Toolforge webservices are in the final stages of migrating to the toolforge.org domain.
Please help us clean up older documentation referring to tools.wmflabs.org!

Main page
Recent changes
Server admin log (Prod)
Server admin log (RelEng)
Deployments
SRE/Operations Help
Incident status

Cloud VPS & Toolforge
Cloud VPS documentation
Toolforge documentation
Request Cloud VPS project
Server admin log (Cloud VPS)

Tools
What links here
Related changes
Special pages
Permanent link
Page information
Cite this page

Print/export
Create a book
Download as PDF
Printable version

Incident documentation/20150930-enwiki-readonly

< Incident documentation

Summary

Yesterday (Sept 29th) Chad H was poking at a fairly old bug that asks for cleanup of superfluous rows in user_properties, specifically the subtask of removing old skin entries that just take up space and end up falling back to Vector (our default) anyway.

After completing the work for all small and medium wikis (per the dblists), Chad began running some sample queries against enwiki to see how the cleanup would scale. Obviously, it's a much larger table. Chad then attempted to delete 16 rows from the table--what appeared to be low hanging fruit to test the queries on a large table--but it took about 2m9s to execute on the master.

The query plan for this delete did not involve the PK, so it ended up selecting about 17m rows to find which ones to delete. I immediately kept an eye on the replag, which did climb to about 3 minutes depending on the slave. Replag began to recede as the query finished but it did happen long enough for icinga to start noticing and enwiki to (very briefly) go into read-only mode.

As a result: Chad stopped all further work on his cleanup and made sure people knew it was him that caused the issues with S1.

Around this point is also when people reported that the small and medium wikis lost their monobook skin preferences. The cause looks to be a bug in the cleanupSkinPrefs code.

After discussion between Chad, Jaime, and Greg it was decided to not restore from the DB. It was deemed to be more disruptive than leaving things as they were at that point. The state of those db rows had changed since the incident, some portion of users had already changed some or all of their personal settings back).

> Extensive discussion in the following 2 weeks: phab:T114208
> Users who want their skin preference set across all wikis are asked to look at phab:T119206

Conclusions

- Cleanup of databases is not very useful and mostly disruptive. At a minimum, it should only be conducted under the supervision of the DBA.
- Cleanup/migration processes need a clear rollback plan in case they go wrong. Doing so would've prevented this.
- Batch scripts should always always use PKs. (see phab:T17441, others)

Actionables

Explicit next steps to prevent this from happening again as much as possible, with Phabricator tasks linked for every step.

- Status: **Done** - Remove cleanupSkinPreferences. It is a dangerous script with no benefit. - Gerrit 242374
- Status: **Done** - Add changeSkinPref maitenance script to change a user's skin preference across all wikis. - Gerrit 254407
- Status: **in-progress** - Restore individual user's previous skin preferences, as requested (until December 21st, 2015). - phab:T119206

Category: Incident documentation

This page was last edited on 21 November 2015, at 00:22.

Text is available under the Creative Commons Attribution-ShareAlike License; additional terms may apply. See Terms of Use for details.