CVMFS Distributed Publishing Infrastructure

Radu Popescu, Jakob Blomer
CERN

Publishing to CVMFS repositories

- Centralized release manager machine
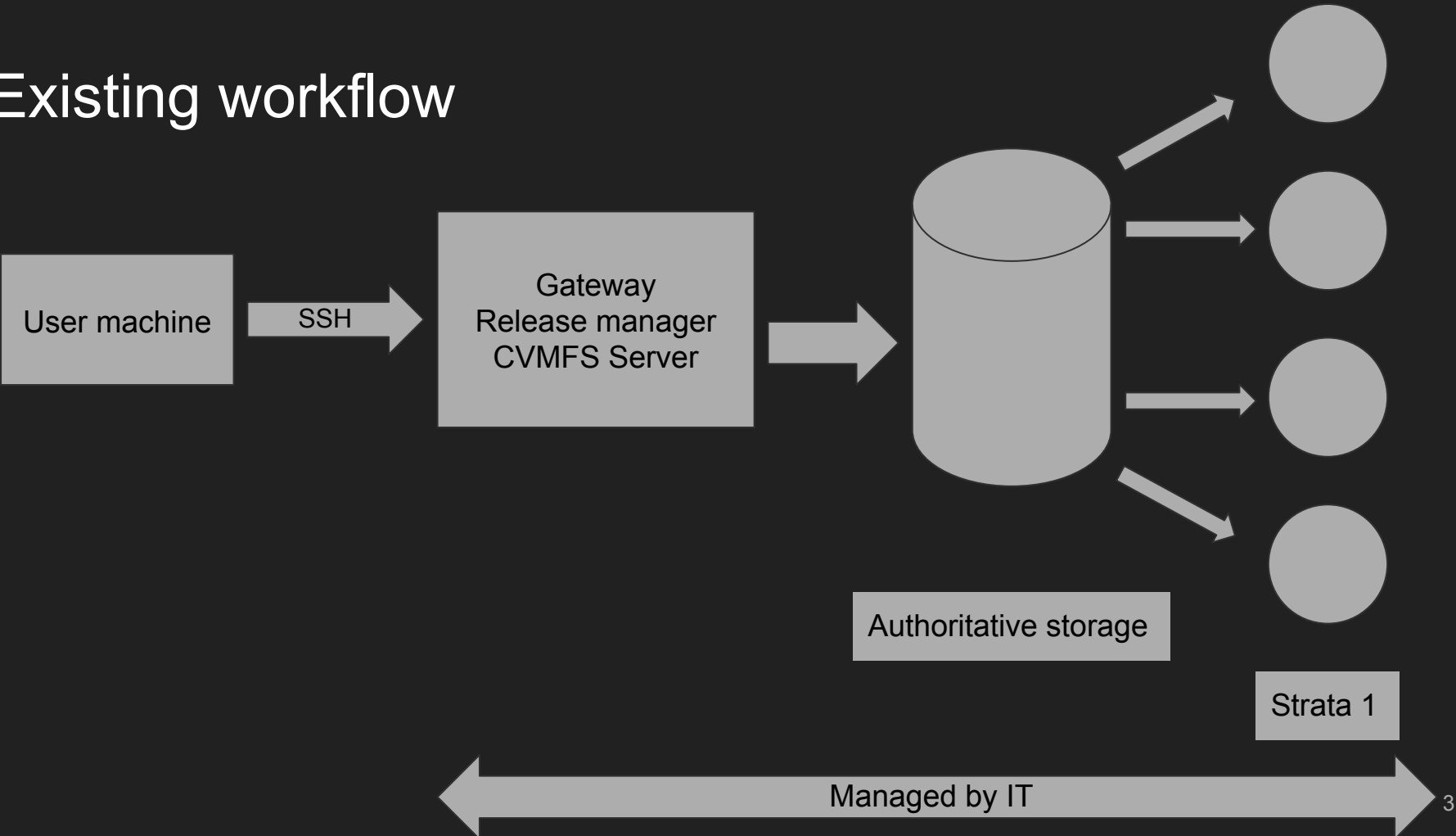- Existing workflow:

```
$ ssh cvmfs-alice.cern.ch

$ cvmfs_server transaction

… (Make changes to files in the R/W mount)

$ cvmfs_server publish
```

Existing workflow

User machine

SSH

Gateway
Release manager
CVMFS Server

Authoritative storage

Strata 1

Managed by IT

Existing workflow

PROS:

- Straightforward to use
- Abstracts somewhat the distributed nature of the system

CONS:

- No support for concurrent writing
  - Performance issues for large changesets
- Can be unsafe (shell access)

Writing is inherently more complex than reading!

Alice conditions data

- Frequent publishing requests
- Potentially large volume of data?
- New files are created on each publish (concurrency!)
- Snapshots

With your help (information) we can design a better system!

Publishing system properties and constraints

1. Consistency is key (should never corrupt the index of a repository)
2. Highest availability possible, given 1.
3. Concurrency can be further exploited due to:
   - immutability of CAS
   - pushing objects is idempotent
   - sub-catalogs
4. The atomic operation (or critical region) is replacing the root catalog

Proposed new workflow
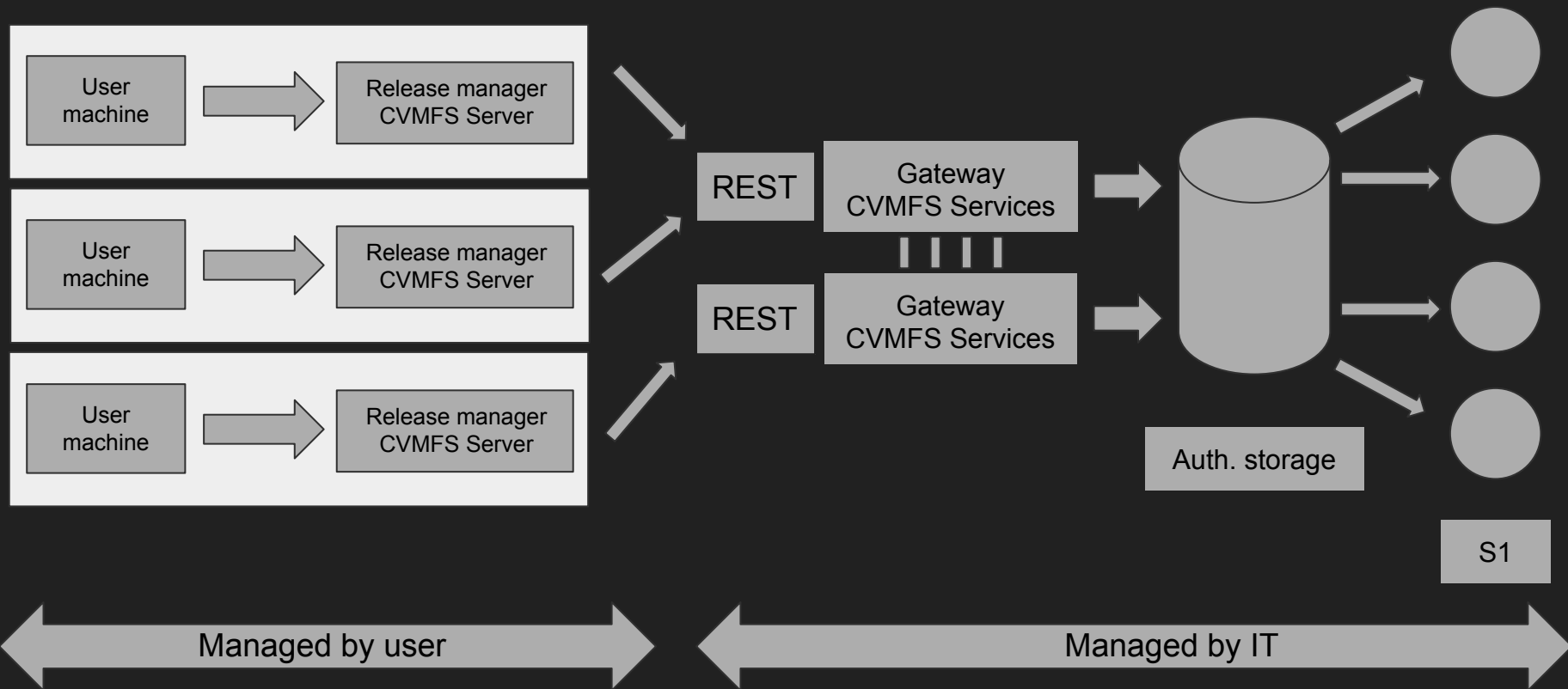
Proposed workflow (contd)

Example:

```
$ cvmfs_server transaction /calibration/2016 # returns an error message if lease cannot be
acquired for given path

…

$ cvmfs_server publish

> Please check status at: http://cvmfs-alice.cern.ch/api/calibration/2016/10/31/p12

$ curl http://cvmfs-alice.cern.ch/api/calibration/2016/10/31/p12

{"revision" : 42, "statum 0" : true, "strata 1" : [ … ], … }
```

Summary

- More concurrent operations will be possible
- Multiple gateway service machines could handle larger (write) traffic
- Feedback (notifications) during publishing and propagation of changes
- More flexible configuration re: number and placement of release managers based on project specifics
- More responsibility for the user
- Timeline:
  - Minimal viable implementation of proposed workflow before end of Q1/17
  - When can it be tested in ALICE?

Thanks!