

Material for the GANGA demo

EGEE Review

Based on K. Harrison at the Computing in High-Energy and Nuclear Physics conference
Mumbai, India, 13-17 February 2006

New material from the GANGA team and from Hurng-Chun Lee (ASGC)

Other material from IT/PSS/ED (ARDA and EIS teams)

Ganga basics

- Ganga is an easy-to-use frontend for job definition and management
 - ▶ Access to local and remote (Grid) resources through a uniform interface
 - The Grid is *one* of the environment for the scientists
 - If the Grid is not integrated in the everyday environment, its impact is reduced
 - ▶ Developed in the context of ATLAS and LHCb
 - built-in support for applications based on Gaudi/Athena framework
 - Potentially interesting for other applications (also non HEP)
 - ▶ Component architecture readily allows extension to other user groups
 - ▶ Implemented in Python

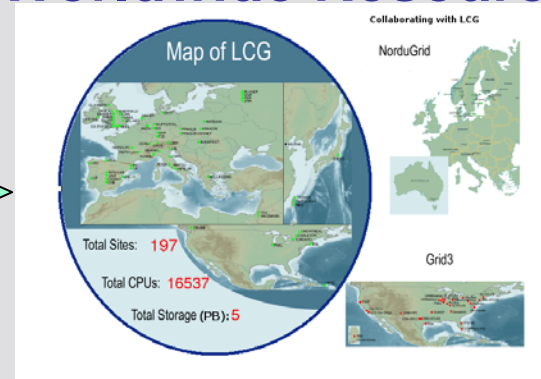
HEP Computing



Uniform User Interface



Worldwide Resources



Gaudi/Athena and Grid Alliance

People/groups involved

- Ganga is an ATLAS/LHCb joint project



- Support for development work from UK (PPARC/GridPP) and EU (EGEE/NA4 HEP ARDA)



- Core team:

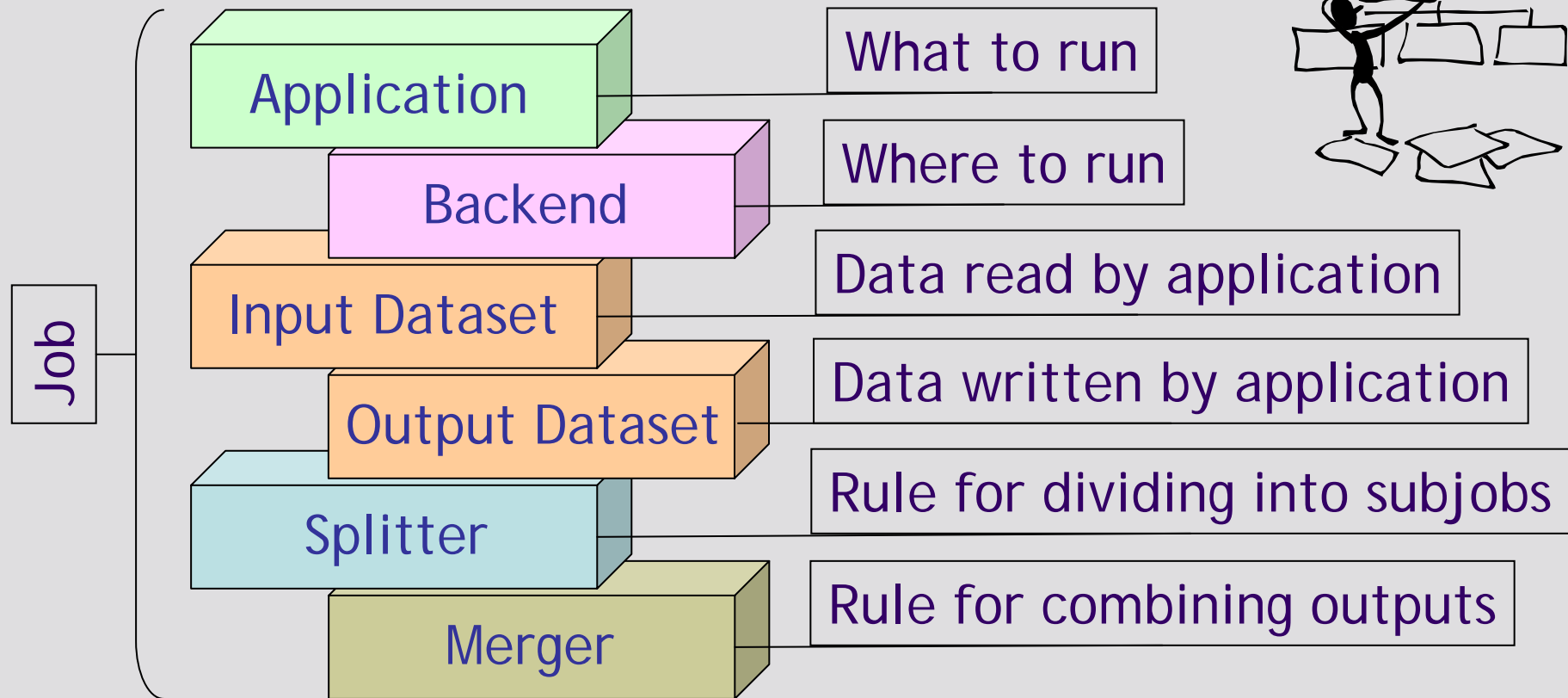
▶ U.Egede (Imperial), K.Harrison (Cambridge),
D.Liko (CERN), A.Maier (CERN), J.T.Moscicki (CERN),
A.Soroko (Oxford), CL.Tan (Birmingham)



- Contributions from many others, from summer students to senior researchers

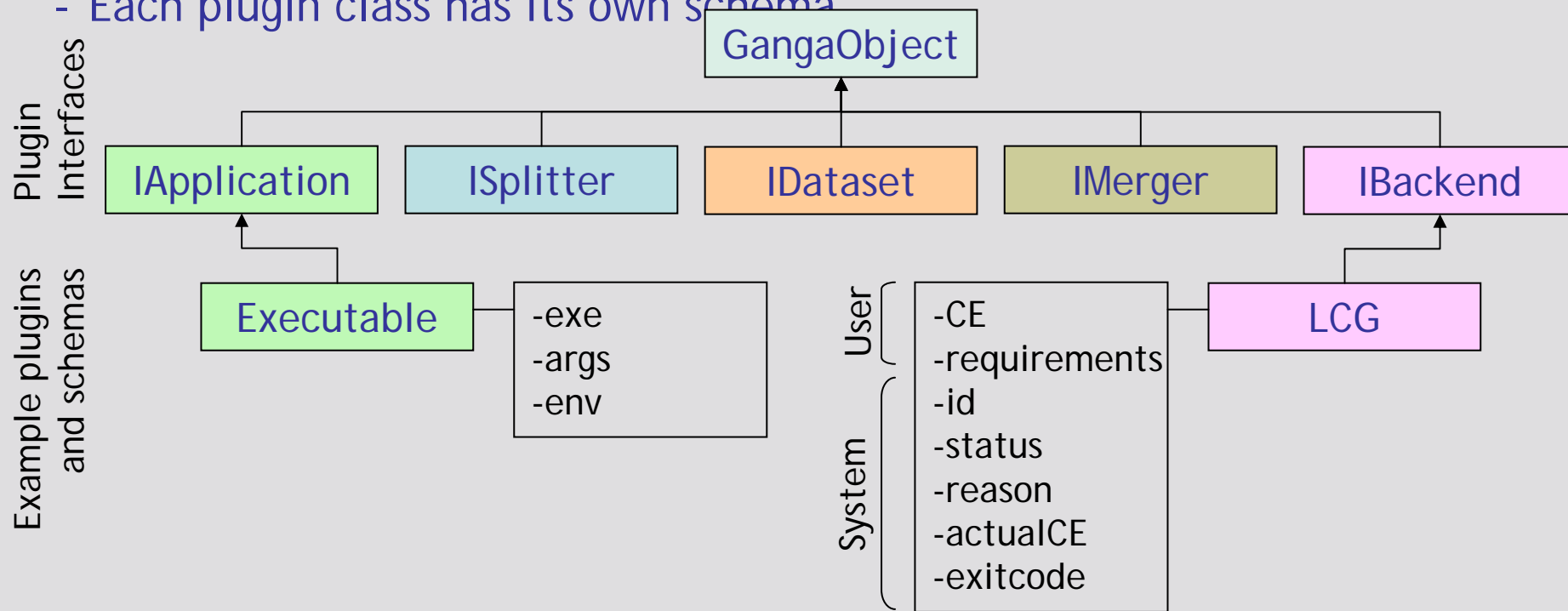
Ganga job abstraction

- A job in Ganga is constructed from a set of building blocks, not all required for every job

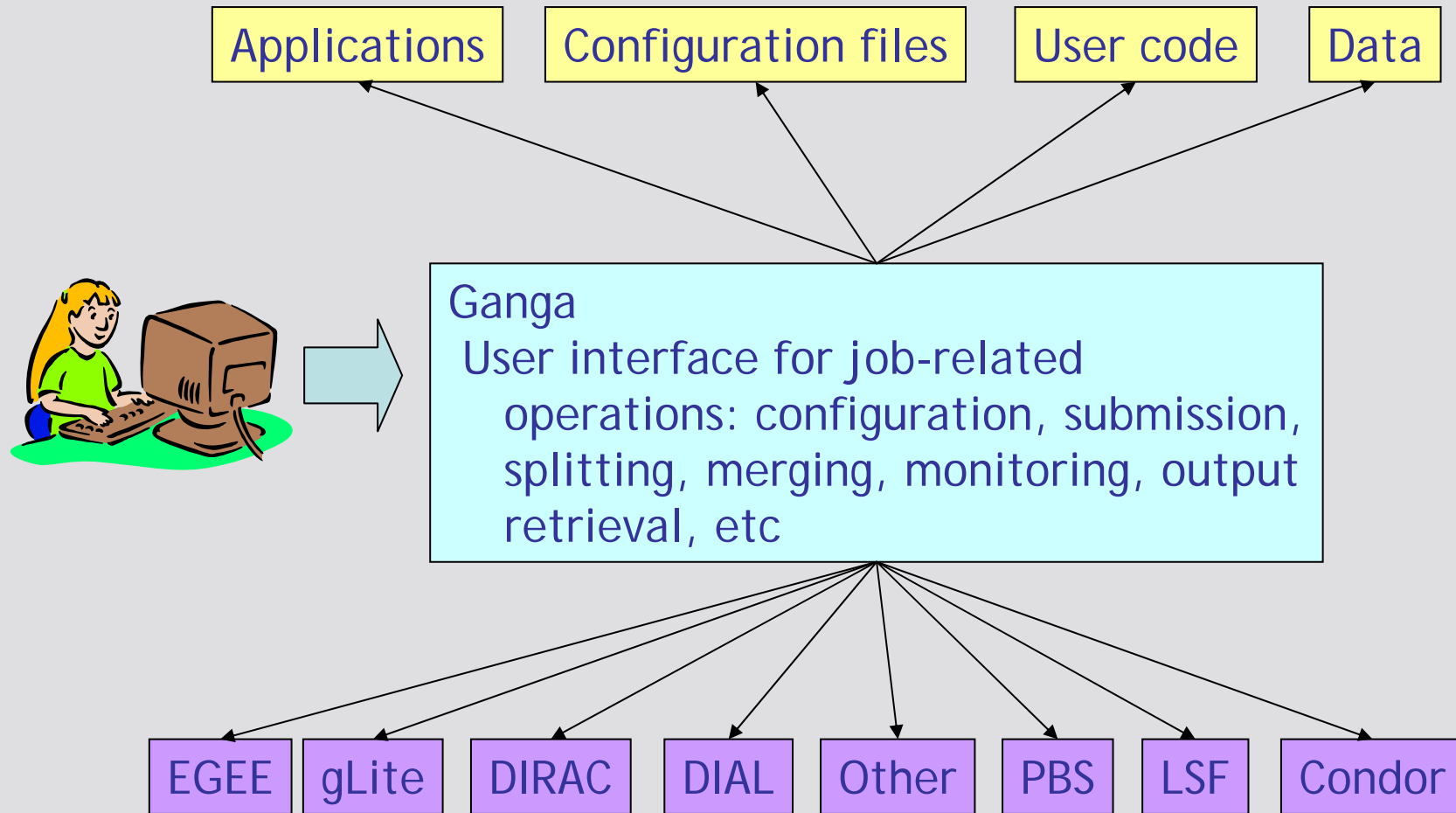


Framework for plugin handling

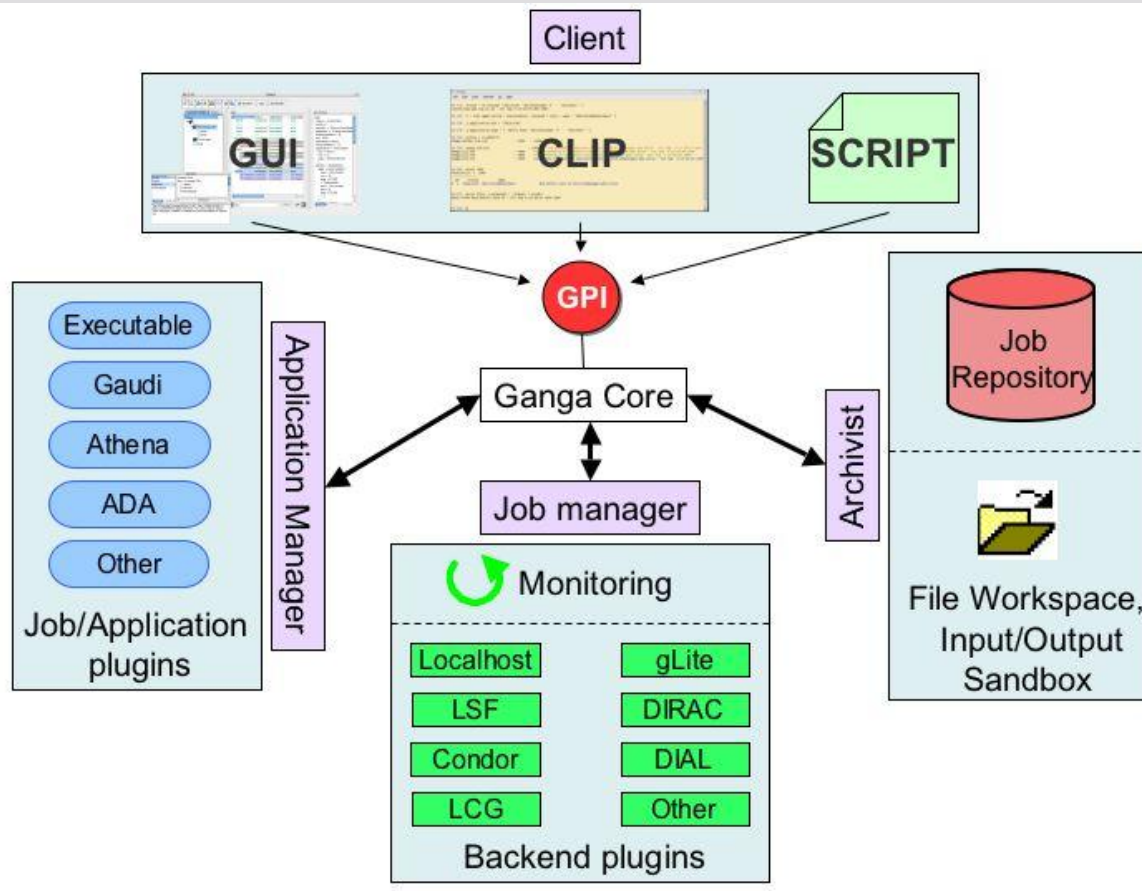
- Ganga provides a framework for handling different types of Application, Backend, Dataset, Splitter and Merger, implemented as plugin classes
- Each plugin class has its own schema



Ganga: single frontend for multiple backends



Design



- Application Manager deals with application configuration
- Job Manager deals with submission to backend
- Archivist records job states and allocates/deallocates workspace
- Core performs startup operations, and mediates communication between components
- Functionality made available through Ganga Public Interface (GPI)
- User accesses GPI through Client

Ganga client (1)

```
IPython Shell [Command Line]
In [1]: status = ex.execute("/bin/echo /bin/helloworld -f - /bin/date")
out[1]: /bin/echo /bin/helloworld -f - /bin/date
In [2]: j = Job(application = helloworld(), backend = XROOT, name = "helloworldtest")
In [3]: j.application.args = "/bin/echo"
In [4]: j.application.args = "Hello from /bin/helloworld -f - /bin/date"
In [5]: status = j.submit()
Ganga-01000-130 Job      : INFO   submitting job 1
In [6]: Ganga-010-130    : INFO   Job 1 Ready at met.metria.sera.nl:2118/jobmanager-gba-afaa - Fri Sep  8 12:00:13 2006
Ganga-010-130    : INFO   Job 1 Scheduled at met.metria.sera.nl:2118/jobmanager-gba-afaa - Fri Sep  8 12:00:18 2006
Ganga-010-130    : INFO   Job 1 Running at met.metria.sera.nl:2118/jobmanager-gba-afaa - Fri Sep  8 12:00:20 2006
Ganga-010-130    : INFO   Job 1 Done (Success) at met.metria.sera.nl:2118/jobmanager-gba-afaa - Fri Sep  8 12:01:26 2006
In [7]: print Job
Out[7]: Job
-----
# 1 completed helloworldtest
met.metria.sera.nl:2118/jobmanager-gba-afaa
In [8]: print $!cat /dev/stdin < "helloworld" | read()
Hello from met.metria.sera.nl - Fri Sep  8 12:01:27 CEST 2006
In [9]:
```

to GPI commands in any of three ways

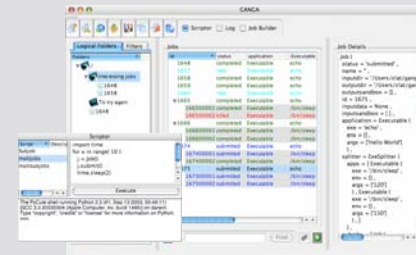


- Command-Line Interface in Python (CLIP)

- ▶ Interactive job definition and submission from enhanced Python shell (IPython)
- ▶ Possibility to organise jobs in logical folders
- ▶ Possibility to create job templates
- ▶ Possibility to export jobs as Python code, which can be loaded back into Ganga: allows sharing with others

Ganga client (2)

- GPI scripting
 - ▶ Python/GPI script `myScript.py` executed in Ganga environment with: `ganga myScript.py`
 - ▶ Possibility to automate repetitive tasks
 - ▶ Scripts for basic operations (submit, query, kill) included in distribution



- Graphical User Interface (GUI)

- ▶ Job definition based on selection with mouse and form completion
- ▶ Graphical monitoring and navigation of logical folders
- ▶ Integrated scriptor, allowing execution of arbitrary GPI commands

CLIP: simple job from first principles

```
Ganga
In [1]: status = os.system( "/bin/echo ` /bin/hostname -f` - ` /bin/date`" )
pcff.hep.phy.cam.ac.uk - Sat Feb 11 09:17:39 GMT 2006

In [2]: j1 = Job( application = Executable(), backend = LCG(), name = "Hello" )

In [3]: j1.application.exe = "/bin/echo"

In [4]: j1.application.args = [ "Hello from ` /bin/hostname -f` - ` /bin/date`" ]

In [5]: status = j1.submit()
Ganga: INFO      submitting job 1

In [6]: Ganga: INFO      Job 1 Waiting at None - Sat Feb 11 09:18:14 2006
Ganga: INFO      Job 1 Ready at lcgce01.nic.ualberta.ca - Sat Feb 11 09:18:30 2006
Ganga: INFO      Job 1 Scheduled at lcgce01.nic.ualberta.ca - Sat Feb 11 09:19:01 2006
Ganga: INFO      Job 1 Running at lcgce01.nic.ualberta.ca - Sat Feb 11 09:22:26 2006
Ganga: INFO      Job 1 Done (Success) at lcgce01.nic.ualberta.ca - Sat Feb 11 09:29:49 2006

In [6]: !cat $j1.outputdir/stdout
Hello from thuner069.ualberta - Sat Feb 11 02:20:05 MST 2006

In [7]: []
```

Time on local machine (Cambridge)

Create job

Specify executable

Specify arguments

Submit job to LCG

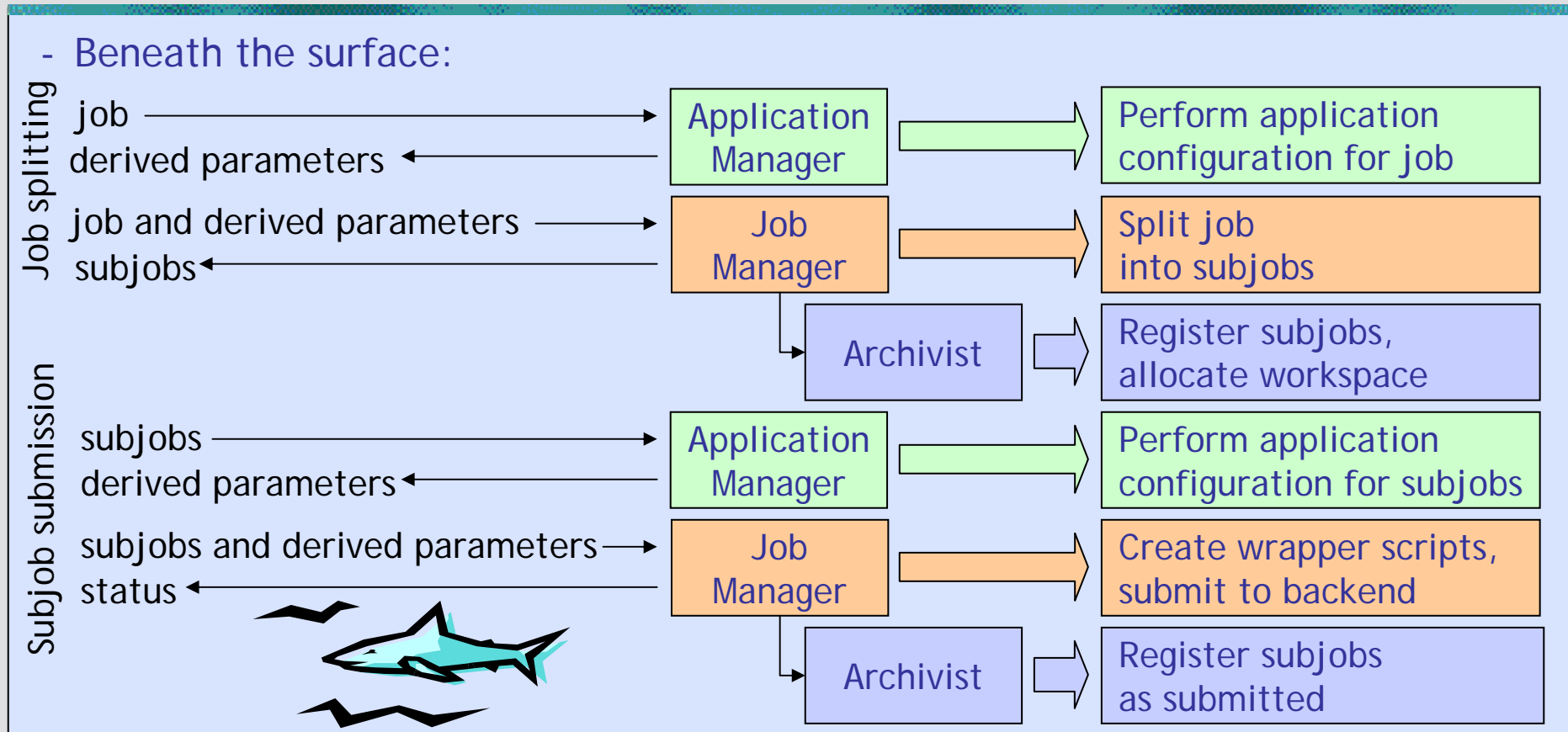
Status on LCG monitored

IPython allows combining of shell and Python syntax

Time on remote machine (Alberta)

Job submission

- User command: `job.submit()`
- Outcome: job submitted, split into subjobs, command status returned



Job cancellation

- User command: `job.remove()`
- Outcome: job/subjobs killed if submitted/running, records deleted from repository, workspace deallocated, command status returned

- Beneath the surface:

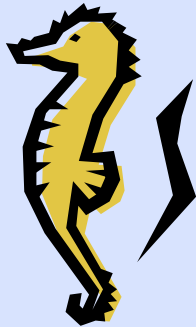
job
status

Job
Manager

Query status of job/subjob,
kill if submitted/running

Archivist

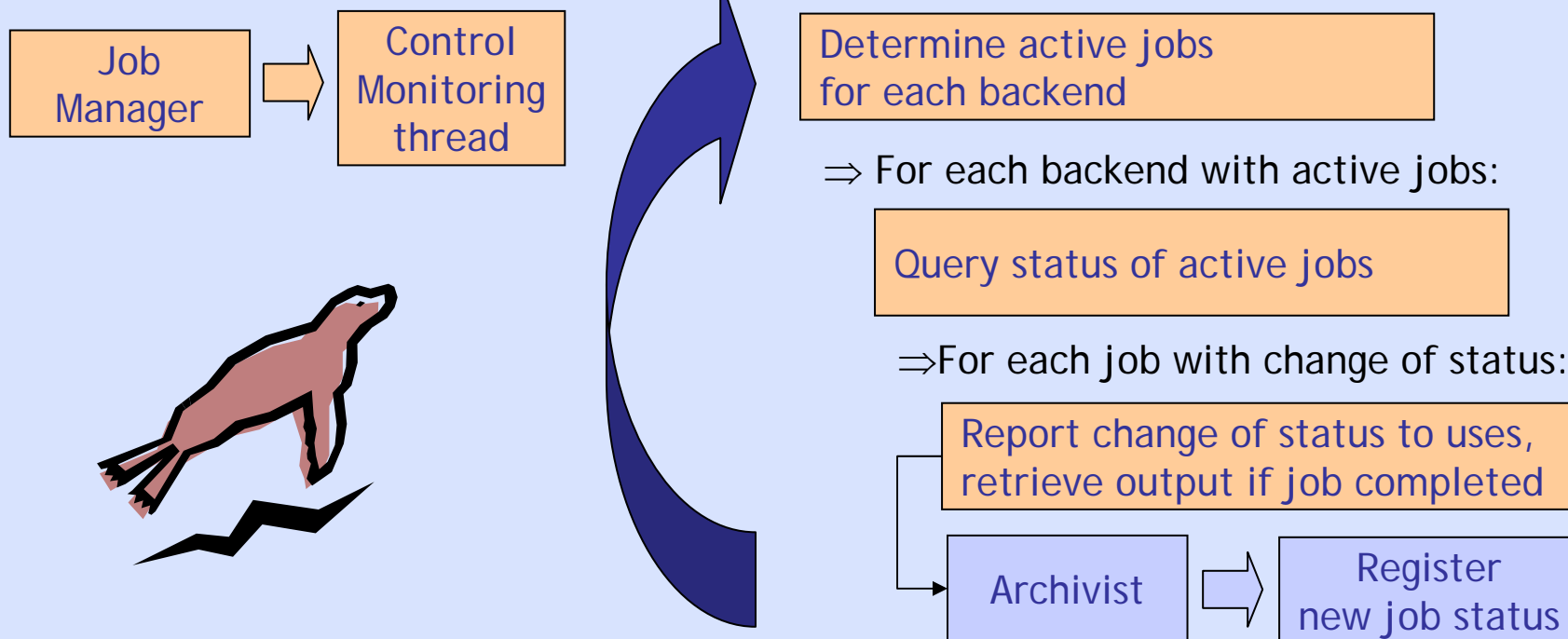
Delete job/subjobs from repository,
deallocate workspace



Job monitoring and output retrieval

- User action: none
- Outcome: changes in job status reported/updated periodically, output retrieved automatically when job completes

- Beneath the surface:



GPI job script

```
Bu2D0K.py
#Ganga# File created by Ganga - Tue Feb  7 21:19:22 2006

#Ganga# Job object (category: jobs)
Job (
  name = 'Bu2D0K' ,
  outputsandbox = [ ] ,
  splitter = None ,
  inputsandbox = [ ] ,
  application = DaVinci (
    version = 'v12r15' ,
    extraopts = "'EventSelector.Input = {
  \"DATAFILE=\\LFN:/lhcb/production/DC04/v2/00980000/DST/Presel-
00980000_00001212.dst\\' TYP=\\POOL_ROOTTREE\\' OPT=\\READ\\'\"
}'" ,
    package = 'Phys' ,
    cmt_user_path = '/usera/cristina/cmtuser' ,
    masterpackage = 'PhysSel/Bu2D0K/v1r1p2' ,
    optsfile = File (
      name = '/lhcb/cristina/cmtuser/PhysSel/Bu2D0K/v1r1p2/op
tions/Bu2D0K_test1_bb.opts' ,
      subdir = '.'
    )
  ) ,
  inputdata = None ,
  backend = LCG (
    CE = None ,
    requirements = Requirements (
      other = [ ] ,
      memory = None ,
      software = [ ] ,
      ipconnectivity = None ,
      cputime = None ,
      walltime = None
    )
  )
)
)
1,1 Top
```

GPI job script written by Ganga (export function)

Application is DaVinci (v12r15): LHCb analysis

Input data passed as job option

Route to user packages

Application directives (job-options file)

Backend is LCG

Requirements to be passed to LCG resource broker

Job can be submitted with
ganga submit Bu2D0K.py
or loaded back into Ganga

Ganga GUI

GUI based on dockable windows

Logical Folders

Scriptor

The screenshot shows the Ganga GUI interface with several dockable windows. At the top, there are tabs for 'Scriptor', 'Log', and 'Job Builder'. The 'Logical Folders' window shows a tree view with folders like 'Interesting jobs' and 'To try again'. The 'Jobs' window displays a table of job status and details. The 'Job Details' window shows a detailed view of a job's configuration. The 'Scriptor' window shows a Python script being executed in a shell.

id	status	application	Executable
1648	completed	Executable	echo
1657	new	Executable	echo
1658	completed	Executable	echo
1659	completed	Executable	echo
1660	new	Executable	echo
1665	completed	Executable	echo
166500001	completed	Executable	/bin/sleep
166500002	killed	Executable	/bin/sleep
1666	completed	Executable	echo
166600001	completed	Executable	/bin/sleep
166600002	completed	Executable	/bin/sleep
1674	submitted	Executable	echo
167400001	submitted	Executable	/bin/sleep
167400002	completed	Executable	/bin/sleep
1675	submitted	Executable	echo
167500001	submitted	Executable	/bin/sleep
167500002	submitted	Executable	/bin/sleep

```
Job (
  status = 'submitted',
  name = "",
  inputdir = '/Users/clat/ganga',
  outputdir = '/Users/clat/ganga',
  outputsandbox = [],
  id = 1675,
  inputdata = None,
  inputsandbox = [],
  application = Executable (
    exe = 'echo',
    env = {},
    args = ['Hello World']
  ),
  splitter = ExeSplitter (
    apps = [ Executable (
      exe = '/bin/sleep',
      env = {},
      args = ['120']
    ), Executable (
      exe = '/bin/sleep',
      env = {},
      args = ['150']
    ), ]
  ), ]
)
```

```
import time
for x in range( 10 ):
    j = Job()
    j.submit()
    time.sleep(2)
```

The PyCute shell running Python 2.3 (#1, Sep 13 2003, 00:49:11) [GCC 3.3 20030304 (Apple Computer, Inc. build 1495)] on darwin. Type "copyright", "credits" or "license" for more information on Python. >>>

Job details

Job Monitor

Splitter

CLIP: running LHCb analysis on LCG and DIRAC

```
Ganga
In [7]: j2 = load( "Bu2D0K.py" )[ 0 ]
In [8]: status = j2.submit()
Ganga: INFO      submitting job 2
In [9]: Ganga: INFO      Job 2 Ready at t2-ce-02.inl.infn.it - Sat Feb 11 09:33:57 2006
Ganga: INFO      Job 2 Scheduled at t2-ce-02.inl.infn.it - Sat Feb 11 09:34:29 2006
Ganga: INFO      Job 2 Running at t2-ce-02.inl.infn.it - Sat Feb 11 09:36:52 2006
Ganga: INFO      Job 2 Done (Success) at t2-ce-02.inl.infn.it - Sat Feb 11 09:48:02 2006
In [9]: !ls $j2.outputdir
DVHistos.root  DVTuple.root  stderr  stdout
In [10]: j3 = j2.copy()
In [11]: j3.backend = Dirac()
In [12]: status = j3.submit()
Ganga: INFO      submitting job 3
In [13]: Ganga: INFO      Job 3 (Dirac Job 10907) waiting at ANY - Sat Feb 11 09:50:30 2006
Ganga: INFO      Job 3 (Dirac Job 10907) matched at LCG.CERN.ch - Sat Feb 11 09:52:48 2006
Ganga: INFO      Job 3 (Dirac Job 10907) running at LCG.CERN.ch - Sat Feb 11 09:55:52 2006
Ganga: INFO      Job 3 (Dirac Job 10907) outputready at LCG.CERN.ch - Sat Feb 11 10:10:39 2006
In [13]: !ls $j3.outputdir/*
DaVinci_v12r15.log  DVHistos.root  DVTuple.root  std.err  std.out
In [14]: []
```

Load job from GPI script

Submit job to LCG

Status in LCG monitored

Output retrieved automatically from LCG

Copy job

Set backend to DIRAC

Submit job to DIRAC

Status in DIRAC monitored

Output retrieved automatically from DIRAC

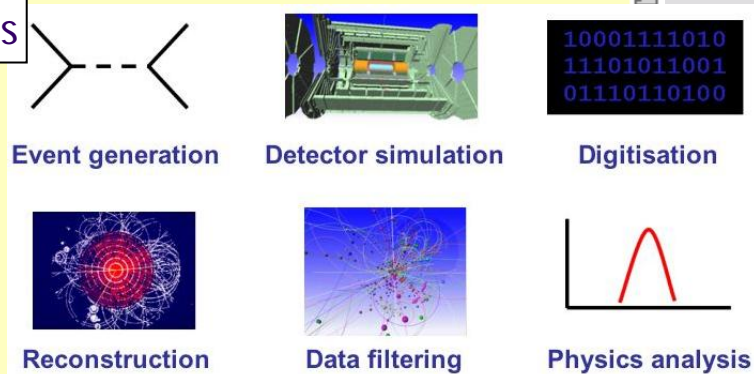
CLIP: creating ATLAS analysis job from template

New session started

```
In [1]: templates
Out[1]: Statistics: 7 templates
```

List templates

#	id	status	name	application	backend
#	1	template	Hello	Executable	LCG
#	2	template	EvGen	Athena	LCG
#	3	template	Simul	Athena	LCG
#	4	template	Digi	Athena	LCG
#	5	template	Reco	Athena	LCG
#	6	template	ESDtoAOD	Athena	LCG
#	7	template	SusyPlot	Athena	LCG



```
In [2]: j4 = Job( templates[ 7 ] )
```

Create job from template

```
In [3]: j4.application
```

List application properties

```
Out[3]: Athena (
  atlas_area = '',
  packages = ['PhysicsAnalysis/SUSYPhys/SUSYPhysAlgs/SUSYPhysAlgs-00-04-02/', 'PhysicsAnalysis/SUSYPhys/SUSYPhysUser/SUSYPhysUser-00-04-01/', 'PhysicsAnalysis/SUSYPhys/SUSYPhysUtils/SUSYPhysUtils-00-04-01/'],
  pool_catalog = '',
  atlas_release = '10.0.2',
  option_files = ['/afs/cern.ch/user/h/harrison/public/gangainputs/SusyPlot_joboptions.py'],
  options = [],
  user_area = '/afs/cern.ch/user/h/harrison/cmtuser',
  evt_max = -1,
  outputdata = None
)
```

User packages

Application directives
(job-options file)

CLIP: running ATLAS analysis on LCG

```
Ganga
In [4]: status = j4.submit()
Ganga: INFO submitting job 4
Submit job to LCG

In [5]: Ganga: INFO Job 4 Waiting at None - Sat Feb 11 11:57:59 2006
Ganga: INFO Job 4 Ready at grid10.lal.in2p3.fr - Sat Feb 11 11:58:14 2006
Ganga: INFO Job 4 Scheduled at grid10.lal.in2p3.fr - Sat Feb 11 11:58:30 2006
Ganga: INFO Job 4 Running at grid10.lal.in2p3.fr - Sat Feb 11 11:59:49 2006
Ganga: INFO Job 4 Done (Success) at grid10.lal.in2p3.fr - Sat Feb 11 12:08:17 2006
Status in LCG monitored

In [5]: !ls $j4.outputdir
gangademo_susyplot_hist.root stderr stdout
Output retrieved automatically from LCG

In [6]: jobs
Out[6]: Statistics: 4 jobs
List jobs

-----
# id    status    name    application    backend    backend.status    backend.actualLCE
# 1    completed    Hello    Executable    LCG    Done (Success)    lcgce01.nic.ualberta.ca
# 2    completed    Bu2D0K    DaVinci    LCG    Done (Success)    t2-ce-02.lnl.infn.it
# 3    completed    Bu2D0K    DaVinci    Dirac    outputready    LCG.CERN.ch
# 4    completed    SusyPlot    Athena    LCG    Done (Success)    grid10.lal.in2p3.fr
Jobs from previous session(s) remembered

In [7]: []
```

Ganga and other applications on the Grid

- GEANT4
 - ▶ Presented at the User Forum
- EGEE - ITU collaboration
 - ▶ Presented at the User Forum
 - ▶ Running in these days (May 15th - June 16th)
- Autodock application
 - ▶ Presented at the User Forum

We need one foil each

User uptake

- Ganga tutorials organised for both ATLAS and LHCb
- Ganga demo at EGEE 06, other events???? Rio etc... , User Forum



CERN, September 2005



Cambridge, January 2006

- Ganga tried out by more than 100 people
 - Feedback positive
 - ▶ “Very handy way to organise job submission” (ATLAS user)
 - ▶ “Clever and nicely designed” (LHCb user)
 - Small but growing group of people regularly using Ganga (also from a laptop)
 - ▶ LHCb analyses of up to 10^6 events run successfully on Grid using Ganga
- ⇒ More on use of Ganga in ATLAS and LHCb in presentations by D.Liko and U.Egede