



Dashboard: Transfers latency monitoring

Alexandre Beche

- Requirements
- Proposal
- User Interface presentation



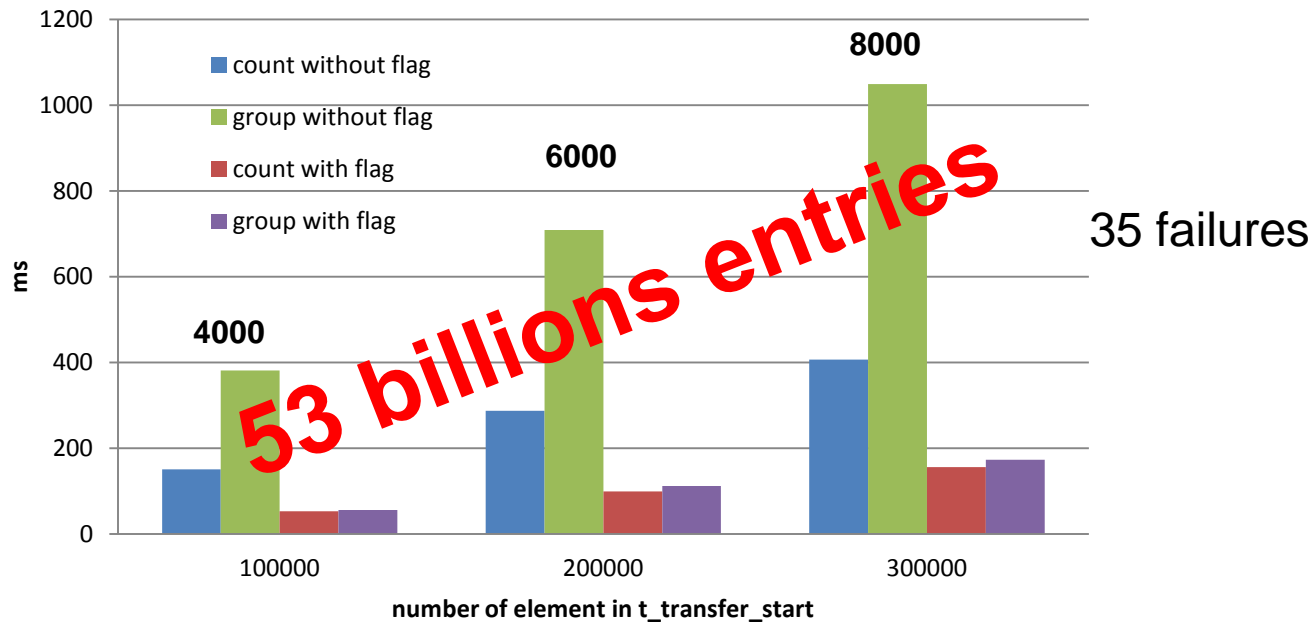
- ATLAS observed slow transfers:
 - 2 disk servers misbehave
 - Few transfers blocked the full channel
- Request:
 - checking the FTS jobs that were lasting for more than N hours

- 2 interesting tables:
 - t_transfer_start
 - t_transfer_complete
- A full scan is required to find the running transfers:
 - Table partitions useless
 - Querying time grows exponentially

- Add a new flag in t_transfer_start table:
 - Is_completed
 - Updated when a transfer is finished
 - A “flat line” is expected in the number of running transfers

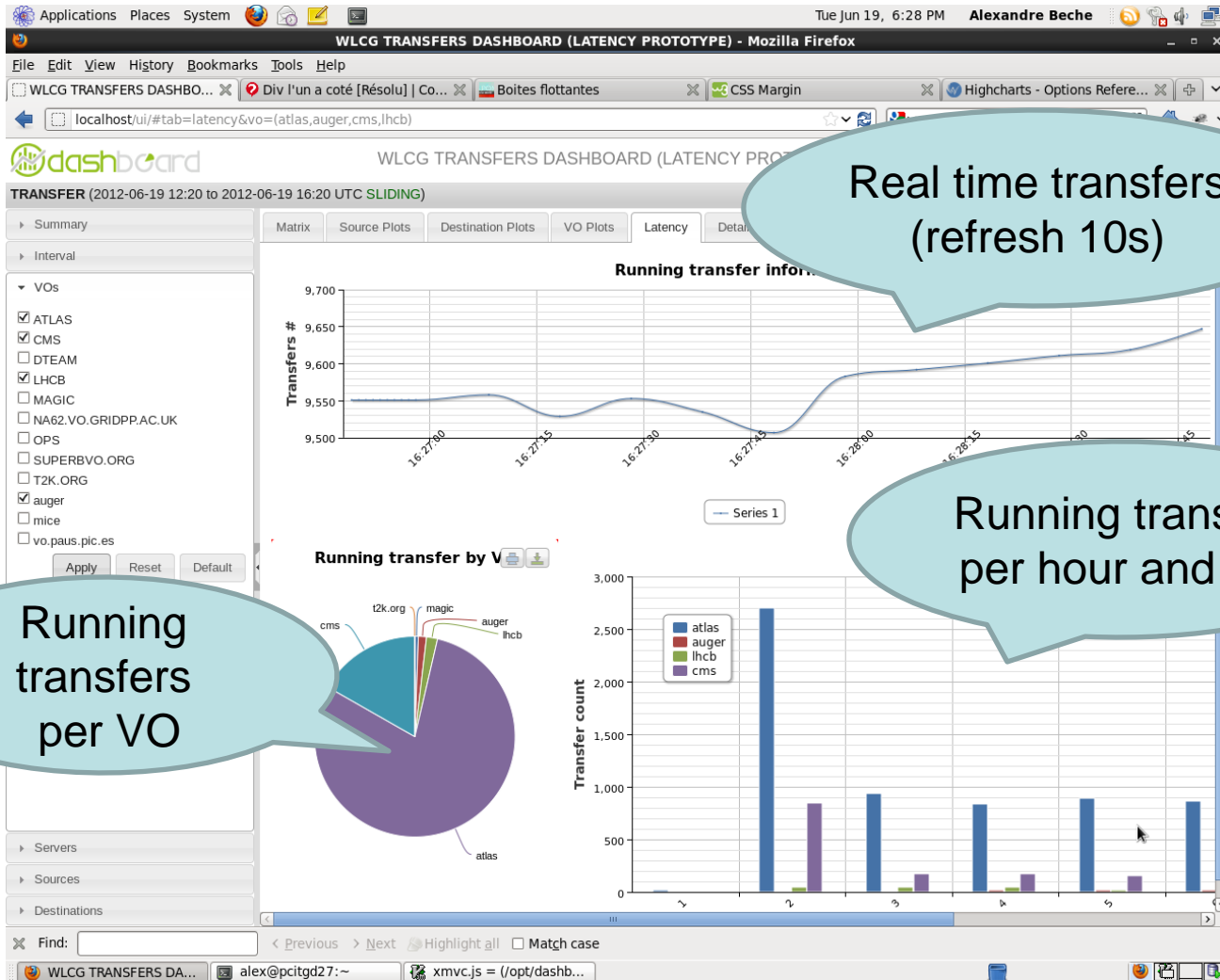
- DB:
 - Schema of t_transfer_start
- New function in the DAO
 - setCompletedFlag (Called by the collector)
- More investigation are required:
 - Impact of the DB lock when updating

Time to query the running transfers



Does a simple flag enough ???
Yes - if row indexed

- Two new tabs:
 - 1 Generic for global information
 - 1 Detailed for list of transfers (like in the DDM2)
- Global information:
 - Real time chart with number of running transfers
 - Pie chart with running transfers per VO
 - Column chart with running transfers per starting hour and VO
- Detailed information:
 - List of every single running transfers



The screenshot displays the WLCG Transfers Dashboard (Latency Prototype) in Mozilla Firefox. The interface includes a navigation menu, a summary section for the last 4 hours, a line graph showing 'Transfers #' over time, a pie chart titled 'Running transfer by VOs' with segments for atlas, magic, t2k.org, and auger, and a table of transfer details. A blue arrow points from the pie chart to the table. The table has columns for transfer id, timestamp, and channel. The bottom of the screen shows a bar chart and system tray information.

