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/20150914-extension1

< Incident documentation

Contents [hide]
1 Summary
2 Timeline
3 Conclusions
4 Actionables

Summary

On 2015-09-14 from about 09:47 UTC until 09:59 UTC, the x1 master was saturated with slow queries caused by a missing index. This caused all Flow boards to be inaccessible: trying to view a Flow board displayed a database error message.

Timeline

- 09:43 Jaime runs the last ALTER statements for a complex schema change for Flow
- 09:47 Matthias notices that every Flow page view dies with `(Cannot access the database: Too many connections (10.64.16.18))`
- 09:51 Jaime figures out this is because all queries are now unindexed: the ALTERs dropped the indexes that were being used
- 09:56 Roan retrieves the old index definitions from git history
- 09:59 Jaime recreates the old indexes; Flow comes back up immediately

Conclusions

The commit message of the change contained detailed instructions about how to carry out the migration. It instructed that patch3.sql (setting the field to NOT NULL and dropping the old indexes) be run before turning off `$wgFlowMigrateReferenceWiki`. However, this was incorrect: as long as `$wgFlowMigrateReferenceWiki` was turned on, the old indexes would be used, so the correct order was to create the new indexes, turn off the setting, then remove the old indexes. This mistake was missed in code review and missed again by the Flow devs who were present when the schema change was executed. The other statements in phase3.sql (setting the fields to NOT NULL, and DROP INDEX statements for temporary indexes that were added later) were appropriate to run at that stage, further hiding the incorrectly placed DROP INDEX statements.

Actionables

- Do we have (or should we develop) something that tells us which indexes are used and which aren't? That would have allowed us to verify each index was unused before dropping it.

  Create a debug mode for MW that blows in your face when an unindexed query is run and enable it by default in Vagrant and betalabs? maxsem (talk) 20:06, 18 September 2015 (UTC)

- ...?
- Better monitoring for connections reaching max_connections and query performance: bug T112473 bug T99485

Category: Incident documentation

This page was last edited on 1 July 2016, at 18:01.

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