2018-2

DATABASE ADMINISTRATION ADVANCED

Institución Universitaria
Acreditada en Alta Calidad

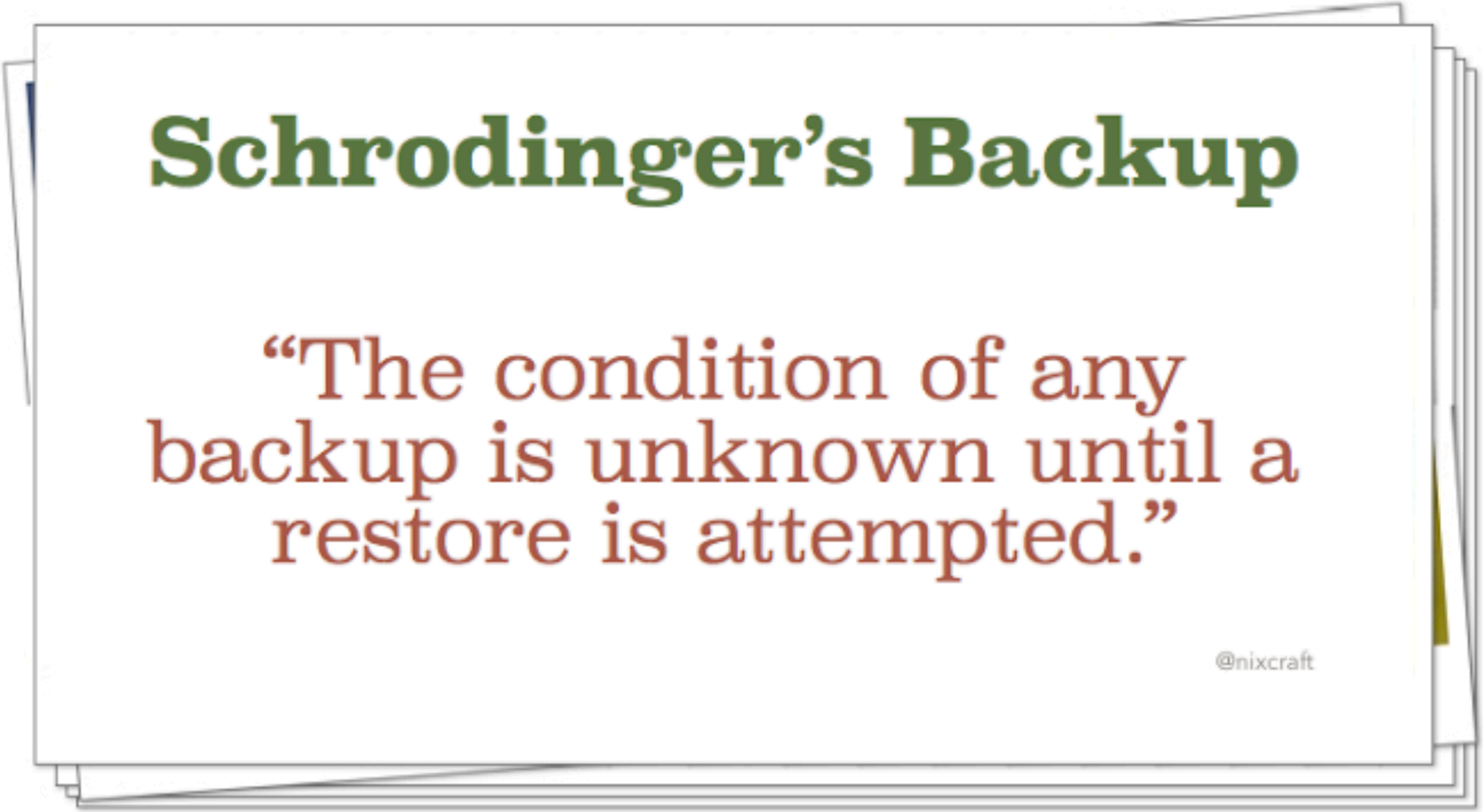
ANDRES MARTINEZ GUTIERREZ

AGENDA

- Gitlab Study Case
- RMAN

Gitlab Study Case

https://about.gitlab.com/2017/02/01/gitlab-dot-com-database-incident/
https://about.gitlab.com/2017/02/10/postmortem-of-database-outage-of-january-31/



Schrodinger's Backup

"The condition of any backup is unknown until a restore is attempted."

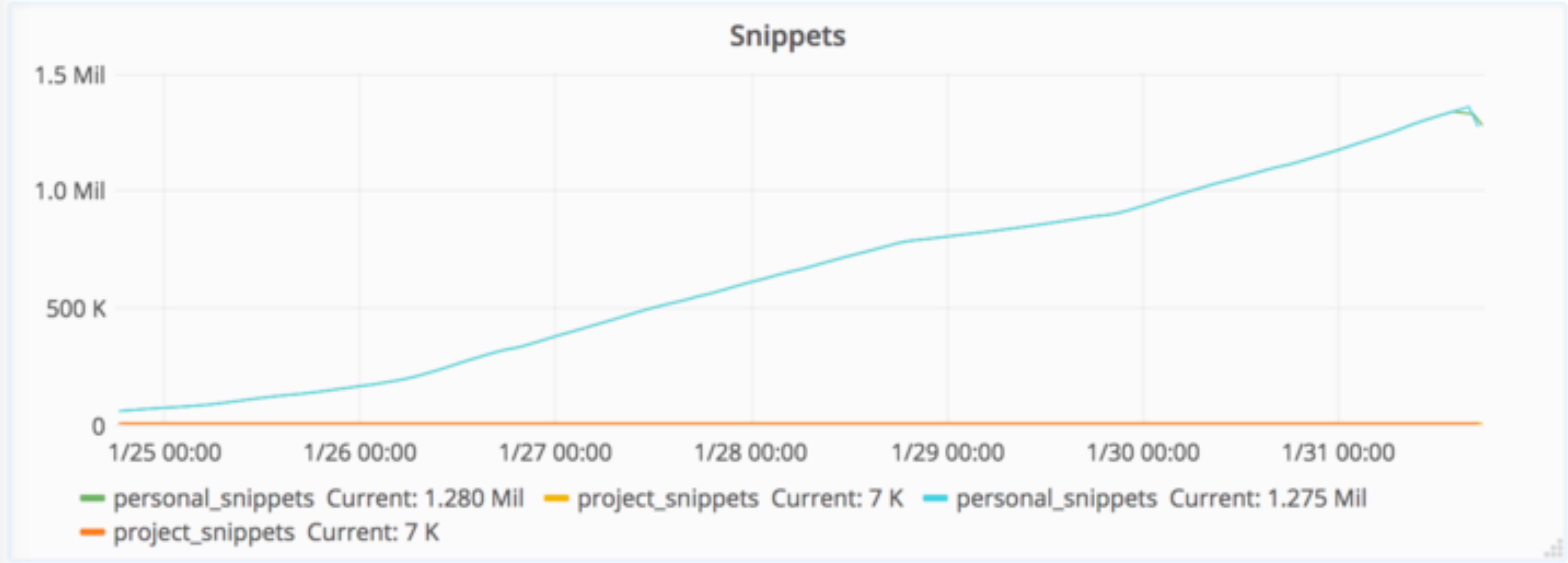
@nixcraft

GitLab.com Database Incident

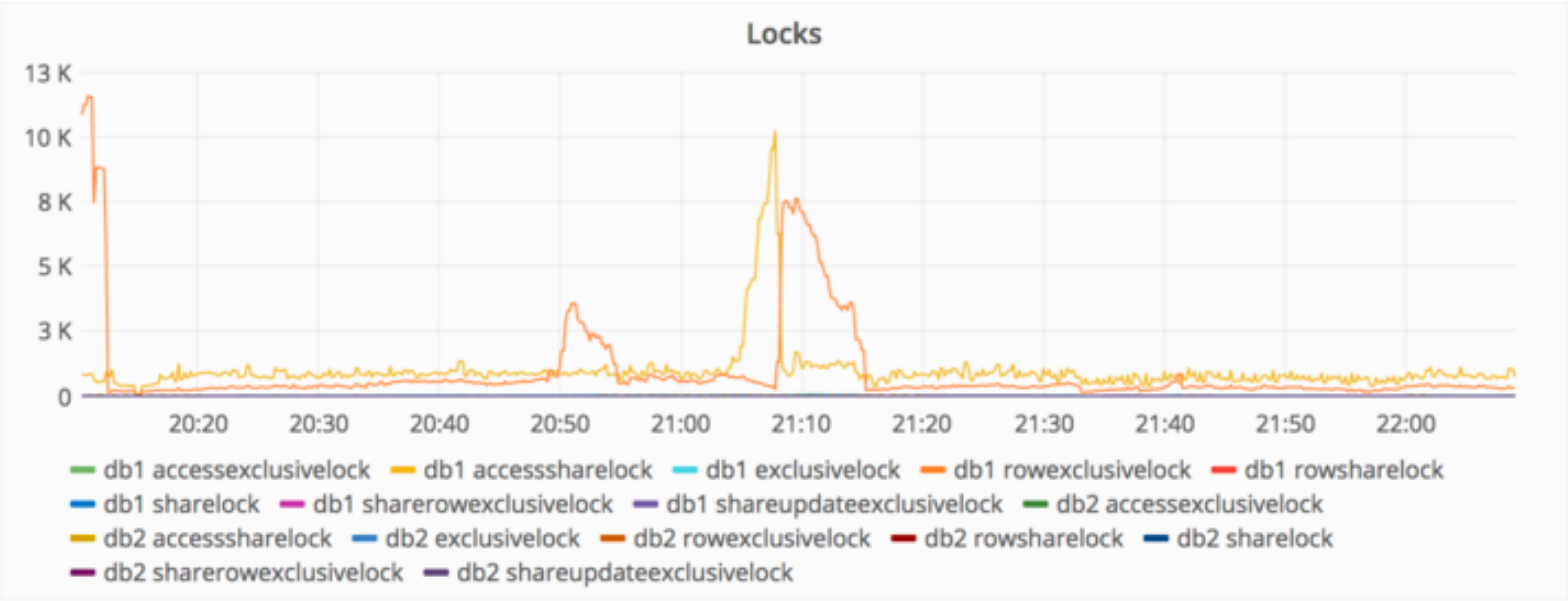
- *"Yesterday we had a serious incident with one of our databases.* ***We lost six hours of database data*** *(issues, merge requests, users, comments, snippets, etc.) for GitLab.com."*

Feb 1st / 2017

GitLab.com. First incident (6pm UTC)



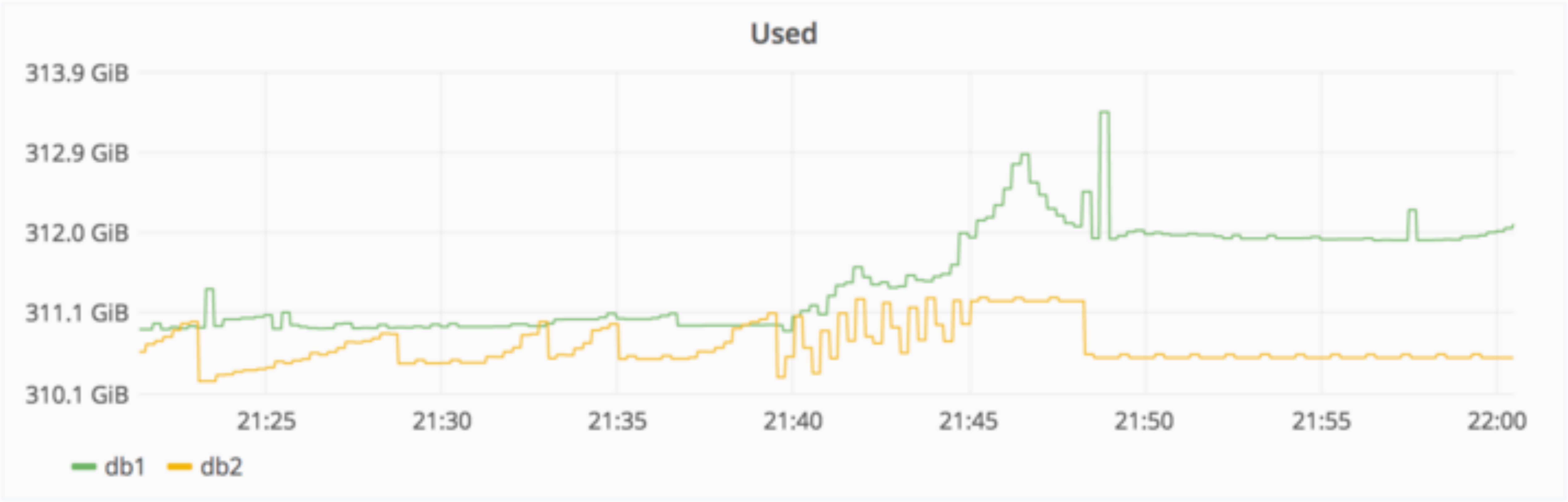
GitLab.com. First incident (6pm UTC)



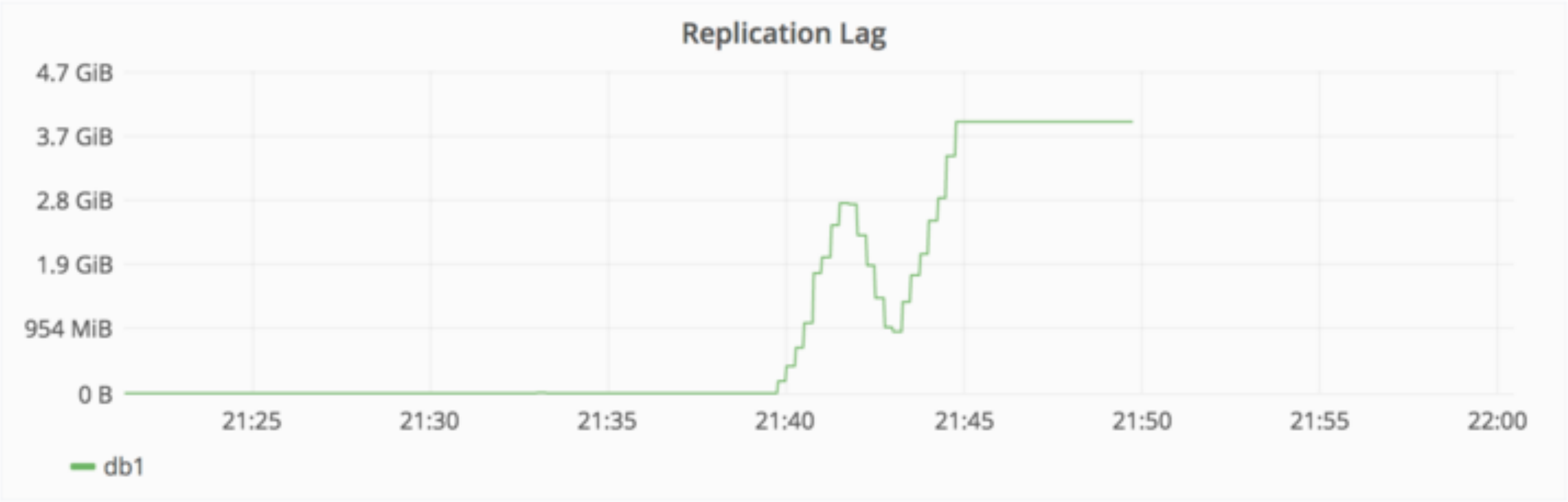
Actions Taken

- “Blocked the spammers based on IP address”
- “Removed a user for using a repository as some form of CDN, resulting in 47 000 IPs signing in using the same account”
- “Removed users for spamming”

GitLab.com. second incident (10pm UTC)



GitLab.com. second incident (10pm UTC)



Actions Taken

- fix db2, it’s lagging behind by about 4 GB
- db2.cluster refuses to replicate, */var/opt/gitlab/postgresql/*data is **wiped** to ensure a clean replication
- db2.cluster refuses to connect to db1, complaining about **max_wal_senders** being too low. This setting is used to limit the number of WAL (= replication) clients

WAL? = Write ahead log

- Servers will be either a Master or a Standby server. Masters can send data, while Standby(s) are always receivers of replicated data

max_wal_senders

- Maximum number of concurrent connections from standby servers or streaming base backup clients

Actions Taken

- Team-member-1 adjusts max_wal_senders to 32 on db1, restarts PostgreSQL
- PostgreSQL complains about too many semaphores being open, refusing to start

Actions Taken

- Team-member-1 adjusts max_connections to 2000 from 8000, PostgreSQL starts again (despite 8000 having been used for almost a year)
- db2.cluster still refuses to replicate, though it no longer complains about connections; instead it just hangs there not doing anything

Actions Taken

- At this point frustration begins to kick in. Earlier this night team-member-1 explicitly mentioned he was going to sign off as it was getting late (23:00 or so local time), but didn’t due to the replication problems popping up all of a sudden

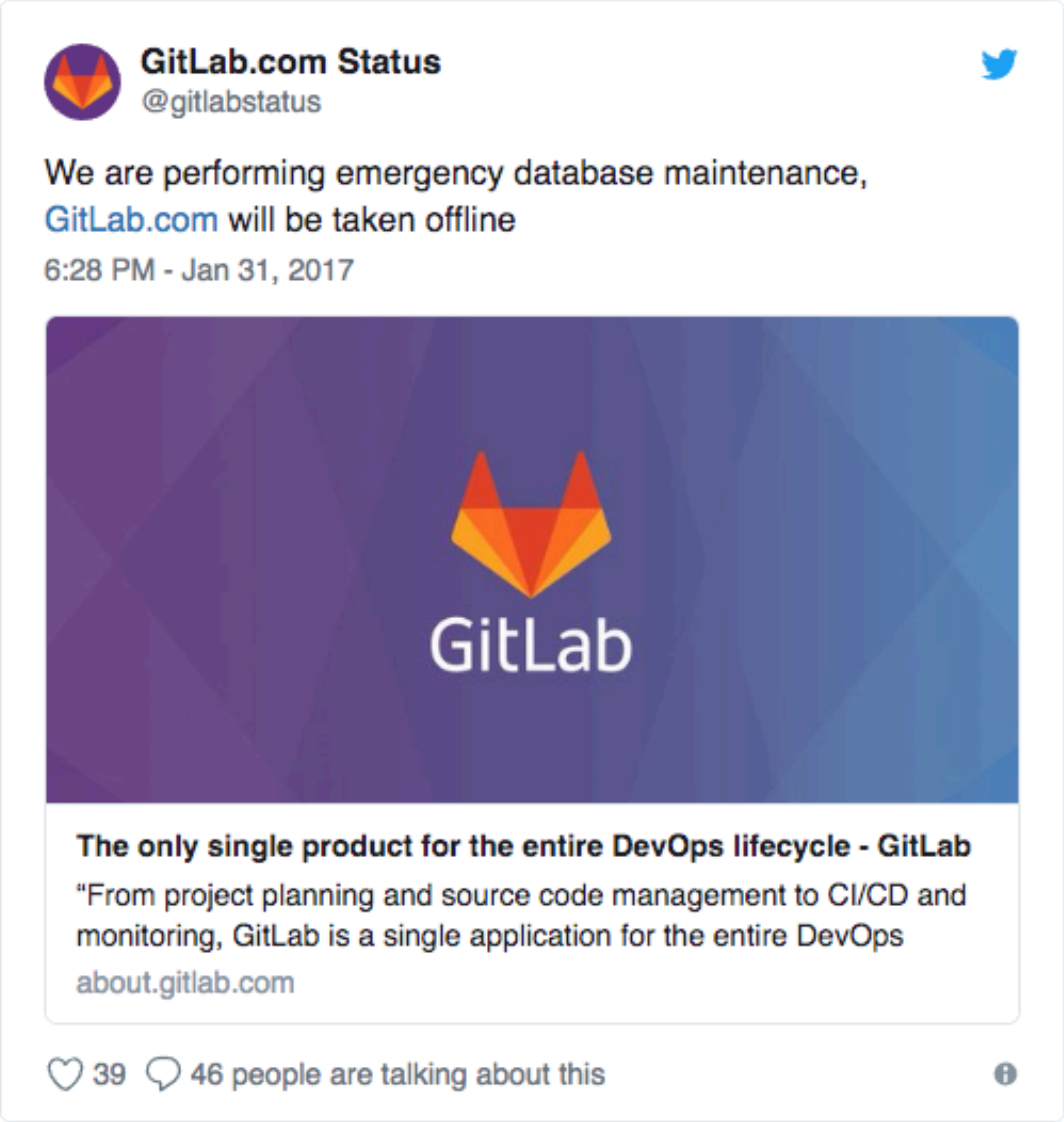
Gitlab.com. Third Incident (11 pm-ish UTC)

- Team-member-1 thinks that perhaps pg_basebackup is refusing to work due to the PostgreSQL data directory being present (despite being empty)

  - Run it on **db1**.cluster.gitlab.com
  - Instead of **db2**.cluster.gitlab.com

Gitlab.com. Third Incident (11 pm-ish UTC)

11:27 UTC. **300Gb** were removed about 4.5 left



GitLab.com Status
@gitlabstatus

We are performing emergency database maintenance, GitLab.com will be taken offline

6:28 PM - Jan 31, 2017

The only single product for the entire DevOps lifecycle - GitLab

"From project planning and source code management to CI/CD and monitoring, GitLab is a single application for the entire DevOps

about.gitlab.com

39 46 people are talking about this

Gitlab.com. Problems encountered

- LVM snapshots are by default only taken once every 24 hours (Team-member-1 run one manually 6 hours before)

LVM? = Logical Volume Manager

- The Linux Logical Volume Manager (LVM) is a mechanism for virtualizing disks. It can create "virtual" disk partitions out of one or more physical hard drives, allowing you to grow, shrink, or move those partitions from drive to drive as your needs change. It also allows you to create larger partitions than you could achieve with a single drive.

LVM? = Logical Volume Manager

- Traditional uses of LVM have included databases and company file servers, but even home users may want large partitions for music or video collections, or for storing online backups. LVM and RAID 1 can also be convenient ways to gain redundancy without sacrificing flexibility

Gitlab.com. Problems encountered

- Regular backups seem to also only be taken once per 24 hours:
  - Team-member-1 does not know where they are stored
  - Team-member-2 thinks is not working

Gitlab.com. Problems encountered

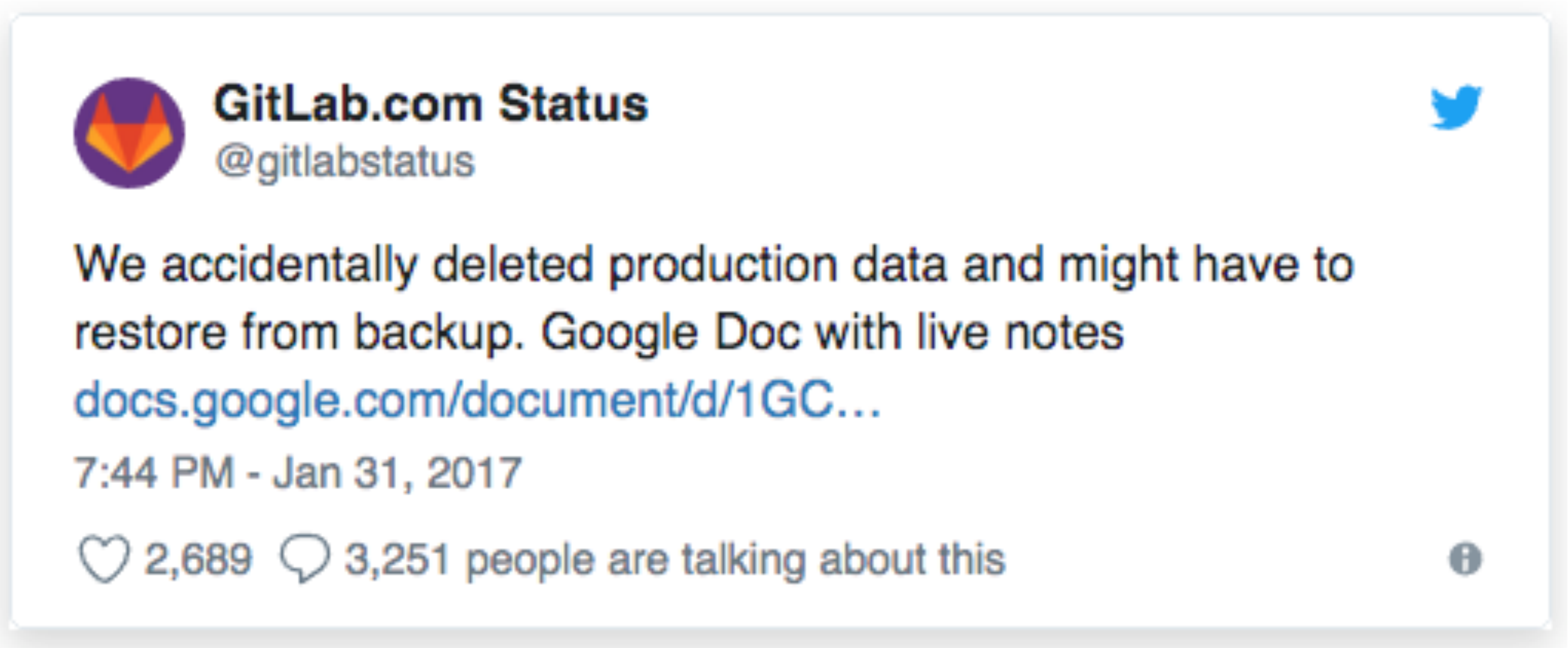
- Team-member-3 “It looks like pg_dump may be failing because PostgreSQL 9.2 binaries are being run instead of 9.6 binaries”
- Disk snapshots in Azure are enabled for the NFS server, but not for the DB servers.

Gitlab.com. Problems encountered

- The replication procedure is super fragile, prone to error, relies on a handful of random shell scripts, and is badly documented
- Our backups to S3 apparently don’t work either: the bucket is empty

Gitlab.com. Problems encountered

- So in other words, ***out of five backup/replication techniques deployed none are working reliably or set up in the first place***. We ended up restoring a six-hour-old backup.



GitLab.com Status
@gitlabstatus

We accidentally deleted production data and might have to restore from backup. Google Doc with live notes docs.google.com/document/d/1GC…

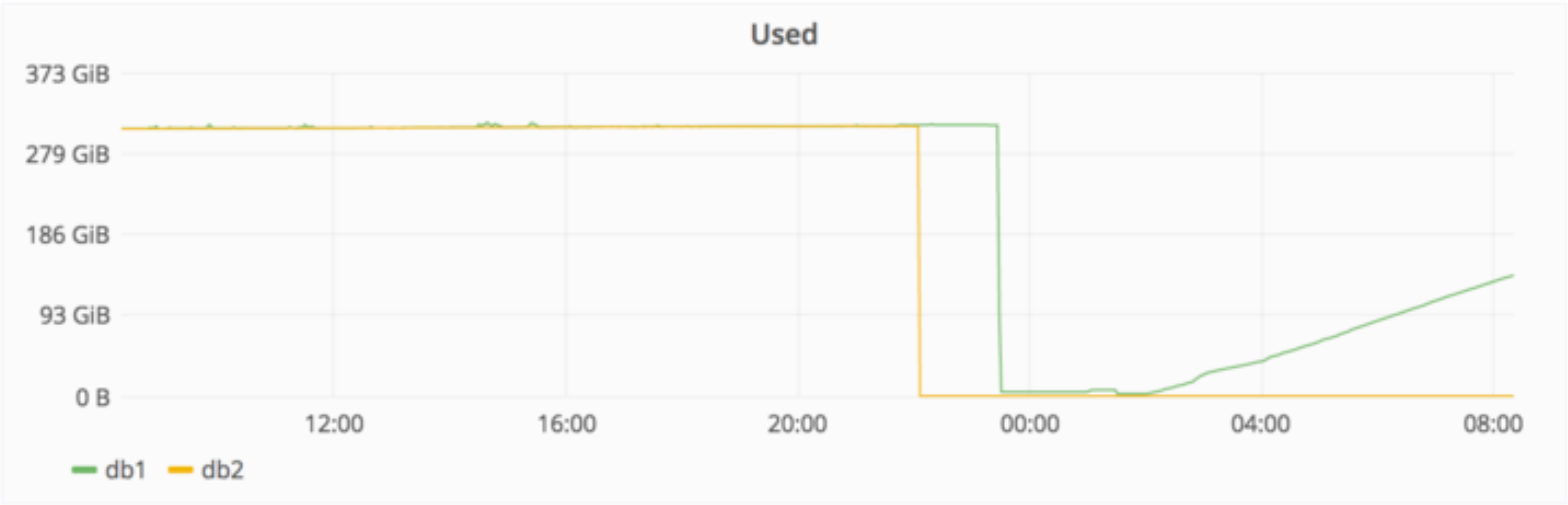
7:44 PM - Jan 31, 2017

♡ 2,689 3,251 people are talking about this

Gitlab.com. Recovery Process

- 2017/02/01 00:36 - Backup db1.staging.gitlab.com data
- 2017/02/01 00:55 - Mount db1.staging.gitlab.com on db1.cluster.gitlab.com
- Copy data from staging **/var/opt/gitlab/postgresql/data/** to production **/var/opt/gitlab/postgresql/data**/
- 2017/02/01 01:05 - nfs-share01 server commandeered as temp storage place in /var/opt/gitlab/db-meltdown

Gitlab.com. Recovery Process



Gitlab.com. Postmortem

- *“Our best estimate is that it affected roughly 5,000 projects, 5,000 comments and 700 new user accounts.”*

  - Feb 1st / 2017

Backups - RMAN

https://docs.oracle.com/cd/E11882_01/backup.112/e10642/rcmintro.htm#BRADV89339

http://www.dba-oracle.com/t_rman_28_noarchivelog_mode.htm

http://www.thegeekstuff.com/2014/11/oracle-rman-restore

http://www.thegeekstuff.com/2013/08/oracle-rman-backup/

https://docs.oracle.com/cd/E11882_01/backup.112/e10642/rcmquick.htm#BRADV89371

http://www.dba-oracle.com/concepts/starting_database.htm

Backup and Recovery - Purpose

- Protect against data loss
- Reconstruct databases
- **Tasks:**
  - Planning and testing
  - Configuring database environment
  - Setting up a backup schedule
  - Monitoring
  - Troubleshooting backup problems
  - Data preservation / Transfer

Backup and Recovery - Protection

- **Data protection:**
- **Physical**
  - Copies of physical files. **Datafiles**
- **Logical**
  - Tables or stored procedures using **Oracle Data pump**

Backup and Recovery - DBA Intervention

1. **Media Failure**
2. **Users Erros**
3. **Application Errors**

Backup and Recovery.
1. Media Failure

1. Physical problem with a disc (Any database can be vulnerable to)
2. Recovery technique depends on files affected
3. Developing a disaster recovery strategy

Backup and Recovery.
2. Users Errors

1. Application logic or manual mistake
2. Greatest single cause of database downtime
3. Can be localized or widespread:
   1. **Localized:** Delete wrong person from table
   2. **Widespread:** Batch job that deletes order for current month

Backup and Recovery.
3. Application Errors

1. Software malfunction can corrupt data blocks
2. Physical corruption
3. Checksum invalid
4. Header and footer of block don’t match

Oracle backup and recovery solutions

1. **Recovery Manager (RMAN)**
   Preferred!
   1. Tool to perform a range of backup and recovery activities (historical data)
2. **User managed backup and recovery**
   1. Mixture commands and SQL *Plus recovery commands

Recovery Manager (RMAN)

1. Incremental backups
2. Block media recovery
3. Binary compression
4. Encrypted backups
5. Automated database duplication
6. Cross-platform data conversion

Let’s practice

https://gist.github.com/amartinezg/7353ffd1439d80cb1e2a598a53f65ba0

Thank you!