

postgresql-snap - Task #5338

Explore app alias functionality

06/02/2017 06:43 AM - Ivan Lezhnjov

Status:	Resolved	Start date:	06/02/2017
Priority:	Normal	Due date:	06/09/2017
Assignee:	Ivan Lezhnjov	% Done:	100%
Category:		Estimated time:	0.00 hour
Target version:		Spent time:	0.00 hour
Description			
<p>App aliases may allow us to use psql instead of postgresql96.psql.</p> <p>However, it is unclear how aliases will work if several different PostgreSQL version are installed alongside each other.</p> <p>This needs to be explored and tested thoroughly to understand if app aliases are of any benefit.</p> <p>Ivan</p>			

History

#1 - 06/06/2017 10:17 AM - Ivan Lezhnjov

- Status changed from New to In Progress
- % Done changed from 0 to 90

Aliases were initially brought up by Canonical and the documentation they referenced states that in order to use aliases one needs to provide them in snapcraft.yml as aliases: entity and rebuild packages.

However, that is not necessary (anymore?). Users can set up aliases manually however they want. Additionally, aliases must be unique and thus enforcing them (which doesn't even seem to work) is not a good idea. If someone needs to install and run more than one version of PostgreSQL side by side I suspect this won't work at all.

This is further complicated by the fact that auto-aliases need to be set up on request in snapcraft forum. It means we can't manage or test them easily and this needs to happen essentially in production. Not to mention that in upcoming 2.26 version which is said to be released soon current behavior is set to change.

I started a couple of discussions on snapcraft forum in order to better understand how aliasing really works.

For now it looks like we don't need to define them in snapcraft.yml and rebuild packages as this will only complicate things unnecessarily.

Ivan

#2 - 06/06/2017 10:24 AM - Ivan Lezhnjov

BTW, this is about <https://github.com/commandprompt/postgresql-snap/pull/44>

Ivan

#3 - 06/09/2017 02:25 AM - Ivan Lezhnjov

- Status changed from In Progress to Resolved

- % Done changed from 90 to 100

Aliases implementation changed indeed since David Calle originally suggested we should use aliases in our packages.

On the one hand, users can create aliases manually without package maintainer's permission today. And, on the other hand, auto-aliasing is cumbersome to manage (via requests posted in snapcraft's forum), is prone to conflicts and may easily lead to unnecessary confusion on user's part.

For example, if we configure aliases for all snap packages and a user wants to install 2 or more PostgreSQL snap packages, the second (third and so on) will simply fail to install when trying to enable aliases that would be in conflict with aliases for the first package.

There are ways to work around this, but it requires prior user knowledge of how to handle aliases, which may not exist.

I'd rather not enable aliases at all. postgresql93.psql syntax also helps to stay clear about which version of psql is being called when working with multiple versions of PostgreSQL, including APT version.

If the aliases implementation changes in the future, and specifically how aliases conflicts are handled during installation time, we may want to enable them. But right now it seems like we would be better off without app aliases.

Ivan