The 8th DUW

Practical introduction to DIRAC

Federico Stagni

This is a beginners' presentation

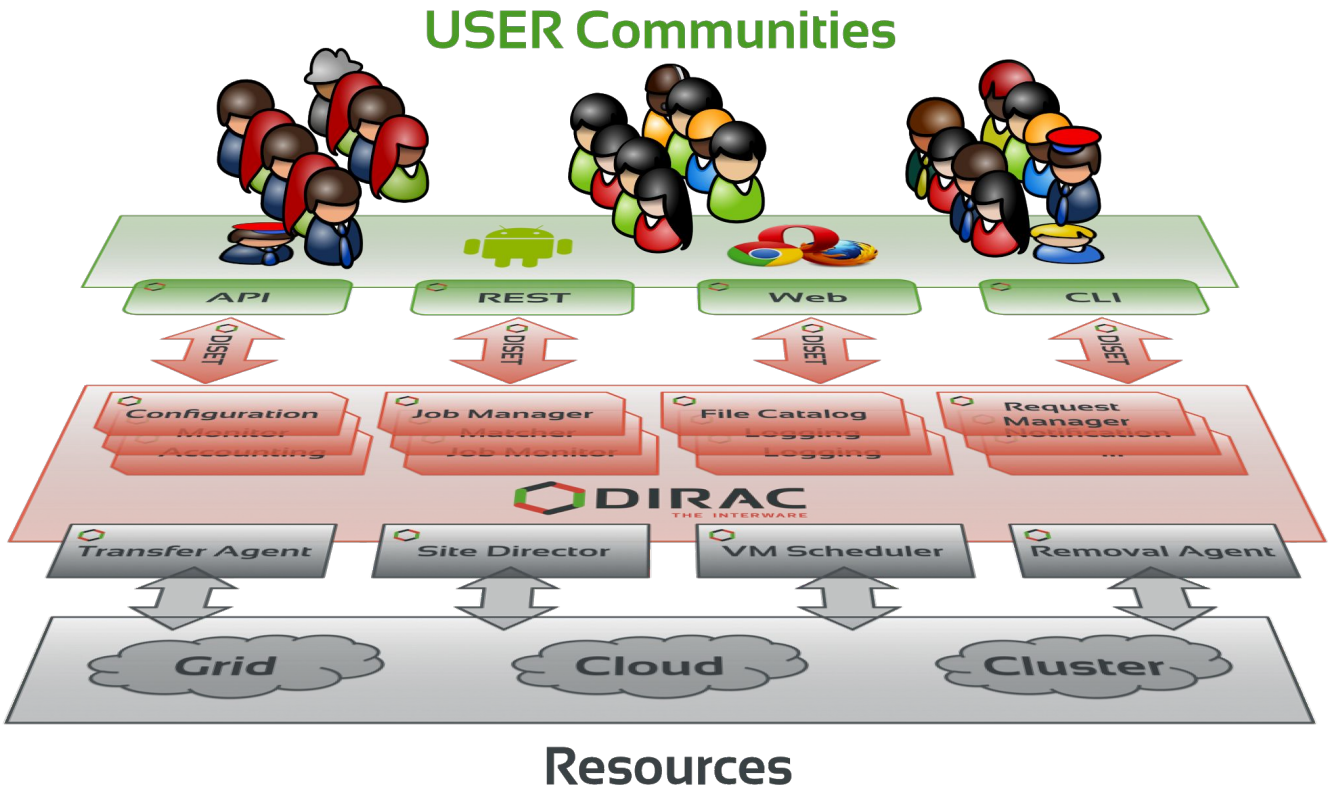
a super-intro

Good part of what's here will probably be repeated at some point of the workshop

(so if you are an expert DIRAC user, you can keep reading your mails)

DIRAC: the interware

- A software framework for distributed computing
- A **complete** solution to one (or more) user community
- Builds a layer between users and resources



… a few examples of what DIRAC can be used for

- sending jobs to “the Grid”
  - the obvious one
- interfacing with different sites
  - with different computing elements
    - and batch systems
  - with different storage elements
- interfacing with different information systems

- managing productions
- managing dataset transfers
  - and removals...
- providing a failover system
  - your jobs won’t fail because a certain SE is down, nor because of central service are down
- transfer data from the experiment to a Grid SE
- … and more

DIRAC
THE INTERWARE

An open source project

- Started as an LHCb project, became experiment-agnostic in 2009
  - First users (after LHCb) end of 2009
- Developed by communities, for communities
  - Open source (GPL3+), GitHub hosted, python 2.7
  - No dedicated funding for the development of the “Vanilla” project
  - Publicly documented, active assistance forum, yearly users workshops, open developers meetings
  - 4 FTE as core developers, a dozen contributing developers
- The DIRAC consortium as representing body

Users/communities/VOs

A *framework* shared by multiple experiments/projects, both inside HEP, astronomy, and life science

<u>Experiment agnostic</u>
<u>Extensible</u>
<u>Flexible</u>

DIRAC
THE INTERWARE

Flexibility: just 1 example

- DIRAC does not impose you any computing model
- Tiers level may not mean anything to certain VOs
- The (fixed) computing model is dead

- Example: full mesh computing model
  - Every job can run everywhere
- A real world computing model is a full mesh with limits
- All configurable in few clicks

Types of installations

1. Client
   a. for users
   b. for pilots
2. Server
   a. single-VO
   b. multi-VO (DIRAC as-a-service)

      with limited functionalities

Supported platforms: **EL6, EL7**

but: not everything “external” to DIRAC is compiled for EL7.

Users' view

DIRAC
THE INTERWARE

Interfaces

- WebAppDIRAC
  - web portal, based on tornado
    - several web apps provided
  - REST interface also provided for few services
- CLI: Scripts (commands)
  - starting with dirac-*
    - COMDIRAC extension provides some simplified UI
- APIs
  - DIRAC.Interfaces.API.

- In the HEP world, Ganga is a common interface to DIRAC

DIRAC
THE INTERWARE

AuthN/AuthZ

X509 certificates and proxies are the only authentication/authorization mean

before anything else: dirac-proxy-init

and put your certificate in the browser

User guides and tutorials:
http://dirac.readthedocs.io/en/latest/UserGuide/index.html

DIRAC releases

DIRAC
THE INTERWARE

Batteries included

Each DIRAC installation comes with software which is not maintained by DIRAC

“DIRAC Externals”

within every release

(stomp, pyparsing, openssl, readline… many more... even mySQL)

“lcg bundle”

if requested

(gfal2, fts, arc, cream, lfc...)

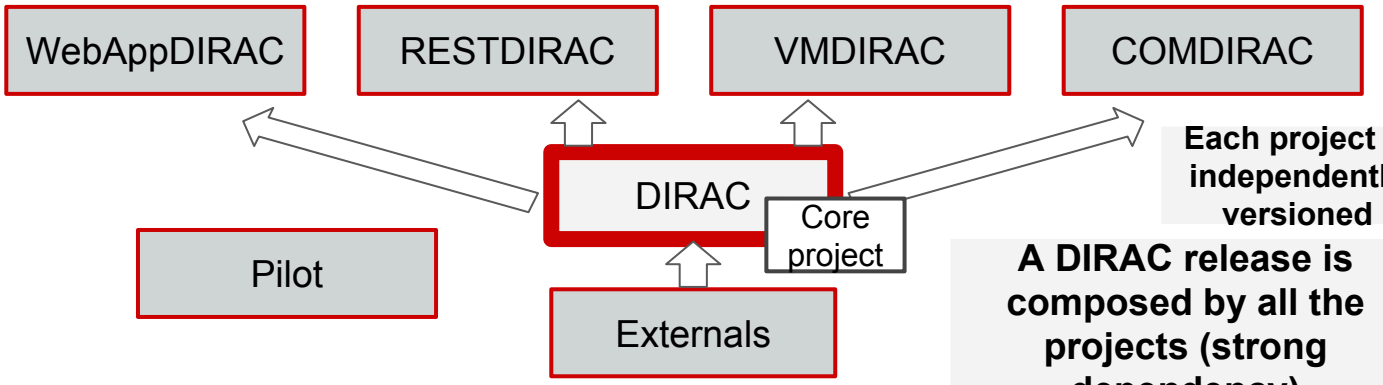
Note: the above will change soon, see pres from Chris tomorrow… but the principle of batteries included won’t change.

Experiment agnostic, and extensibility

"Horizontal" extensibility

-

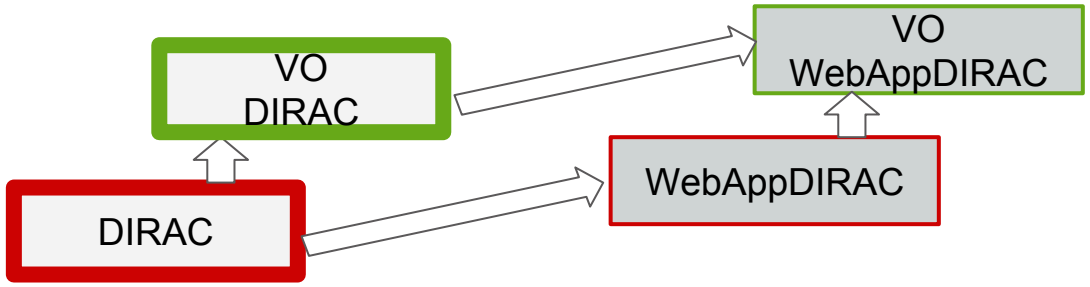
For specific requirements



"Vertical" extensibility

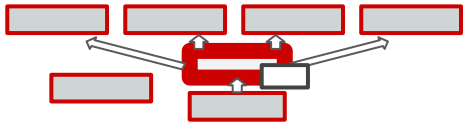
-

Community driven



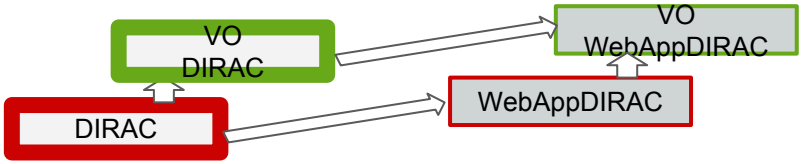
What's a DIRAC release

```
  v6r20-pre23
  {
    Modules = DIRAC, VMDIRAC:v2r2, RESTDIRAC:v0r5, COMDIRAC:v0r17,
WebAppDIRAC:v3r1p6
    Externals = v6r6p8
  }
  v6r19p21
  {
    Modules = DIRAC, VMDIRAC:v2r2, RESTDIRAC:v0r5, COMDIRAC:v0r17, Web:v1r4p3,
WebAppDIRAC:v3r0p1, MPIDIRAC:v0r1, FSDIRAC:v0r3, BoincDIRAC:v0r1
    Externals = v6r6p3
  }
  v6r19p20
  {
    Modules = DIRAC, VMDIRAC:v2r2, RESTDIRAC:v0r5, COMDIRAC:v0r17, Web:v1r4p3,
WebAppDIRAC:v3r0p1, MPIDIRAC:v0r1, FSDIRAC:v0r3, BoincDIRAC:v0r1
    Externals = v6r6p3
  }
```



A (VO)DIRAC release

```
v9r2-pre4
{
  Modules = LHCbDIRAC:v9r2-pre4, LHCbWebDIRAC:v4r4p5
  Depends = DIRAC:v6r20-pre23
  LcgVer = v14r1
}
v9r1p13
{
  Modules = LHCbDIRAC:v9r1p13, LHCbWebDIRAC:v4r4p6
  Depends = DIRAC:v6r19p21
  LcgVer = v13r0
}
```



dirac-install.py

...the script that installs DIRAC.

e.g., for client:

```
wget -np -O dirac-install
https://github.com/DIRACGrid/DIRAC/raw/integration/Core/scripts/dirac-install.py
--no-check-certificate


chmod +x dirac-install

dirac-install -r v6r19p2 -g v13r0
```

Unless extended, or changed, the DIRAC pilot will do the ~same

How/what to install

Today:

- 2 “minor/major” release per year
- Patches: ~every week

➔ every patch is tested with automated tests

➔ every minor/major release is tested with a deep, long, certification process

- ◆ and you’ll probably need to apply some changes “by hand”: https://github.com/DIRACGrid/DIRAC/wiki

DIRAC installations, and administrators' view

Time to install a server

- You need to know what to install for doing a certain activity
  - no one-click install
  - there's no "install me WMS"
    - you need to install the various components of the WMS, if submitting/monitoring jobs is what you want to do
- You can install DIRAC components on as many hosts as you want
- Redundancy:
  - executors can be duplicated
  - services can *normally* be duplicated (but not all!)
    - each one will have one more URL
    - there are few cases of master/slave services
  - agents *may be* duplicates
    - but you need to know what you're doing

Reminders

DIRAC is a complete solution - ~200K lines of code

Maybe stating the obvious, but:

- Administration is a daily work
  - There's quite some documentation, but many things are learned simply using it
  - Looking at the logs, or in the DB, at some point will become necessary
  - Services and agents DO get stuck and needs to be debugged
- At some point, you'll probably need to code something for your extension

Single-VO vs multi-VO

A DIRAC installation can be used for one VO only, or for more than one VO (multi-VO)

As of today, not all the functionalities are multi-VO aware:

- Transformations (productions) management
- Distributed data management
- Resources (status) management

A multi-VO installation is, most and foremost, for user jobs ("small" VOs)

Communication

DIRAC
THE INTERWARE

Mostly for Ops

Google forum

and, it’s enough

DIRAC
THE INTERWARE

For Devs

BILD:

"BiWeekly 'Loyal' DIRAC Developers meetings"

(almost) every 2nd week, Thursday at 10:00 AM CET

LHCb hosted

ILC, Belle2, CTA, BES3, GridPP represented

Where releases and issues are discussed!

you are welcome to participate actively

Supporting DIRAC

DIRAC
THE INTERWARE

By writing Code

...just like any other project hosted on GitHub!

Don’t know how? Come at the Dev tutorial On Thursday

By writing DOC, or answering questions

- this: http://dirac.readthedocs.io/en/integration/index.html
  click on low, right side, search for “edit”, click, write...
- or this: https://github.com/DIRACGrid/DIRAC/wiki
- or on the forum
- ...or by contributing to this same workshop

The elephant in the room

https://github.com/DIRACGrid/DIRAC/graphs/contributors

LHCb is, still, the driving force here.

contributions from CLIC and GridPP very appreciated!!!

We are happy to spread our knowledge, but:

- we could put a lot of our code in LHCbDIRAC only
  - but we don't, and *you shouldn't*
- ultimately, LHCb effort in DIRAC is with LHCb in mind
  - LHCb doesn't have any specific interest in developing with multi-VO in mind
  - ...nor integrating software that don't use

DIRAC
THE INTERWARE

Questions/comments

?