all of a sudden, your server is likely experiencing overload.

Chapter 4.The Wallboard

QueueMetrics includes user-definable wallboard facilities that allow you to design your own wallboards and run them either interactively or as proper stand-alone wallboards.
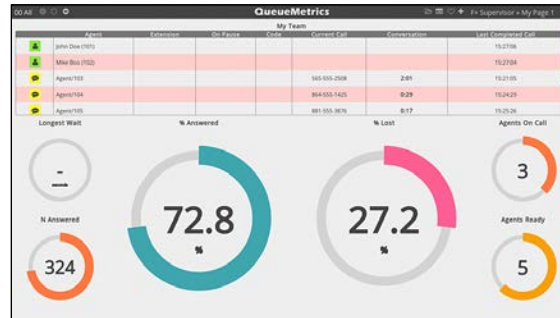
Wallboards are built as a series of rotating **views**, each of which includes a number of **widgets**. Widgets are positioned as items on a square grid that is meant to fill the screen, and you can control the number of columns that you want to use.

Wallboards are meant to be customized - you can change colors, set alarms on specific widgets, include text and images and in general customize them to suit your specific needs. Wallboards can be private or public, in which case they can be shared between different users and can be used to run automated stations (see "Kiosk mode" below).

Understanding wallboards
A wallboard is controlled through the top bar. The top bar:
- Displays the **current queue** or queues that are being used as data sources.
- Displays a number of circles, one for each **view**. By clicking on a widget, you can jump to a specific view. The current view is displayed as a dotted circle.
- Displays a **Play/Pause** button that lets you decide whether views rotate in order according to their expected durations or the current view remains visible. If you do any change to the page, rotation stops automatically until manually reactivated.
- Displays a **Load** button that lets you load other wallboards that are visible to you or create a new, blank wallboard.
- If the wallboard has unsaved data (e.g. because you adde some widgets or moved them around), a **Save** button apperas in the top bar that lets you save the current wallboard. Switching view when you have unsaved data will lose any unsaved data.
- There is a **Reorder** button that lets you change the sequence of views in the current wallboard and add/remove views.
- There is a heart-shaped icon that displays whether the wallboard you are looking at is **public or private**. By clicking on that icon within a public wallboard, you get the URL that you need to run the wallboard in kiosk mode copied to your clipboard.
- There is an **Add Widget** button that lets you add new widgets to the current view.
- There is a **title**, showing the name of the current wallboard and of the current view. By clicking on it, you can customize some properties for both wallboard and view.
- There is an **Exit** button that gets you back to your QueueMetrics session.

In order to access the wallboard, you need the security key *REALTIME*. To edit a wallboard you need the key *WALLBOARD_ADD* and to create a public wallboard you need the key *WALLBOARD_FORALL*. You can easily add those keys though the User editor or by adding them to a security class.

Saved Wallboards
The default wallboard is accesible though the link "Wallboard" on the home page of QueueMetrics, using the current queue(s) selected. From the wallboard itself you can load other wallbords though the **Load** button.

Your wallboards, and all public wallboards visible to you, will appear on the home page in the "Wallboards" section. By clicking on them, they will open with the current selected queue. You can also delete existing wallboards by clicking on the "Remove" icon hovering over their home page link.

Settings
By clicking on the title of the wallboard, you can edit the following properties:

For the whole wallboard:
- **Name** that the wallboards appears under. You can have multiple wallboards with the same name
- **Comment** is a free text comment you may want to save
- **Public** controls whether the wallboard appears on every user's home page (as restricted by the security key) or is private for you
- **Security key** lets you protect this wallboard from users who do not hold the specified key

For the current view:
- **Name** is the name of the current view. Multiple views within a wallbord may have the same name, though it would be better to use different names
- **Duration** is the duration in seconds that a wallbard is displayed on the screen before the next wallboard. Data is updated live when a wallboard is being displayed.
- **Number of Columns** is the size of the grid that the screen is horizontally split into. This is the minimum size for a widget, though you can stretch a widget to use multiple grid cells as you best see fit.

Editing Widgets
Widgets can be added, moved, cloned or removed freely within a view. You should position a widget in a way that makes sense for your use case, giving it enough room to display its data in a way that is easy to see.

Most widgets will resize automatically in the space they are given to offer a good readability and to make full use of the space they are assigned.

If you can edit a widget, when you fly over it with your mouse it will display an **Edit widget** icon and it will go in editor mode. Though the editor you can:
- Clone the widget
- Remove the widget
- Change the title
- …and edit any special configuration it has (see below for a description of all availabe widgets)

As of QueueMetrics 17.06.11, most of the widgets can be filtered by queue. If you are using the wallboard to monitor multiple queues at the same time, you can use the queue filtering function to apply the widget to a single queue instead of applying it to the aggregation of all the queues data.

Whey you save a widget, the whole wallboard is saved automatically.

Alarms
Most widgets have a set of possible **custom alarms** that you can use to display them in a different color when an alarm condition is met. Custom alarms are expressed as simple expressions, e.g. ">10" or "<15".

If QueueMetrics has a system alarm set on the queue shown and no custom alarm set, then the correct system alarm is used. Custom alarms always take precedence on system alarms.

Running in kiosk mode
If you want to run a wallboard in kiosk mode, the first thing that you need is to get its kiosk URL. You can obtain this by clicking on the "heart" icon. The URL might appear like the one below:

http://127.0.0.1:8080/QM/qm_wab2.do?user=robot&pass=*** &queues=300.301&wallboardId=21

As you can see, you can edit which (atomic) queues you want to use for your wallboard by separating them with a pipe "|" character.

The URL will always include a user called *robot* with password *****; edit it as necessary given the current settings of your "robot" user. Make sure that the *robot* user is enabled, as it is usually present but disabled on new QueueMetrics installs.

If you connect a small-cost Linux box (e.g. a Raspberry PI or similar) to a large screen or a video beamer and set it in your call-center where it will be visible by your agents, you can set-up a wallboard at a very low cost using commodity hardware and requiring no human intervention but turning it on in the morning and turning it off at night.

Component Details
Here is an overview of the components currently available in Queuemetrics' Wallboard.

Circular Panels
PC01 - Waiting Calls
This Component diplays how many calls are currently waiting to be answered.

The following properties can be modified:
- **Title**: The title of the component
- **Color**: The color of the stroke
- **Red alarm**: The expression used to trigger a **red alarm** (eg. >20)
- **Yellow alarm**: The expression used to trigger a **yellow alarm** (eg. >20)
- **Queue**: The queue you want to filter the data for

Available since	17.06
Name	Waiting calls
Shortcut code	PC01
Parameters	-

PC02 - Lost Calls
This Component diplays the number of lost calls.

The following properties can be modified:
- **Title**: The title of the component
- **Color**: The color of the stroke
- **Red alarm**: The expression used to trigger a **red alarm** (eg. >20)

- **Yellow alarm** The expression used to trigger a **yellow alarm** (eg. >20)
- **Queue**: The queue you want to filter the data for

Available since	17.06
Name	Lost calls
Shortcut code	PC02
Parameters	-
See also	

PC03 - Answered Calls
This Component diplays the number of answered calls.

The following properties can be modified:
- **Title**: The title of the component
- **Color**: The color of the stroke
- **Red alarm**: The expression used to trigger a **red alarm** (eg. >20)
- **Yellow alarm** The expression used to trigger a **yellow alarm** (eg. >20)
- **Queue**: The queue you want to filter the data for

Available since	17.06
Name	Answered calls
Shortcut code	PC03
Parameters	-
See also	

PC04 - Offered Calls
This Component diplays the number of offered calls.

The following properties can be modified:
- **Title**: The title of the component
- **Color**: The color of the stroke
- **Red alarm**: The expression used to trigger a **red alarm** (eg. >20)

- **Yellow alarm** The expression used to trigger a **yellow alarm** (eg. >20)
- **Queue**: The queue you want to filter the data for

Available since	17.06
Name	Offered calls
Shortcut code	PC04
Parameters	-

PC05 - Longest Wait
This Component diplays the duration of the longest waiting time currently happening on the selected queues.

The following properties can be modified:
- **Title**: The title of the component
- **Color**: The color of the stroke
- **Red alarm**: The expression used to trigger a **red alarm** (eg. >20)
- **Yellow alarm** The expression used to trigger a **yellow alarm** (eg. >20)
- **Queue**: The queue you want to filter the data for

Available since	17.06
Name	Longest Wait
Shortcut code	PC05
Parameters	-
See also	

PC06 - Agents On Call
This Component diplays the number of agents on call on the selected queues.

The following properties can be modified:
- **Title**: The title of the component
- **Color**: The color of the stroke
- **Red alarm**: The expression used to trigger a **red alarm** (eg. >20)
- **Yellow alarm** The expression used to trigger a **yellow alarm** (eg. >20)

- **Queue**: The queue you want to filter the data for

Available since	17.06
Name	Agents On Call
Shortcut code	PC06
Parameters	-
See also	

PC07 - Lost Calls %
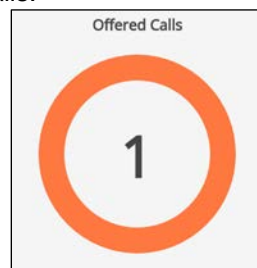This Component diplays the percentage of lost calls.

The following properties can be modified:
- **Title**: The title of the component
- **Color**: The color of the stroke
- **Red alarm**: The expression used to trigger a **red alarm** (eg. >20)
- **Yellow alarm** The expression used to trigger a **yellow alarm** (eg. >20)
- **Queue**: The queue you want to filter the data for

Available since	17.06
Name	Lost Calls %
Shortcut code	PC07
Parameters	-
See also	

PC08 - Answered Calls %
This Component diplays the percentage of answered calls.
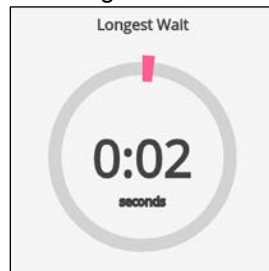

The following properties can be modified:
- **Title**: The title of the component
- **Color**: The color of the stroke
- **Red alarm**: The expression used to trigger a **red alarm** (eg. >20)
- **Yellow alarm** The expression used to trigger a **yellow alarm** (eg. >20)
- **Queue**: The queue you want to filter the data for

Available since	17.06
Name	Answered Calls %
Shortcut code	PC08
Parameters	-
See also	

PC09 - Agents Ready

This Component diplays the number of agents ready on the selected queues.

The following properties can be modified:
- **Title**: The title of the component
- **Color**: The color of the stroke
- **Red alarm**: The expression used to trigger a **red alarm** (eg. >20)
- **Yellow alarm** The expression used to trigger a **yellow alarm** (eg. >20)
- **Queue**: The queue you want to filter the data for

Available since	17.06
Name	Agents Ready
Shortcut code	PC09
Parameters	-
See also	

PC10 - SLA Level %

This Component diplays the percentage of calls answered within the selected SLA level.
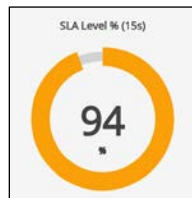
The following properties can be modified:
- **Title**: The title of the component
- **SLA Level**: The SLA time interval
- **Color**: The color of the stroke
- **Red alarm**: The expression used to trigger a **red alarm** (eg. >20)
- **Yellow alarm** The expression used to trigger a **yellow alarm** (eg. >20)

Available since	17.06
Name	SLA Level %
Shortcut code	PC10
Parameters	-
See also	

Tables
TC01 - Live Agents
This Component diplays a table showing all the agents logged in on the selected queues.

The following properties can be modified:
- **Title**: The title of the component
- **Pause Red alarm**: The expression used to trigger a **red alarm** on the pause time of an agent (eg. >20)
- **Pause Yellow alarm** The expression used to trigger a **yellow alarm** on the pause time of an agent (eg. >20)
- **Call Red alarm**: The expression used to trigger a **red alarm** on the conversation time of an agent (eg. >20)
- **Call Yellow alarm** The expression used to trigger a **yellow alarm** on the conversation time of an agent (eg. >20)
- **Queue**: The queue you want to filter the data for
  The table will also be responsive to any **Red** or **Yellow alarm** that has been set in the **Edit Queues** page under the following fields:
- **Call talking duration**

Available since	17.06
Name	Live Agents
Shortcut code	TC01
Parameters	-

TC02 – Live Calls
This Component diplays a table showing all the calls in progress on the selected queues.

The following properties can be modified:
- **Title**: The title of the component
- **Wait Red alarm**: The expression used to trigger a **red alarm** on the waiting time of a call (eg. >20)
- **Wait Yellow alarm:** The expression used to trigger a **yellow alarm** on the waiting time of a call (eg. >20)
- **Call Red alarm**: The expression used to trigger a **red alarm** on the conversation time of a call (eg. >20)
- **Call Yellow alarm** The expression used to trigger a **yellow alarm** on the conversation time of a call (eg. >20)
- **Queue**: The queue you want to filter the data for
  The table will also be responsive to any **Red** or **Yellow alarm** that has been set in the **Edit**

Queues page under the following fields:

- Call wait duration
- Call talking duration

Available since	17.06
Name	Live Calls
Shortcut code	TC02
Parameters	-

TC03 - Live Queues

This Component diplays a table showing all the selected queues, and their relative information.

The following properties can be modified:

- **Title**: The title of the component
- **Agents On Queue Yellow Alarm**: The expression used to trigger a **red alarm** on the number of agents on queue (eg. >20)
- **Agents On Queue Red Alarm**: The expression used to trigger a **red alarm** on the number of agents on queue (eg. >20)
- **Agents Ready Yellow Alarm**: The expression used to trigger a **red alarm** on the number of agents ready (eg. >20)
- **Agents Ready Red Alarm**: The expression used to trigger a **red alarm** on the number of agents ready (eg. >20)
- **Agents On Pause Yellow Alarm**: The expression used to trigger a **red alarm** on the number of agents on pause (eg. >20)
- **Agents On Pause Red Alarm**: The expression used to trigger a **red alarm** on the number of agents on pause (eg. >20)
- **Agents Busy Yellow Alarm**: The expression used to trigger a **red alarm** on the number of busy agents (eg. >20)
- **Agents Busy Red Alarm**: The expression used to trigger a **red alarm** on the number of busy agents (eg. >20)
- **Agents Unknown Yellow Alarm**: The expression used to trigger a **red alarm** on the number of agents in an unknown status (eg. >20)
- **Agents Unknown Red Alarm**: The expression used to trigger a **red alarm** on the number of agents in an unknown status (eg. >20)
- **Waiting Calls Yellow Alarm**: The expression used to trigger a **red alarm** on the number of waiting calls (eg. >20)
- **Waiting Calls Red Alarm**: The expression used to trigger a **red alarm** on the number of waiting calls (eg. >20)
- **Inbound Agents Yellow Alarm**: The expression used to trigger a **red alarm** on the number of inbound agents (eg. >20)
- **Inbound Agents Red Alarm**: The expression used to trigger a **red alarm** on the number of inbound agents (eg. >20)
- **Outbound Agents Yellow Alarm**: The expression used to trigger a **red alarm** on the number of outbound agents (eg. >20)
- **Outbound Agents Red Alarm**: The expression used to trigger a **red alarm** on the number of outbound agents (eg. >20)
- **Queue**: The queue you want to filter the data for
 The table will also be responsive to any **Red** or **Yellow alarm** that has been set in the **Edit Queues** page under the following fields:
  o Number of agents waiting
  o Number of agents paused
  o Number of agents on call

- o Number of calls in queue

Available since	17.06
Name	Live Queues
Shortcut code	TC03
Parameters	-
See also	

Items
IT01 - N. of Calls Offered
This Component diplays a text item showing the number of calls offered on the selected queues.

N Calls Offered

1742

Num. of Offered Calls

The following properties can be modified:
- **Title**: The title of the component
- **Red alarm**: The expression used to trigger a **red alarm** on the number of offered calls (eg. >20)
- **Yellow alarm**: The expression used to trigger a **yellow alarm** on the number of offered calls (eg. >20)
- **Color**: The color scheme for the background and text.
- **Queue**: The queue you want to filter the data for

Available since	17.06
Name	N. of Calls Offered
Shortcut code	IT01
Parameters	-

IT02 - Longest Wait
This Component diplays a text item showing the longest wait time among the ongoing calls on the selected queues.

Longest Wait

0:03

Queue b

The following properties can be modified:
- **Title**: The title of the component
- **Red alarm**: The expression used to trigger a **red alarm** on the longest wait value (eg. >20)
- **Yellow alarm**: The expression used to trigger a **yellow alarm** on the longest wait value (eg. >20)
- **Color**: The color scheme for the background and text.
- **Queue**: The queue you want to filter the data for

Available since	17.06
Name	Longest Wait
Shortcut code	IT02
Parameters	-
See also	

IT03 - N. of Calls Lost

This Component diplays a text item showing the number of calls lost on the selected queues.

The following properties can be modified:
- **Title**: The title of the component
- **Red alarm**: The expression used to trigger a **red alarm** on the number of answered calls (eg. >20)
- **Yellow alarm**: The expression used to trigger a **yellow alarm** on the number of answered calls (eg. >20)
- **Color**: The color scheme for the background and text.
- **Queue**: The queue you want to filter the data for

Available since	17.06
Name	N of Calls Lost
Shortcut code	IT03
Parameters	-

IT04 - N. of Calls Answered

This Component diplays a text item showing the number of calls answered on the selected queues.

The following properties can be modified:
- **Title**: The title of the component
- **Red alarm**: The expression used to trigger a **red alarm** on the number of answered calls (eg. >20)
- **Yellow alarm**: The expression used to trigger a **yellow alarm** on the number of answered calls (eg. >20)
- **Color**: The color scheme for the background and text.
- **Queue**: The queue you want to filter the data for

| Available since	17.06
Name	N of Calls Answered
Shortcut code	IT04

Parameters	-
See also	

Composite Items

MI02 - Live Agent Info

This Component is composed of three sections, each detailing a different information about an agent.

- The **agent's name** is shown in the **uppermost part** of the component.
- The **agent's state** is shown in the **middle part** of the component. To represent the agent's status there are different status badges. To read a description of the status you can hover on the status badge. Together with the status badge you will see a time counter, detailing how long the agent has been in that particular state.
- The **bottom part** of the component will show the **name of the queue** the agent is receiving a call from, if the agent is in a BUSY status. If the agent is on PAUSE, the **pause name** will be shown.
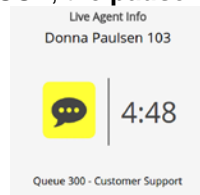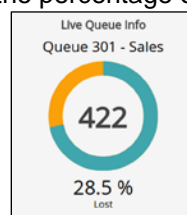

The following properties can be modified:

- **Title**: The title of the component
- **Pause Red alarm**: The expression used to trigger a **red alarm** on the pause time of an agent (eg. >20)
- **Pause Yellow alarm**: The expression used to trigger a **yellow alarm** on the pause time of an agent (eg. >20)
- **Call Red alarm**: The expression used to trigger a **red alarm** on the call time of an agent (eg. >20)
- **Call Yellow alarm**: The expression used to trigger a **yellow alarm** on the call time of an agent (eg.>20)
- **Idle Red alarm**: The expression used to trigger a **red alarm** on the idle time of an agent (eg. >20)
- **Idle Yellow alarm**: The expression used to trigger a **yellow alarm** on the idle time of an agent (eg. >20)

Available since	17.06
Name	Live Agent Info
Shortcut code	MI02
Parameters	-

MI03 - Live Queue Info

This Component is composed of three sections, each detailing a different information about a queue.

- The **queue's name** is shown in the **uppermost part** of the component.
- The **middle part** of the component is comprised of a circular panel detailing the number of **answered calls** on the selected queue.
- The **bottom part** of the component will show the percentage of **lost calls** on the selected queue.
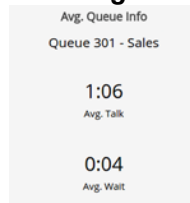
The following properties can be modified:
- **Title**: The title of the component
- **Red alarm**: The expression used to trigger a **red alarm** on the number of answered calls (eg. >20)
- **Yellow alarm**: The expression used to trigger a **yellow alarm** on the number of answered calls (eg. >20)
- **Color**: The color of the circular panel's stroke.
- **Secondary Color**: The secondary color for the circular panel's stroke.

Available since	17.06
Name	Live Queue Info
Shortcut code	MI03
Parameters	-

MI04 - Average Queue Info
This Component is composed of three sections, each detailing a different information about a queue.
- *The **queue's name** is shown in the **uppermost part** of the component.*
- *The **middle part** of the component shows the **average talk** value for the selected queue.*
- *The **bottom part** of the component will show the **average wait time** for a call on the selected queue.*
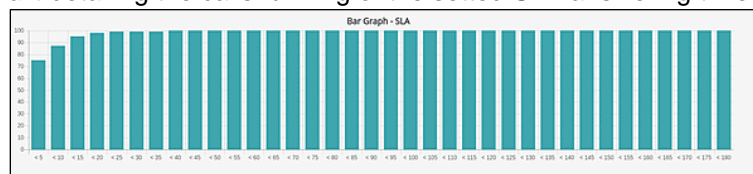

The following properties can be modified:
- **Title**: The title of the component
- **Talk Red alarm**: The expression used to trigger a **red alarm** on the average talk time (eg. >20)
- **Talk Yellow alarm**: The expression used to trigger a **yellow alarm** on the average talk time (eg. >20)
- **Wait Red alarm**: The expression used to trigger a **red alarm** on the average wait time (eg. >20)
- **Wait Yellow alarm**: The expression used to trigger a **yellow alarm** on the average wait time (eg. >20)
- **Color**: The color of the circular panel's stroke.
- **Secondary Color**: The secondary color for the circular panel's stroke.

Available since	17.06
Name	Average Queue Info
Shortcut code	MI04
Parameters	-

Charts
BC01 - Bar Graph SLA
This Component is a **Bar Chart** detailing the calls fulfilling of the setted SLA answering time requirements.

The following properties can be modified:
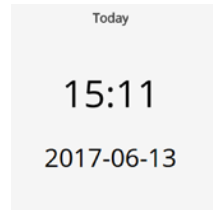- **Title**: The title of the component

- **Color**: The color of the circular panel's stroke.
- **Yellow Line Limit**: A numerical value between 0-100 that set's the Y coordinates for a yellow limit line.
- **Red Line Limit**: A numerical value between 0-100 that set's the Y coordinates for a red limit line.

Available since	17.06
Name	Bar Graph SLA
Shortcut code	BC01
Parameters	-

Other
CL01 - Clock
This Component is a Clock component showing the current Date and Time.

The following properties can be modified:
- **Title**: The title of the component

Available since	17.06
Name	Clock
Shortcut code	CL01
Parameters	-

TX01 - Text
This Component is a Text component showing a string of text. The string of text can be formatted in MD markup format, allowing the user to display images, gifs and links.

The following properties can be modified:
- **Title**: The title of the component
- **Text**: The text to display

Available since	17.06
Name	Text
Shortcut code	TX01
Parameters	-
See also	