



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!

Incident documentation/20141114-EventLogging

[< Incident documentation](#)

Contents [\[hide\]](#)

- [1 Summary](#)
- [2 Timeline](#)
- [3 Conclusions](#)
- [4 Actionables](#)

Summary

Event volume grew beyond what EventLogging's database writer could handle, and not all events could get written to the database between 2014-11-14T01:00 and 2014-11-21T21:00

Timeline

2014-11-14T01:00

With the lightning deploy at that hour, changes got merged that increase the total count of messages from ~140 msg/s to ~220 msg/s during busy hours.

From now on, during hours with higher traffic (~10:00--24:00), the EventLogging database writer failed to bring all events to the database. Extrapolating from a few high volume schemas, it seems that during peak hours only about 70% of the events made it to the database.

Throughput monitoring on the EventLogging processors did not go off as it was configured for the event validation bottle-neck (which can handle more messages than the database bottle-neck).

2014-11-19T11:41

Gilles sent [notice](#) to the analytics mailing list about a suspicious decrease in MediaViewer actions.

2014-11-20T21:00

Ori deployed a fix for EventLogging's database writer.

Conclusions

- Having multiple EventLogging issues at the same time does not help when trying to find root causes :-(
- Teams are not to trust when asking whether or not an increase in numbers is expected. We need to be able to walk the deployment lists on our own and find out what code the teams deployed when and what numbers they should expect. This is frustrating, as it explodes the code base we need to understand and be able to cover.
- Monitoring throughput only at EventLogging's "front-end" processors is not sufficient. We need monitoring that checks whether all events could get written to the database during peak hours.
- We need to re-evaluate the monitoring thresholds for EventLogging's throughput monitoring. They are tuned to the volume that event validation can take. But here the bottle-neck of the database writer became relevant before the event validation bottle-neck became relevant.

Actionables

- Status: ■ **Done** Make sure that log files exist for the affected period, so we can backfill.
- Status: ■ **Done** Backfill the data.
- Status: ■ **Unresolved** Implement monitoring that makes sure that the expected volume of events end up in the database.
- Status: ■ **Done** Re-evaluate the throughput monitoring's thresholds.

Categories: [Incident documentation](#) | [EventLogging/Incident documentation](#)

[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](#)

This page was last edited on 2 December 2014, at 13:38.

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

[Privacy policy](#) [About](#)

[Disclaimers](#) [Code of Conduct](#) [Developers](#) [Statistics](#) [Cookie statement](#) [Mobile view](#)

[Wikitech](#)

