GridPP
UK Computing for Particle Physics

Developing LHCb Grid Software: Experiences and Advances

Ian Stokes-Rees
University of Oxford
Department of Particle Physics
2 September 2004

Oxford Physics

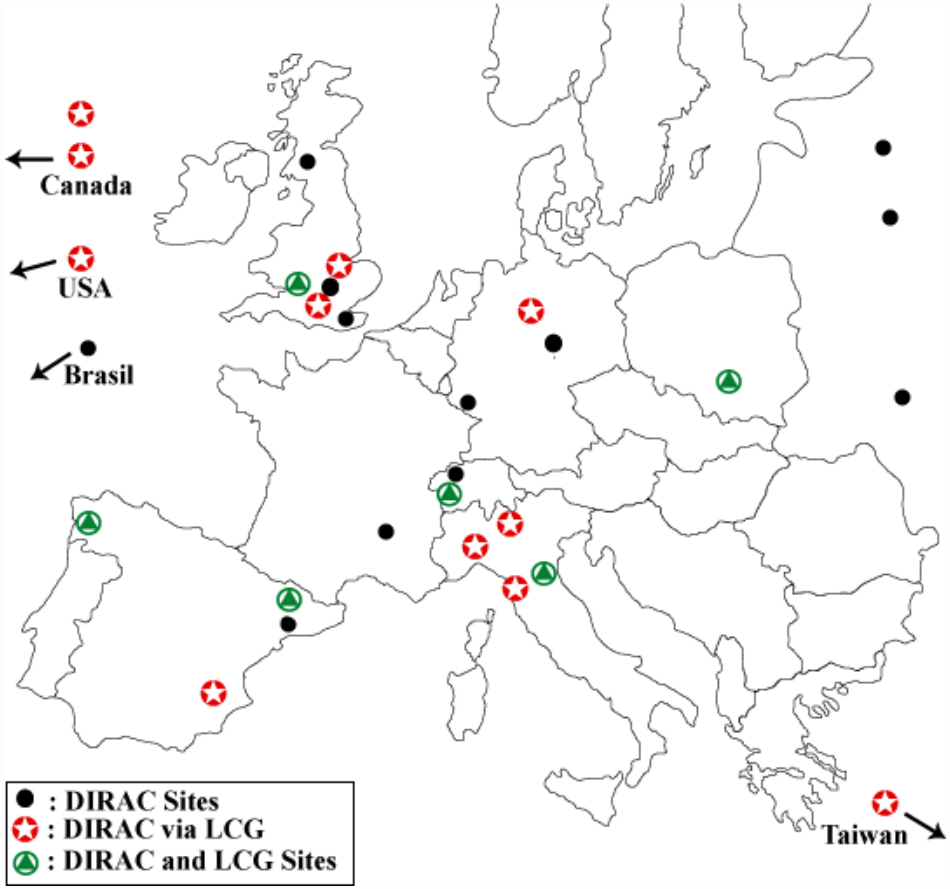
60 second version

- LHCb Particle Physics Experiment developed a computational grid infrastructure, starting in 2002
- Deployed on 20 "normal", and 30 "LCG" sites
- Effectively saturated LCG and all available computing resources during 2004 Data Challenge
- Supported 3500 simultaneous jobs across 50 sites
- Produced, transferred, and replicated 58 TB of data, plus meta-data
- Consumed over 400 CPU years during last 3 months
- Achieved by
  - lightweight Services and Agents
  - developed in Python
  - with XML-RPC interfaces
  - and, of course, a lot of blood, sweat, and tears

GridPP
UK Computing for Particle Physics

Overview

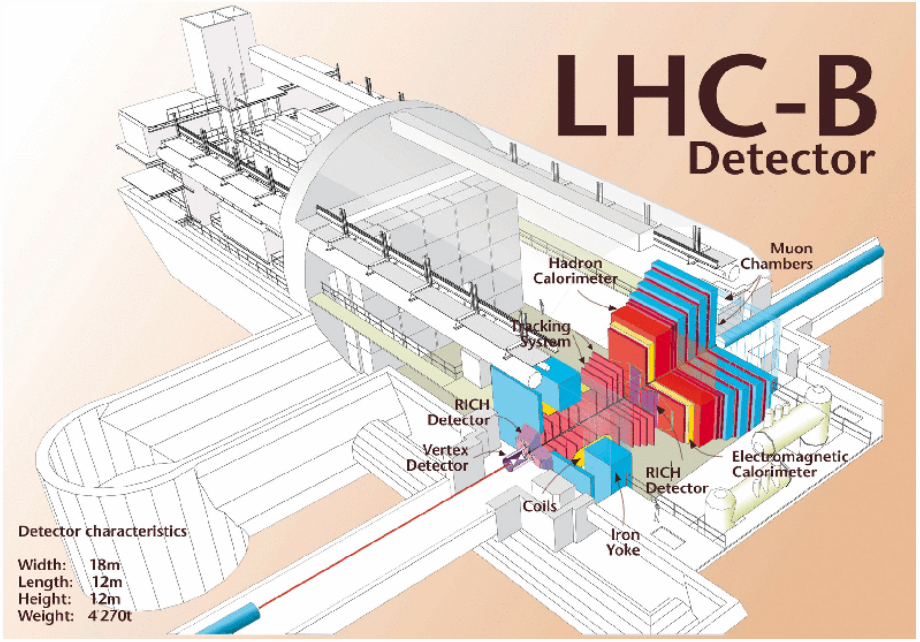
- Requirements
- Architecture
- Integration with LCG
- Project Management
- OGSI/GT3 Flop
- Instant Messaging
- Future

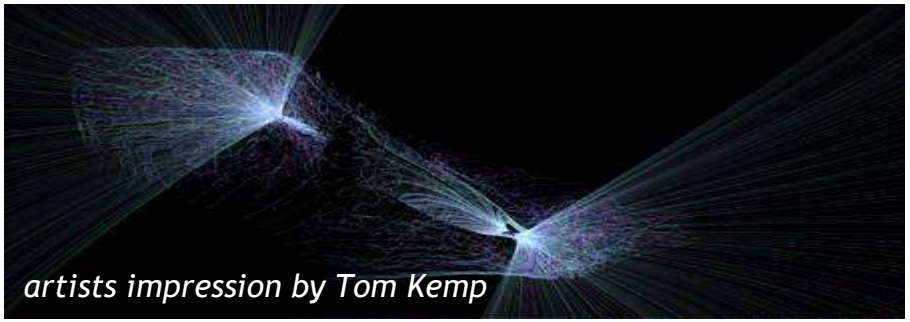


DIRAC Agent Network, July 2004

Background

- LHCb experiment
  - particle physics detector at CERN
  - will generate data at 40 MB/s from 2007
    - that's 3.4 TB/day
  - 500 physicists
  - 100 institutes/sites
  - simulations already running
  - software development and testing underway





Requirements

- Required simple integration with existing computing centres
  - support different batch systems
  - easy for LHCb site representatives to install, configure, and run "Grid Software"
  - little or no intervention while running
- Needed to support LHCb computing
  - Existing data management and simulation software and services
  - Regular software updates
  - Large data files with associated meta-data and replication

Requirement Metrics

100,000 queued jobs

10,000 running jobs

100 sites

We think this is what computational grids look like

CERN LHCb

LCG?

- But why not just use EDG or LCG?
  - In 2002, EDG was not ready for serious use
  - Lots of existing computing resources still not (yet) tied in to LCG
  - LHCb sought to develop a stepping stone to LCG computing
- … and LHCb had some ideas on how to do "grid computing" a bit differently …

Architecture

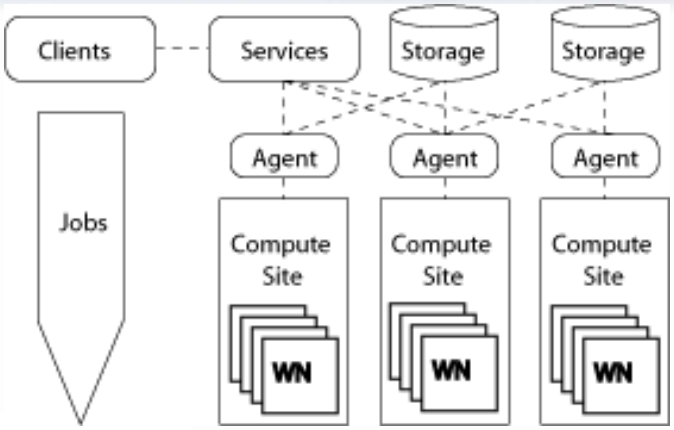
DIRAC: Distributed Infrastructure with Remote Agent Control

- Service Oriented Architecture
  - Services exposed via simple XML-RPC interface
  - accessible over HTTP
- 99% Python 

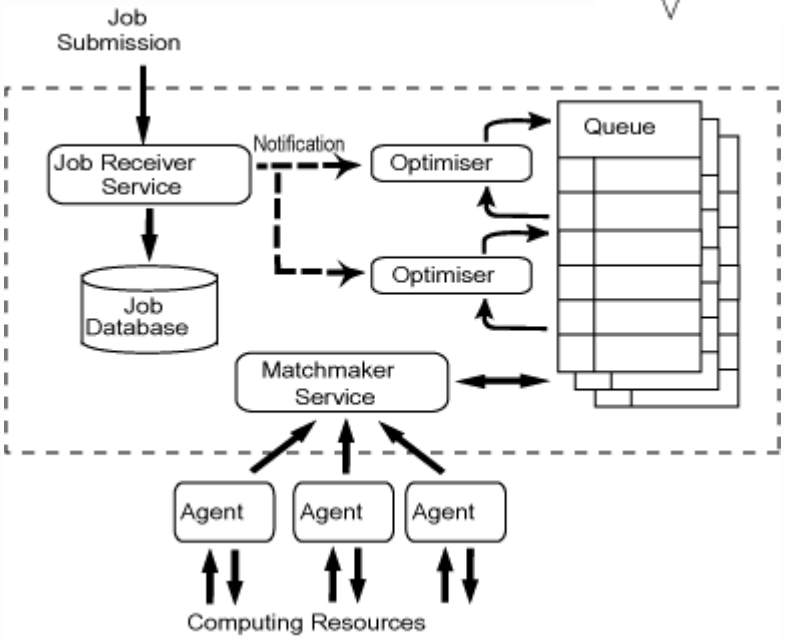
- DIRAC Agents deployed at computing centres
- Job Pull Paradigm, similar to Condor
  - in fact, using Condor ClassAds, and Condor Matchmaker

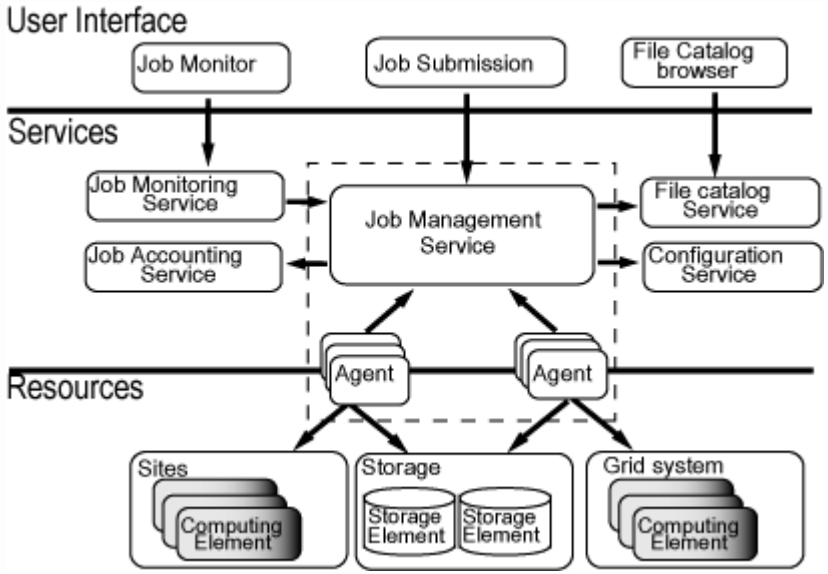
Architecture

Services
Agents
Clients



Users
Jobs
Data





Sound Familiar?

- DIRAC architecture followed:
  - OGSA/OGSI direction towards "grid services"
  - Direction of ARDA proposal to EGEE
    - Now implemented as gLite
- DIRAC was meant to fit into this brave new world of Grid Services
  - ... and we tried (GT3, OGSI, pyGridWare, Clarens)

Aside: ARDA

- Dream was that ARDA, possible successor to EDG architecture, would propose a service decomposition and simple, clear, interfaces
  - Allow alternative/pluggable/replaceable service implementations
    - For competition
    - For bug fixing
    - For different feature/performance emphasis
  - Allow extension of "Grid Functionality" through new services
  - Allow rapid development of services

Architectural Aspects of DIRAC

Python

- LHCb Experiment Standardized on Python wherever possible
- I had serious doubts about the performance of an interpreted language for a production grid system
  - Proved wrong! Python worked just fine.
- Facilitated rapid development and bug fixing
- Good object oriented construction
- "Dynamic Typing" (aka not type safe) is a challenge and requires careful coding
- "Batteries Included" meant that DIRAC Agents and Clients were super lightweight and only required:
  - 1.2 meg tarball (Python code and associated libraries)
  - Python 2.2 interpreter installed
  - Outbound internet connection

Service Oriented Architecture

- Allowed reconfiguration of overall system
- Encouraged rapid development
- Automatic paralellism
- Easy deployment and maintenance
- Forced separation of functionality
- Scaled well
- Significant complexity of co-ordinating configuration and location of services

Multi-threaded XML-RPC

- Fast
  - 40 queries per second
- Easy
  - 3 lines of Python

```
server = ThreadingXMLRPCServer(...)
server.register_instance(service)
server.serve_forever()
```

- Didn't need anything more complicated
  - SOAP, WSDL, etc.

XML-RPC

```
POST /RPC2 HTTP/1.0
User-Agent: Frontier/5.1.2 (WinNT)
Host: betty.userland.com
Content-Type: text/xml
Content-length: 181

<methodCall>
   <methodName>examples.getStateName</methodName>
   <params>
      <param>
         <value><i4>41</i4></value>
      </param>
   </params>
</methodCall>
```

SOAP

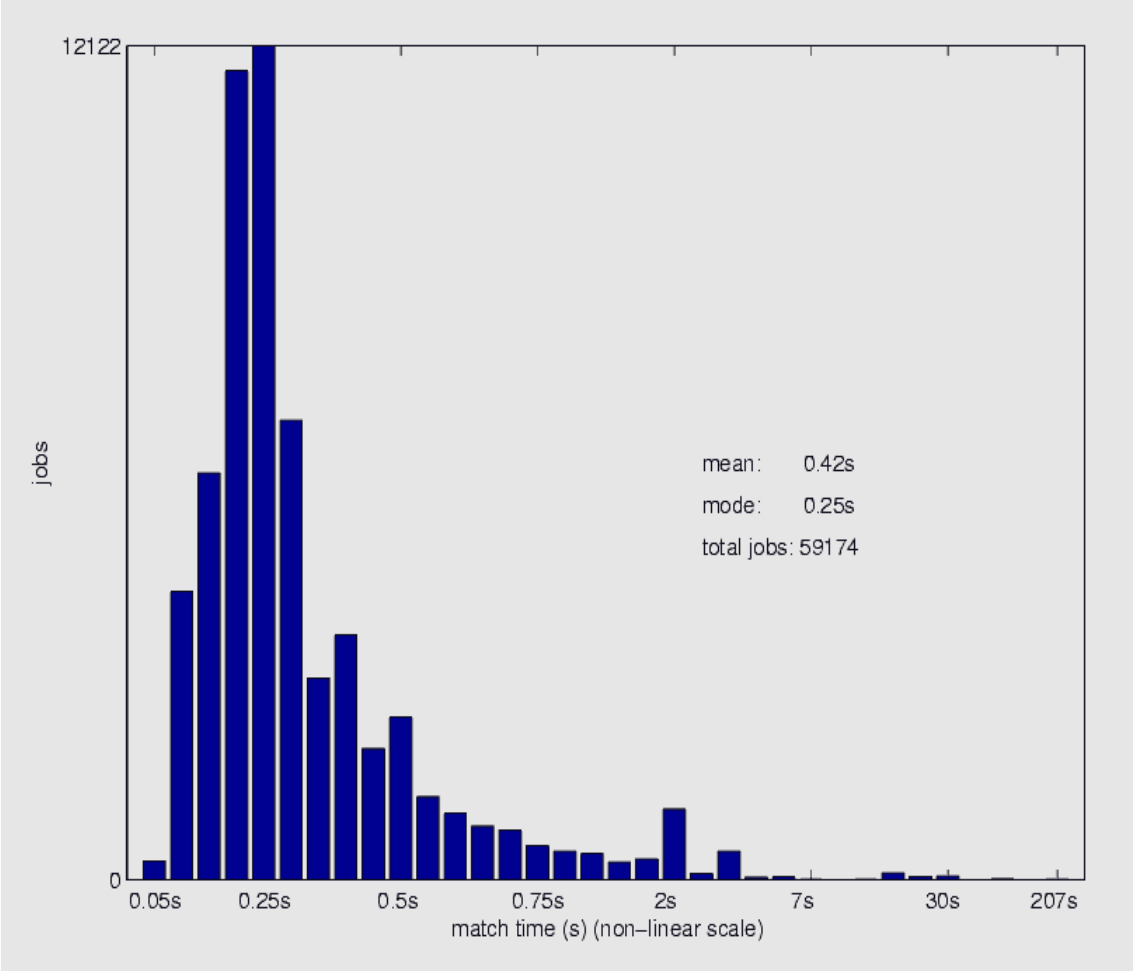
```
POST /InStock HTTP/1.1
Host: betty.userland.com
Content-Type: application/soap+xml; charset=utf-8
Content-Length: nnn

<soap:Envelope
 xmlns:soap="http://www.w3.org/2001/12/soap-envelope"
 soap:encodingStyle="http://www.w3.org/2001/12/soap-encoding">
  <soap:Body xmlns:m="http://userland.com/examples">
    <m:GetStateName>
      <m:Index>41</m:Index>
    </m:GetStateName>
  </soap:Body>
</soap:Envelope>
```

Pull Scheduling

- Unreasonable to ever expect a single machine to schedule all jobs in the grid, or even all jobs for a VO
  - Assumes complete view of system
  - Assumes up to date information
- Push scheduling introduces single point of failure and overloading in presence of 1000s of jobs (NP hard)
- Pull scheduling is **Obviously Better**
  - Resources ask for jobs when they are ready
  - Job serves "next best job" for that resource

Match Time

- Averaged 420ms match time over 60,000 jobs
  - Using Condor ClassAds and Matchmaker
- Queued jobs grouped by categories
- Matches performed by category
- Typically 1,000 to 20,000 jobs queued
- We still suffered from single point of failure



GridPP
UK Computing for Particle Physics

Instant Messaging on the Grid

- Lots of Agents, Clients and Services
- Changing location
- Restricted network access
- Need for reliable two-way communication

- Idea: Use asynchronous, buffered, reliable messaging framework – Jabber/XMPP IM

CERN
LHCb

Jabber

- "Chat Rooms" provide *ad hoc* broadcast messaging hubs, and dynamic list of "active" jobs, services, agents, clients.
- Information/Query mechanism can be used to expose RPC API
- Presence can be used for component status
- Connection based:
  - "tunnel" back to component, even if on NAT and/or behind firewall
  - Authenticate once
- Humans can interact with components using standard IM client – just open a chat session!

Experiences

OGSI and GT3

- Initial plan for DIRAC v2 was to implement all services with OGSI
  - ideally pyOGSI or pyGridWare (stay 100% Python)
  - ... but GT3 and maybe Jython would do in a pinch
- Conceptually, OGSI was excellent
- In practice, it was too complicated
- And GT3 was impossible to work with
  - Insufficient documentation
  - Buggy implementation
  - Performance was terrible
  - Development was arduous

Grid Library Shopping List

1. Robust libraries
2. Good documentation
   - tutorials, APIs, installation, developers guide, FAQ
3. Conceptually simple
4. Ease of installation
5. Ease of development
6. Smooth integration with existing tools
   - Tomcat, Axis, Globus
7. Performance
8. Scalability
9. Portability
10. Lightweight clients
11. Operation in unprivileged user space

Of course we expect it to work with:

- expert administrator, "root" access
- 2 gigs free hard drive space
- 512 megs of RAM
- 100% "default" install
- 10-100 services deployed
- No firewall
- Access only from other systems and users who are similarly equiped

But will it work with:

- 1000 services on one machine
- 5000 connections to one service
- 10,000 grid jobs running at once
- jobs interacting with 100 services
- services distributed across 20 machines at 5 sites
- 20,000 users, many novice

GridPP
UK Computing for Particle Physics

Integration with LCG

- Transition from "classic" computing centres to "grid" computing was achieved
  - Started 100% classic, 0% LCG (600-1200 jobs)
  - Soon moved to 80% classic, 20% LCG (1000-1500 jobs)
  - Finished at 20% classic, 80% LCG (2500-3500 jobs)
- Initial efforts to utilise LCG were plagued by endless problems:
  - Jobs aborted mysteriously
  - Jobs disappearing
  - Queue times not as reported
  - Difficult to submit large numbers of jobs

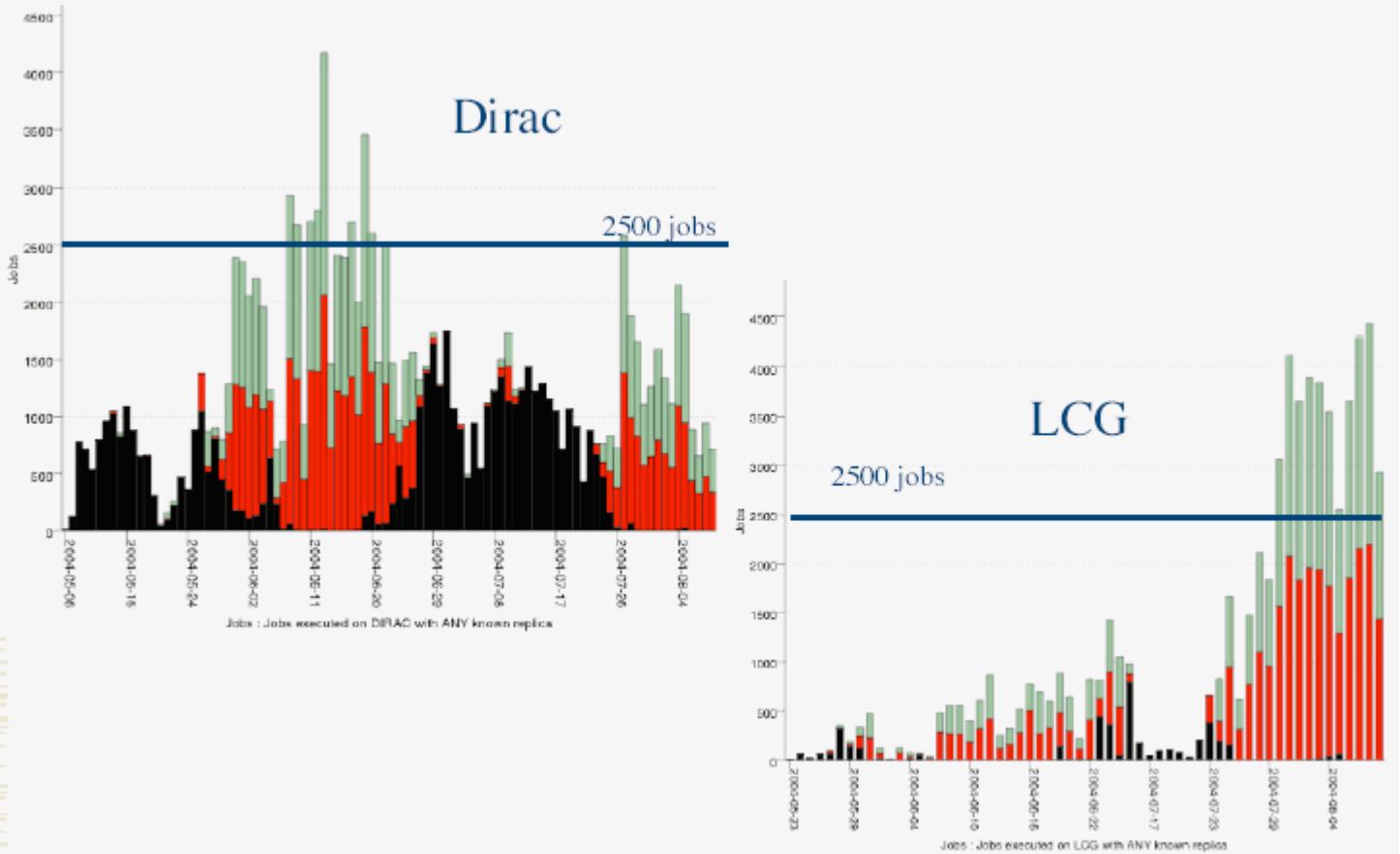
CERN

LHCb

Integration with LCG

- LCG were very supportive
  - Assigned two support contacts
  - Provided 3 dedicated LHCb Resource Brokers
  - Arranged weekly phone conferences
- But it was still difficult to run 3000+ jobs a day on LCG
  - Resource Broker couldn't cope
  - Commands not designed for large numbers of jobs
  - Difficult to diagnose problems
- Heroic efforts by Ricardo Graciani (LHCb member from Barcelona) and collaboration with LCG team got new RB in place and running 4000+ jobs at >95% success.

Transition to LCG



GridPP
UK Computing for Particle Physics

Specific Issues with LCG

- Queue time normalisation
  - Hyper Threading
  - Overloading
- Job scratch space
  - Not enough
- Output files erased
  - Made debugging impossible
- Security certificates
  - RB used wrong ones
- Working with large numbers of jobs
  - Almost impossible
- Major problems with RB
  - Largely resolved now
- Lack of API documentation

CERN

LHCb

Expect The Worst

- On the grid, if something can go wrong, it will:
  - Network failures
  - Drive failures
  - Systems hacked
  - Power outage
  - Bugs in code
  - Flaky memory (parity errors)
  - Time outs
  - Overloaded machine/service
  - Simultaneous operations (mutex, thread safety)

Fault Tolerance

- Everything must be fault tolerant, because faults are guaranteed to happen
  - Retries
  - Duplication
  - Fail-over
  - Caching
  - Watchdogs
- `runit` package was incredible
  - Watchdog
  - Auto-restart
  - Daemons
  - Auto-logging with timestamps
  - Setuid
  - Log rotation
  - Dependency mgmt
  - Sending signals

GridPP
UK Computing for Particle Physics

Human Factors for a Successful Grid Project

CERN

LHCb

DIRAC and DC04 Project Management

- Project management was key to success of DIRAC development and DC04 grid computing
- Three interest groups
  - Core DIRAC developers
  - Physicists and managers for simulation
  - Computing site representatives

Project Management

- Weekly phone meetings
  - Between developers and simulation managers
  - Between site reps and simulation managers
- Two mailing lists
  - One for developers and planning
  - One for discussing ongoing simulations
- Quarterly "LHCb Software Weeks" at CERN

Web Tools

- Use of CVS from outset (and WebCVS)
- Tied in to CERN Savannah System
  - Bug Tracking
  - Task Tracking
  - Support Requests
  - Excellent Software Project Mgmt Tool!
- Project Wiki for workbook and notes
  - Now using GridSite

DIRAC-2 - Summary

LHCb Projects

Agent Download

The following commands will download the latest CVS version of the dirac agent script. This script has built in options for installing or updating the agent software. Execute "./dirac_agent -h" or "-d" for documentation.

```
wget -O dirac-install http://tinyurl.com/yupa9
chmod a+x dirac-install
```

Agent Install

The following commands outline a basic installation, which will enable an InProcess agent. Editing of the `Agent.ini` file is necessary to customise the DIRAC Agent for your local sites batch system, queues, and storage configuration.

```
./dirac-install
```

Membership Info

Project Admins:
- Ian Stokes-Rees
- vincent garonne
- Andrei Tsaregorodtsev
- Manuel Sanchez Garcia

Members: **6**
[View Members]

Group Identification

- #192
- dirac2

GridPP
UK Computing for Particle Physics

Savannah Bug Tracker

Item ID	Summary	Submitted On	Assigned To	Submitted By
4218	Option to provide environment variable on command line	2004-Jul-23 16:03	None	stokes
4159	Upgrades mean sites are unavailable for long periods of time	2004-Jul-19 08:27	None	stokes
4084	Difficult to access LCG tools	2004-Jul-08 15:52	None	stokes
4053	Mystery job abortion	2004-Jul-07 10:03	None	stokes
3926	Log file format is not easily parseable	2004-Jun-28 14:43	None	stokes
3925	LCG lacks tools for users to efficiently managing large numbers of jobs	2004-Jun-28 14:37	None	stokes
3924	RB lxn1176 reinstalled from scratch	2004-Jun-28 14:24	None	stokes
3909	Cannot specify maximum run time for jobs	2004-Jun-25 18:01	None	stokes
3904	LCG queue times need to offer normalised value	2004-Jun-25 15:09	None	stokes
3902	Job working directories don't have enough space or are NFS mounted	2004-Jun-25 13:58	None	stokes
3873	EstimatedResponseTime values for site ranking are completely bogus	2004-Jun-24 11:51	None	stokes
3845	LCG sites going down with little notice	2004-Jun-22 13:44	None	stokes
3838	Jobs aborted with message "canceled by user", but user didn't cancel them	2004-Jun-22 09:54	None	stokes
3831	Submitting about 1000 jobs a day is too much for the RB	2004-Jun-21 12:57	None	stokes
3822	Fuzzy selection to avoid all jobs going to same site is broken	2004-Jun-19 07:50	None	stokes
3789	std out and std err not returned on aborted/canceled jobs	2004-Jun-17 12:45	None	stokes
3785	RB restarting jobs in progress	2004-Jun-17 10:54	None	stokes
3764	Proxy expiry	2004-Jun-16 07:52	None	stokes

Items 1 to 18

CERN

LHCb

Savannah
Bugs
Support
CVS
Downloads
Documentation & Publications
Mailing Lists
FAQ
Twiki
Team
Supporters
LHCb DC04
GridSite Guide

LHCb Data Challenge 2004

The Data Challenge has four goals:

1. Produce 60 TB of Monte Carlo simulation data for later analysis
2. Excercise the LHCb software tools and computing infrastructure
3. Validate the useability of LCG computing resources
4. Demonstrate the ability to perform distributed analysis

There are 20 sites explicitly participating in the Data Challenge, with a further 40 sites participating indirectly through LCG.

Participants

Monitoring and Accounting

Production System:

Monitoring: http://fpegaes1.usc.es/dmon/DC04/joblist.html
Accounting: http://lhcb.ecm.ub.es/DC04/Accounting/
Backup Monitoring: http://lhcb02.usc.cesga.es/dmon/DC04/joblist.html

Test System:

Monitoring: http://fpegaes1.usc.es/dmon/DC04test/joblist.html
Accounting: http://lhcb.ecm.ub.es/DC04test/Accounting/
DC04 Summary Tables: http://lbnts3.cern.ch:8110/BkkSummary?-summary=index.htm

---

Last modified Tue 31 August 2004 . View page history

You are /C=UK/O=eScience/OU=Oxford/L=OeSC/CN=Ian Stokes-rees
Edit page . Manage directory . Switch to HTTP . Website Help . Print View . Built with GridSite 1.0.4

Web Based Job Monitoring

GridPP
UK Computing for Particle Physics

Details

Production ID:
All

Site:
All
ANY
CERN
CERN.tsar-agent.ch
DIRAC.atsareg-cern.ch

Job Status:
All

App Status:
All
Boole execution, step 4
Boole, step 4 done
Brunel execution, step 3

Owner:
All

Max results: 40

Job Ids:

Jobs updated after: 25/08/2004

Submit Reset

LHCb DC'04 Monitoring (test)

Showing 40 out of 1271 jobs.

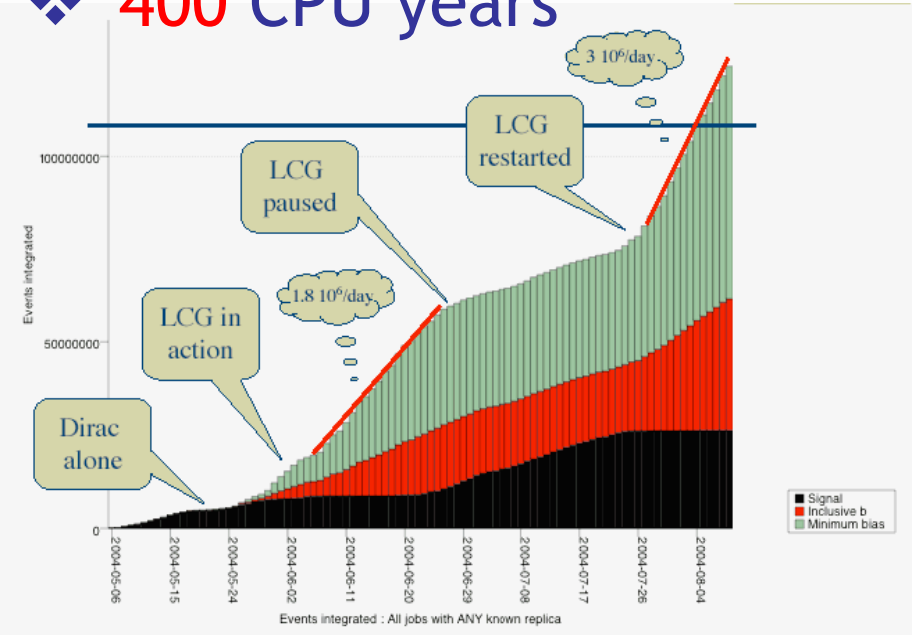
Pages: 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32

JobId	JobStatus	AppState	Site	JobName	Last Update	Owner
8854	waiting	Unknown	LCG.TAU.il	00005615_00000033	2004-09-01 11:53:28	lhcbprod
8855	waiting	Unknown	LCG.TAU.il	00005615_00000034	2004-09-01 11:53:29	lhcbprod
8856	waiting	Unknown	LCG.TAU.il	00005615_00000035	2004-09-01 11:53:31	lhcbprod
8857	waiting	Unknown	LCG.TAU.il	00005615_00000036	2004-09-01 11:53:33	lhcbprod
8858	waiting	Unknown	LCG.TAU.il	00005615_00000037	2004-09-01 11:53:35	lhcbprod
8859	waiting	Unknown	LCG.TAU.il	00005615_00000038	2004-09-01 11:53:37	lhcbprod
8860	waiting	Unknown	LCG.TAU.il	00005615_00000039	2004-09-01 11:53:38	lhcbprod
8861	waiting	Unknown	LCG.TAU.il	00005615_00000040	2004-09-01 11:53:40	lhcbprod

Results of DC04

- Typical Job
  - 2 GB local storage
  - 300-600 MB transferred at end
  - 15-24 hours execution

- 58 TB of data produced
- 175M events
- 50+ sites
- 400 CPU years



Future

- Increased integration with LCG
- Investigation of gLite
- May look at WSRF and GT4 (no promises)
- Expose services via Apache, mod_python, and mod_gridsite
- (much) better security mechanisms
- Explore Instant Messaging opportunities

GridPP
UK Computing for Particle Physics

Questions?

For further information on DIRAC and LHCb:

GridSite:

http://dirac.cern.ch

email:

Ian Stokes-Rees

i.stokes-rees1@physics.ox.ac.uk

... or talk to me at the break

DIRAC and the results from DC04 are the result of many peoples efforts and the support of numerous participating institutions: