

## Quassel IRC - Bug #42

### Make Core multi-user capable and clean up architecture

06/18/2007 11:01 AM - Sputnik

<b>Status:</b>	Closed	<b>Start date:</b>	
<b>Priority:</b>	High	<b>Due date:</b>	
<b>Assignee:</b>	Sputnick	<b>% Done:</b>	0%
<b>Category:</b>	Quassel Core	<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>	0.2.0-alpha1	<b>OS:</b>	Any
<b>Version:</b>	0.13.1		
<b>Description</b>			
<p>As planned for quite a long time, a quassel core should be able to handle multiple users simultaneously in order to, for example, allow shared hosting solutions. This also includes user authentication (BR <a href="#">#17</a>).</p> <p>As it turns out, this involves major changes in the Core architecture, an opportunity which I will use to also clean up things a bit. Things that come to mind:</p> <ul style="list-style-type: none"><li>- Core functionality needs to be split into global and user-specific parts (classes Core and CoreSession)</li><li>- Global data (class Global) needs to be handled per-user</li><li>- CoreProxy needs to be user-specific, so I'll transform it from a pseudo-singleton to a proper object, with one instance per user</li><li>- Connection handling moves from CoreProxy to Core</li><li>- User auth will be part of Core (BR <a href="#">#17</a>)</li></ul> <p>A question that comes to mind: Do we ever <i>end</i> a user session? Even if all GUIs disconnect, the session should continue running. What happens if core executions ends - should we save active sessions and restart them when the core is restarted, or should we rather wait for a GUI to log in and explicitly get a session started for the particular user? An admin mode comes to mind, where active sessions can be monitored (stats) and sessions started/stopped...</p> <p>If we plan to automatically restore sessions, does this involve reconnecting to all nets that where connected at shutdown, and joining all channels? Or should we only reconnect/join stuff that is marked as auto-connect and auto-join in the appropriate preferences?</p>			
<b>Related issues:</b>			
Related to Quassel IRC - Bug #17: Implement authentication for GUI/Core		<b>Closed</b>	

### Associated revisions

#### Revision 7ec4585c - 06/19/2007 02:50 AM - Manuel Nickschas

Big update this time - Core has been redesigned to be multi-user capable. At least partly. (BR #42)

- Architecture changes mostly complete:
  - Core has been split in Core (static) and CoreSession (per-user objects)
  - Connection stuff has been moved out of CoreProxy into Core
  - Lots of cleanups, and we have finally real singletons!
  - Global is now (externally) a static class -- changes throughout the code
- Monolithic Quassel works (or at least it seems to)
- Standalone Core and GUI disabled for now:
  - While most of the new infrastructure is in place in Core, we still need GUI stuff for auth (BR #17)
  - Syncing with Core as well as getting session states not done yet

Next steps will include the redesign of the GUI to make that cleaner as well (BR #41)

### History

#### #1 - 07/08/2007 04:52 PM - Sputnik

Done. Has not been tested with more than one user though, since we don't have auth yet.