



COMDIRAC, the *killer* client module

<https://github.com/DIRACGrid/COMDIRAC/>

Pierre Gay  
4th DIRAC User Workshop – CERN 2014/05/27

# Motivation

- DIRAC command line clients are not user-friendly enough
  - At least for really end users
- Web portal is not sufficient for all uses
- iRODS « iCommands » (<http://wiki.irods.org>) offer a good user experience
  - A kind of extension to Unix file management commands
  - `ls` → `ils`
  - `mv` → `imv`
  - `cp` → `iput`, `iget` or `icp`
  - + `icd`, `ichmod`, `imkdir`, etc.
  - Directly integrated in the shell, with a kind of environment variables (home directory, current working directory, etc.)
- Why DIRAC couldn't have the same feature ?
  - COMDIRAC : « dCommands »

# Session and configurable defaults

- User configurations stored in `~/.dirac/dcommands.conf` (a « .ini » file)
  - Contains profiles with default values (DIRAC group name, file catalog home directory, etc.)
  - Can be easily bootstrapped with default values from the DIRAC CS, issuing the command « `dconfig --guess` »
- Command « `dinit` » creates a session file from a configuration profile
  - Contains session variables (working directory, default destination SE, etc.)
  - Subsequent « `dCommands` » will use and modify these session variables
  - Also creates a DIRAC session proxy

# Data management

- As simple as :

```
$ dinit [<profile name>]
```

```
$ dpwd
```

```
$ dcd [<DFC relative dirname>]
```

```
$ dls [<DFC relative path>]
```

```
$ dput <local file path>... [lfn relative path]
```

```
$ dget <lfn relative path>... [local path]
```

```
$ dmkdir <DFC relative dirname>
```

- And many more with `drm`, `dfind`, `dmeta`, ...
- Can customize default behavior with configuration/session variables
  - Default SE used
  - Basic replication scheme for « `drepl` »

# Job management

- Inspired from classic cluster batch system commands (Torque qsub, qstat)
- A versatile « dsub » command for job submission with payload script/executable or the old good JDL way
  - « dsub » now allows to put JDL directives directly inside the job payload script with « #JDL » directives
- A simple « dstat » command to display jobs
- « dkill » to delete/kill jobs
- « dlogging » to query history of a particular job
- « dinput », « doutput » to download ISBs and OSBs
- dCommands can be used inside jobs
  - provided your community pilots are installed with the « -e COMDIRAC » option
  - use « dinit --fromProxy » for session initialization
  - *not fully tested*

# Possible improvements

- Improve error handling, more complete testing
- « dcp », « dmv » commands
- Refactor metadata management command « dmeta »
- Ease the use of StorageProxy for clients that don't have access to LCG bundle
- Reduce dCommands Script initialization lag ?
- Comments, suggestions and PR welcome

# COMDIRAC and you

- Try it :
  - Easy installation : just add « -e COMDIRAC » to your client installation command
  - Tutorials at :  
<https://github.com/DIRACGrid/COMDIRAC/wiki>
  - Each dCommand has its own « --help » option
- Need a new feature ? Discovered a bug ?
  - File an issue on GitHub
  - Reach me by mail : [pierre.gay@u-bordeaux.fr](mailto:pierre.gay@u-bordeaux.fr)