

# iLCDirac: DIRAC for the linear collider

André Sailer

CERN-EP-LCD

DIRAC User Workshop  
Montpellier  
May 23, 2016

# Contents



- 1 iLCDirac Use Case
- 2 Current Status
- 3 Testing and Documentation for iLCDirac
- 4 Developments
- 5 Room for Improvements
- 6 Conclusions

- ILC VO: virtual organisation for linear colliders (ILC and CLIC)
- ILCDirac is an extension of the DIRAC system for the ILC VO
  - ▶ Workflow Modules for LC Software, Overlay System
  - ▶ J. Phys.: Conf. Ser. ILCDirac, a DIRAC extension for the Linear Collider community. Proceedings of CHEP2013. 513 [CLICdp-Conf-2013-003](#)
- Centralized MC Production (Event Generation, Geant4 Simulation, Reconstruction)
- User jobs (Generation, Simulation, Reconstruction, Analyses)

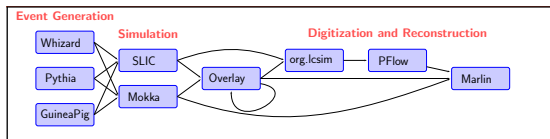
## Capacity:

- Using WLCG and OSG resources
  - ▶ Mostly opportunistic, some dedicated
  - ▶ Around 15k to 20k job slots available at best of times

Code: <https://gitlab.cern.ch/CLICdp/ILCDIRAC>

- Define application payload via interfaces
- Chain applications (append one after the other)

```
from DIRAC.Core.Base import Script
Script.parseCommandLine()
import UserJob
import Marlin
import DiracILC
d = DiracILC()
j = UserJob()
j.setOutputSandbox("recEvents.slcio")
m = Marlin()
m.setVersion("0116")
m.setSteeringFile("Steering.xml")
m.setInputFile("SimEvents.slcio")
j.append(m)
j.submit(d)
```



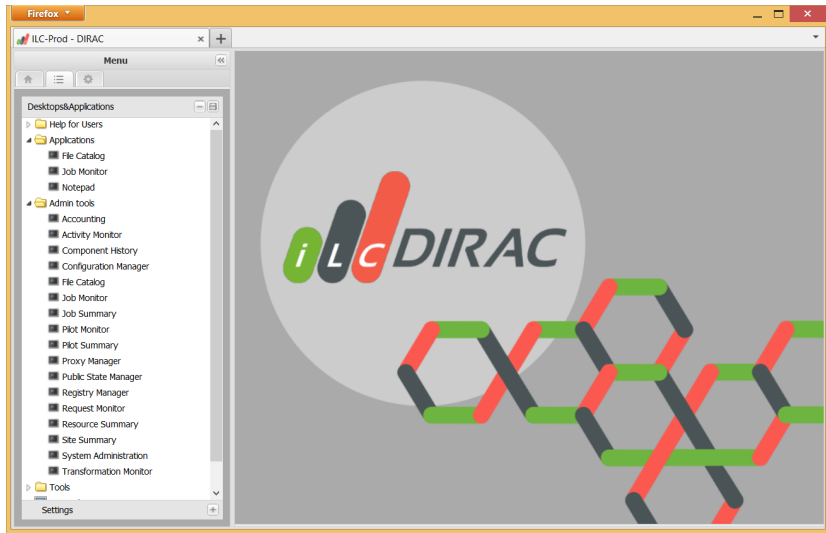
- DIRAC Version: v6r14p18 (servers), v6r14p26 (pilots)
- Took a long time to update from v6r12 to v6r14 to get all the things no longer working back in shape
  - ▶ XROOTStorage
  - ▶ InputDataResolution
  - ▶ ...

Added a set of redundant servers for backup, in case primary servers go down  
Total of 100 Cores and 200 GB of Ram, SLC6 Virtual Machines

- 2× 3 Servers running Agents and Services: 8 Cores, 16 GB RAM; Split by DIRAC-System
  - 1 Framework, Transformation, DataManagement, Configuration
  - 2 StorageManagement, WorkloadManagement
  - 3 RequestManagement, Accounting, ResourceStatus
- 3 DIRAC DIP-Storage SEs: 4 Cores, 8 GB Ram, 1 TB Volume
  - ▶ DIP-SE, Log-SE, SB-SE
- Web Server 4 Cores, 8 GB RAM
- DBs hosted on CERN DB on Demand (iLCDirac, ilcacdb (accounting DB), ilcdtest)
- Development, Testing, continuous integration (8 × 1 core), and spare VMs

# WebApp

Nicely Customised by Marko



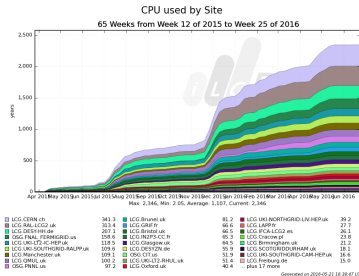
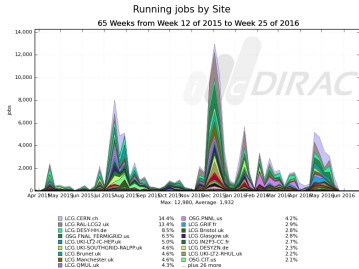
Getting used to it, some issues reported to have same functionality as old

webportal

# CPU Usage



- Activity in bursts
- Maximum > 12k jobs
- Integrated all OSG resources allowing ILC-VO
  - ▶ HTCondor-CE and Globus Computing Elements
  - ▶ No SiteDirectors at Sites

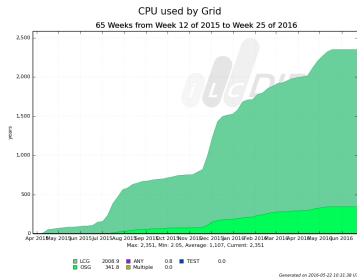
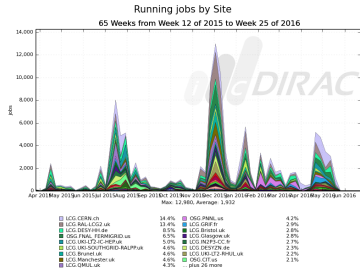




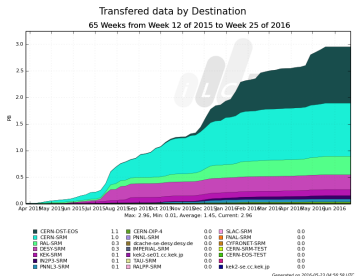
# CPU Usage



- Activity in bursts
- Maximum > 12k jobs
- Integrated all OSG resources allowing ILC-VO
  - ▶ HTCondor-CE and Globus Computing Elements
  - ▶ No SiteDirectors at Sites



- Using the DiracFileCatalog
- 17 Million files (20 million replicas), 3.8 PB (4.1 PB total), 6 Million files, 1.8 PB since the last workshop (if the reporting from DFC can be trusted)
- Metadata used to define input files for transformations



- Set up Continuous Integration for iLCDirac via GitlabCI
- Unit and integration tests (running jobs, copying files) for iLCDirac
- Installation of iLCDirac on SL5/SL6/CC7
- Using the HEAD of DIRAC release branch (rel-v6r14)
- Aiming for as complete coverage as possible in iLCDirac
  - ▶ Catch bugs in our code
  - ▶ Catch interface changes in DIRAC

- Moved to sphinx based documentation
- Linked with DIRAC code documentation for baseclasses, functions, . . .
- Documenting the iLCDirac API for application configuration
- <http://lcd-data.web.cern.ch/lcd-data/doc/ilcdiracdoc/>



The screenshot shows a web browser window with the title "iLCDIRAC v25r0p7 documentation". The browser's address bar shows "next | modules | index". The page content is as follows:

- Table Of Contents**
  - iLCDIRAC Documentation
    - Interfaces for User Jobs
      - Scripts
    - Production Scripts
    - Contents
    - Release Notes
    - Indices and tables
- Next topic**
  - Applications
- This Page**
  - Show Source
- Quick search**
  - Enter search terms or a module, class or function name.

The main content area has a blue header "iLCDIRAC Documentation" and a sub-header "Interfaces for User Jobs". The text reads: "Welcome to the iLCDIRAC Documentation." and "If you are looking for how to submit jobs for Linear Collider Software please look at the `UserJob` class and the `Applications` modules and finally at the `DiracILC` class".

- Applications
- UserJob
- DiracILC

Below this is a section titled "Scripts" with the text: "Scripts of interest to the casual user are part of the `Interfaces` module".

- Interfaces Scripts
  - `dirac-lic-find-in-FC`
  - `dirac-lic-show-software`
  - `dirac-lic-show-software`

## ■ Some developments for iLCDirac

- ▶ Workflow and interfaces for new applications
- ▶ Re-implemented our DataRecoveryAgent for better consistency of productions
  - ★ Check each job if outputfile exists, inputfile still exists, what the status(es) are, if other jobs treated the inputfile, and act accordingly
- ▶ Executor to prevent jobs going to sites where requested software is not available

## ■ Some contributions to DIRAC

- ▶ HTCondor-CE
- ▶ XROOTStorage fixes
- ▶ Script to sync folders between local/SE
- ▶ TravisCI
- ▶ Documentation (Agent Parameters, Code Doc)
- ▶ Bugfixes

# Future Plans for Developments



- Support for additional linear collider specific applications
- RequestTransformation with more than 1 one operation, running something on our development instance with exactly 2 operations to move files
- Larger test coverage
- Maintenance

- Issues on github sometimes feel as if they are ignored
  - ▶ If things will not be changed, just say that;
  - ▶ Should we just “bump” issues every couple days? weeks?

## New features/re-writes:

- Where are they discussed?
- Where are they documented?

## Release Notes, Change Notes

- What, how, and *why* things changed
- Document additional or changed configuration parameters
  - ▶ Much easier now that documentation is in the DIRAC repository
  - ▶ Announce when defaults changed, and why, because one might need to adapt the existing configuration
- Changes in DB



Documentation is great, there should be more of it

- Operation parameters
- Agent and service parameters

# Tests



Testing is great, there should be more of it

# Conclusions



- We are making good use of our resources via DIRAC
- Our users are generally happy, and the system is easy to use for them
- Life for us admins and developers could be a bit easier