

# Some news from dCache

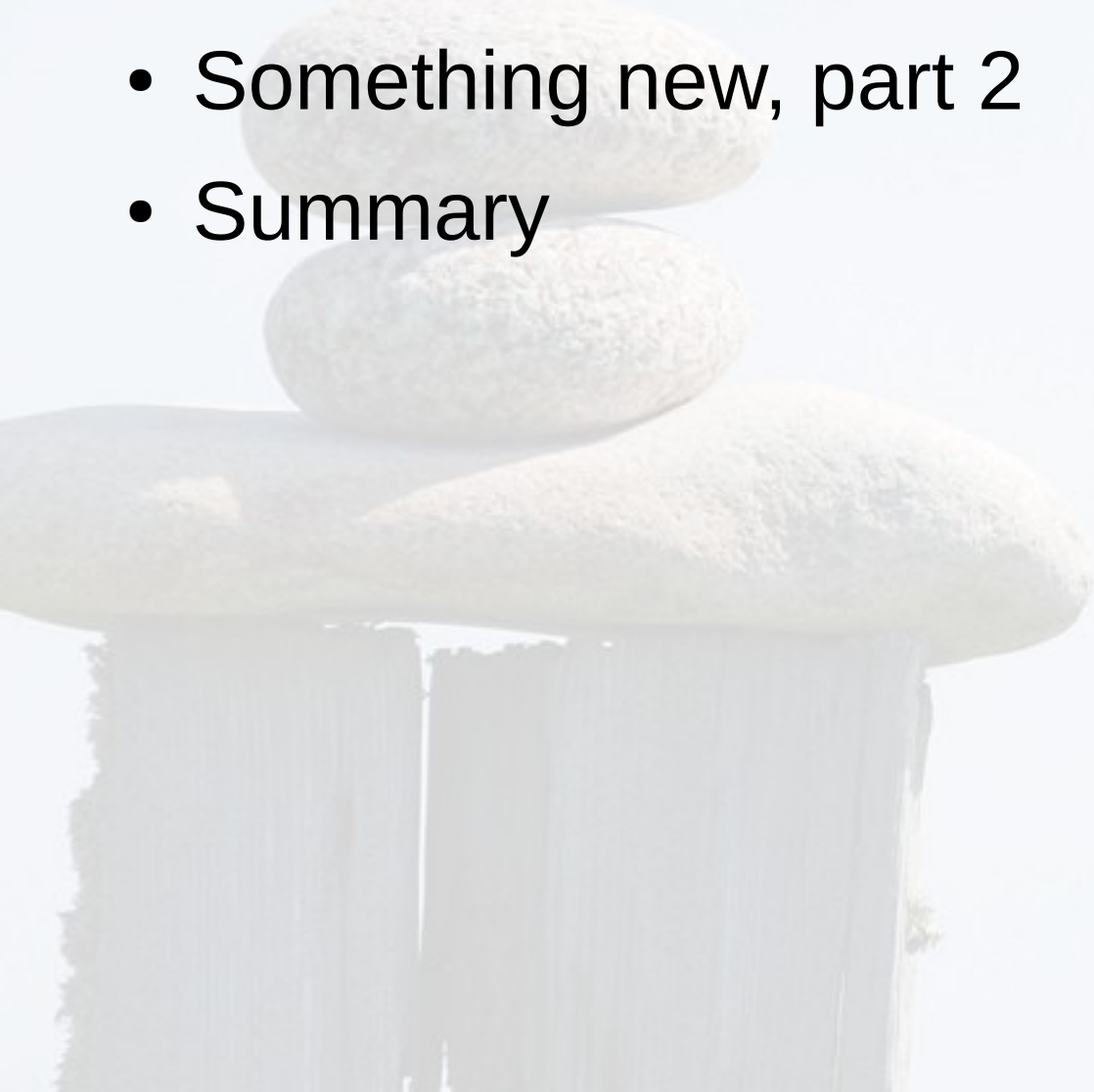


Paul Millar

on behalf of  
the dCache team

# Outline

- Something new, part 1
- Something new, part 2
- Summary



# Something new, part 1

- dCache xrootd support has two parts:
  - **The core** xrootd protocol support (stable)
  - **Adaptation** around xrootd (rapidly changing) e.g.:
    - ALICE provide token information for authentication
    - Federated namespace require VO-specific name-space mappings
- dCache's xrootd support is monolithic:
  - Difficult for people to help or “take joint-ownership” of getting things working.
  - Updates of the “rapidly changing” part linked to dCache release cycles.

# Something new: plugins

- Introducing plugins for dCache's xrootd.
  - Two kinds of plugins:
    - Authentication
    - Authorisation and **name-space mapping**.
- Deployment is simple: drop in a single jar file
- Plugins use normal dCache configuration
- We have moved existing **source code** to public repo,
  - we continue to maintain this code ... but we welcome help
- Anyone can create their own plugins
- What about **testing** all this? ...

# Something new, part 2

- We are currently restructuring dCache source code
  - Moving to maven
  - Splintering dCache into (many) small, reusable modules
  - Some of these will live outside dCache,  
... encouraging reuse.
- Introducing ...

# Something new, part 2

- We are currently restructuring dCache source code
  - Moving to maven
  - Splintering dCache into (many) small, reusable modules
  - Some of these will live outside dCache,  
... encouraging reuse.
- Introducing ...

xrootd4j



# Something new: xrootd4j

- dCache's xrootd support as a library
  - Pure Java implementation of xrootd protocol (now with plugin support)
  - Written using Netty: a high-performance IO framework from JBoss
  - Runs on any platform that supports a Java Virtual Machine (i.e., any platform)
- Located in a public github repository:  
<https://github.com/gbehrmann/xrootd4j>
- Available under an open-source licence (LGPL)
- Comes with example, working stand-alone server



# Xrootd4j: getting started

- Three steps:

- Check out:

```
git clone git://github.com/gbehrmann/xrootd4j.git
```

- Compile:

```
cd xrootd4j; mvn package
```

- Run:

```
java -jar xrootd-standalone/target/xrootd4j-standalone-1.0.SNAPSHOT-jar-with-dependencies.jar
```

- Congratulations: you have a running xrootd server, serving the local filesystem



# What next?

- Releasing **xrootd4j v1.0**
  - Pushing release into public maven repo
  - Projects can use this to pull in xrootd support
    - dCache will be doing this
- Use as a basis for **plugin development**
  - Easy to deploy
  - Easy to test a dCache plugin
- Used in **new xrootd projects**

# What next?

- Releasing **xrootd4j v1.0**
  - Pushing release into public maven repo
  - Projects can use this to pull in xrootd support
    - dCache will be doing this
- Use as a basis for **plugin development**
  - Easy to deploy
  - Easy to test a dCache plugin
- Used in **new xrootd projects**



# Summary

- dCache now supports xrootd **plugins**
- dCache's xrootd protocol support now an external library: **xrootd4j**
- Provide a stand-alone server to allow **easy testing**

Thanks

