

## Quassel IRC - Feature #616

### Limit Backlog Retrieval Time

03/10/2009 03:18 AM - hagabaka

<b>Status:</b>	New	<b>Start date:</b>	03/10/2009
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>		<b>% Done:</b>	0%
<b>Category:</b>		<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>			
<b>OS:</b>	Any		

#### Description

Problem:

1. Retrieving backlog can take a long time, and during this time quassel is not "usable".
2. Client-core connection sometimes dies during backlog retrieval.
3. Although the amount of backlog to retrieve is configurable, this configuration scales with the number of open buffers.

Suggestion:

1. Have a time limit for retrieving backlog, and when the time is out, stop retrieving more backlogs, and start "operating" (display the buffer tree, showing current messages, etc).

Or

2. Have a limit of lines of backlog retrieved globally, which is divided between buffers.

Details:

I run Quassel core and client on the same computer, and every time when I reattach the client to a running core, retrieving the backlog takes about 30 seconds. During this time I cannot use the client, (messages flow in the chat monitor, but the buffer trees are not displayed completely, and chat area is gray), and what's worse, frequently the connection between client and core just dies during this phase. I think that backlog is less essential than current conversations for most people, and retrieving it should not take so long to make the user wait before being able to use the client.

I can configure Quassel to retrieve fewer lines of backlog per buffer to alleviate the problem, but the configuration is relative to the number of buffers I have. So if I open more channels next time, I would need to lower the setting again. Limiting the number of buffers might be another solution to the problem. I don't know whether Quassel retrieves backlog only for buffers listed in the buffer views; I usually part a channel and not manually delete the buffer, but just use a buffer view that hides them.

So there should be a configurable timeout with a short default value to retrieve backlog, and when that expires, the client should just start "working" without trying to retrieve more backlog. Alternatively, there should be a configurable limit for the total lines of backlog retrieved for all buffers.

#### History

##### #1 - 12/22/2009 01:36 PM - DevUrandom

Having the same problem, I presented an alternative approach in [#885](#).

##### #2 - 06/14/2010 03:07 PM - sdancer

This is still an issue, especially with large databases.

Looking at a Postgres log, it should be possible to make backlog fetching far more responsible, since it creates a lot of small queries (at least one for each channel), each taking around a few seconds on my virtual root server package. So if the server could start sending backlog to a client as soon as it arrived from the database instead of retrieving everything before sending it out, it should work far better.

Right now, I cannot connect to my core at all when trying to get more than maybe 10 lines of backlog per buffer, since the core will take minutes to send many dozen queries to the server. The backlog table is at a bit under 3.5 million rows.