

Understanding queue end-points in WombatDialer



An end-point of type queue in Wombat tries to determine the number of available channels based on the number of available agents on the queue it is observing. An agent is considered available if it is logged on to the queue, is not paused and is not currently in conversation.

After getting the number of available agents, it multiplies it by the “boost factor” (that is used to account for the success rate in connecting the numbers to be dialed) and tries to schedule as many calls. Therefore, if you have e.g. 7 available agents and a boost factor of 1.3, it will try to connect to 9 numbers at once ($7 \times 1.3 = 9.1$).

It will also enforce two limits:

- If there are more than “max waiting calls” waiting on a queue, it will not try and place new calls - in theory there should never be calls queued, as they follow the number of available agents, but it is possible that either some agent logs off after being counted to place calls or some calls reach a queue without passing through Wombat. This also acts as a counterbalance to high “boost factor” values.
- it will never place more than “max channels” calls on the queue - this lets you use a shared queue where some calls come from inbound activity and some come from Wombat itself. Of course you can turn this off by setting “max channel” to a value higher than the total number of agents on the queue.

It is also important to notice that the queue is used only as a way to know how many calls can be placed on a queue, but calls will still be transferred to a point in the dial-plan that should lead to the queue. This lets you execute dial-plan logic (e.g. playing pre-recorded messages or running IVRs) on the calls it just placed.

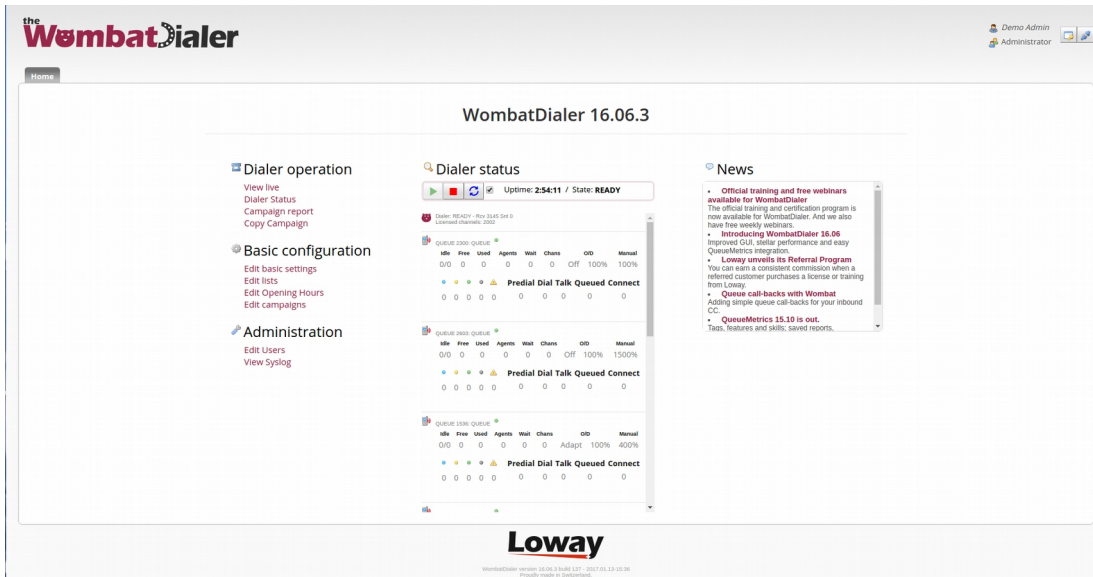
In order to use Wombat effectively with a queue, the following guidelines are best followed:

- though Wombat would work with static member channels, if you want your calls to go through to agents who may or may not be available (e.g. some days they may be sick) it is strongly advisable to use dynamic agents who log on and off from the queue.
- as an agent cannot be physically available at all times during the day, it is important that they have a way to pause themselves, be it to run “wrap up” activities after calls or to take breaks. The QueueMetrics web interface offers an excellent panel that lets you add pause codes as well
- the queue must provide events to Wombat about agent activities. Therefore you must set “eventswhencalled=true” - otherwise the queue will be unobservable. It is also important that extension presence is correctly observed - e.g. if an end-point is busy because the agent is doing a personal call, its queue status should immediately reflect this. Whether this happens or not on your system is a matter of Asterisk version and type of channel that is used to reach the agent - with recent versions of Asterisk and SIP channels this should work automatically.
- the queue should connect calls to agent as efficiently as possible when there are multiple calls waiting and multiple available agents, so it should have the “autofill” option set to true.

In order to run a campaign with a Queue endpoint, it is best to:

- create a campaign with a Queue endpoint. You may create it IDLE and add no call lists, so the campaign does not actually do anything until fed some numbers.
- run that campaign
- reload the Dialer status in order to see if the queue is being observed (you have to click on the reload icon manually each time). If the queue is present, you should see it something saying “Free 4 of 7 W:2”. This means that Wombat is seeing 7 agents connected of which 4 are free (where 4 is the result of multiplying the actual number of observed channels by its boost factor), and that there are 2 calls waiting on the queue.

- try and log on, log off, pause and unpaue an agent. You should see the number of free and available channels. Try also sending calls to the queue and see if the number of agents and of waiting calls is correct.
- try also placing calls from some agent extensions and see if the number of free channels reflects this correctly.



The Queue functionality of Wombat lets you use Wombat as a powerful progressive dialer and can lead to very complex integration scenarios, but as it involves actual people being called and answering the phone, it is better to understand it well before actually using it in production.