

# Introduction to Ganga

Belle II Computing Workshop

Thursday 17 June 2010

HanGi Kim, KISTI



# What is Ganga?

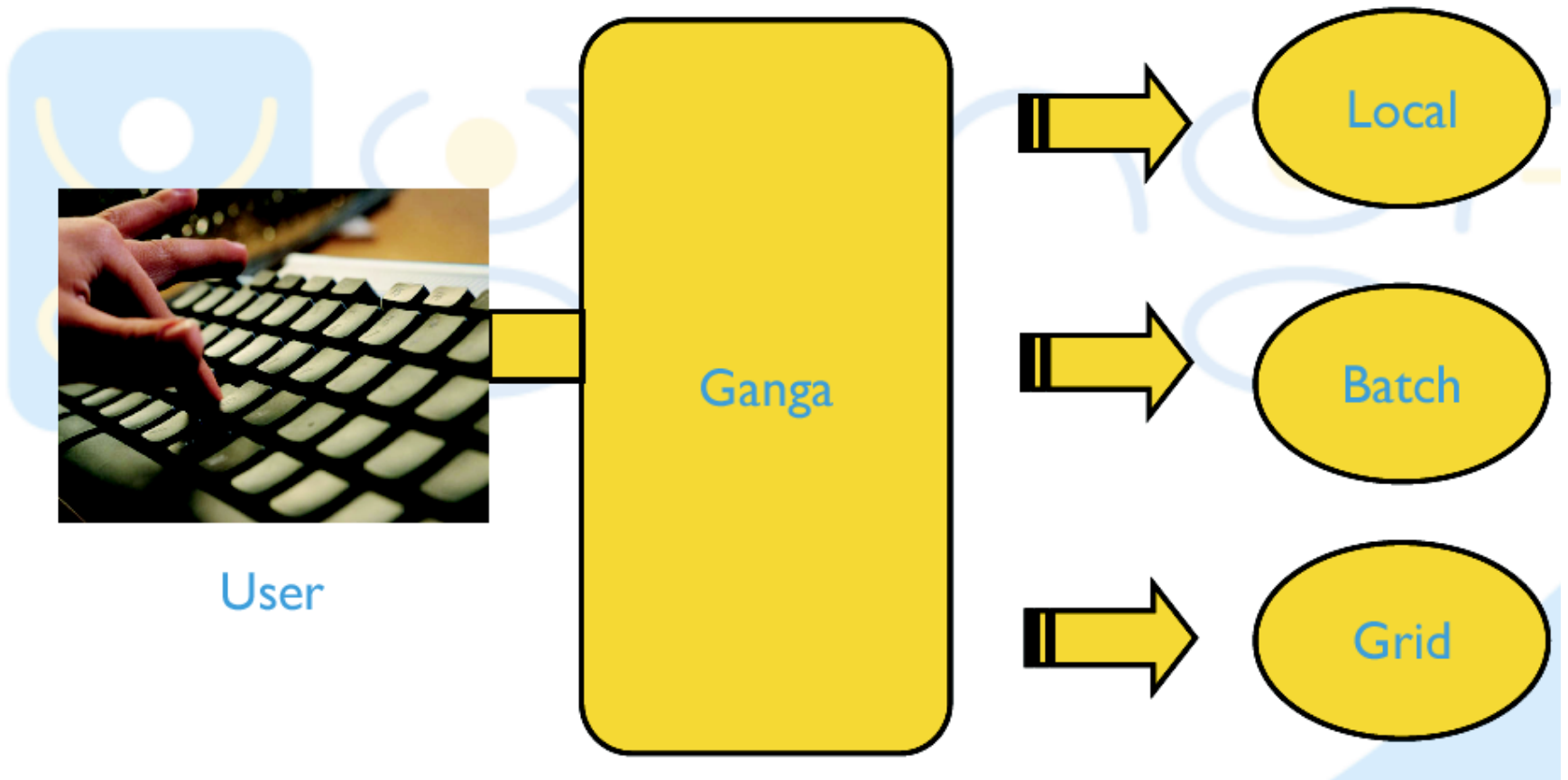
- A user-friendly yet powerful job definition and management tool written by Python
- Started as an ATLAS/LHCb project
- Ganga is an application that enables a user to

**Configure – Prepare – Submit – Monitor**

Jobs and applications using a variety of resources



# The Ganga Mantra



Configure once, run anywhere!



# Possible resources

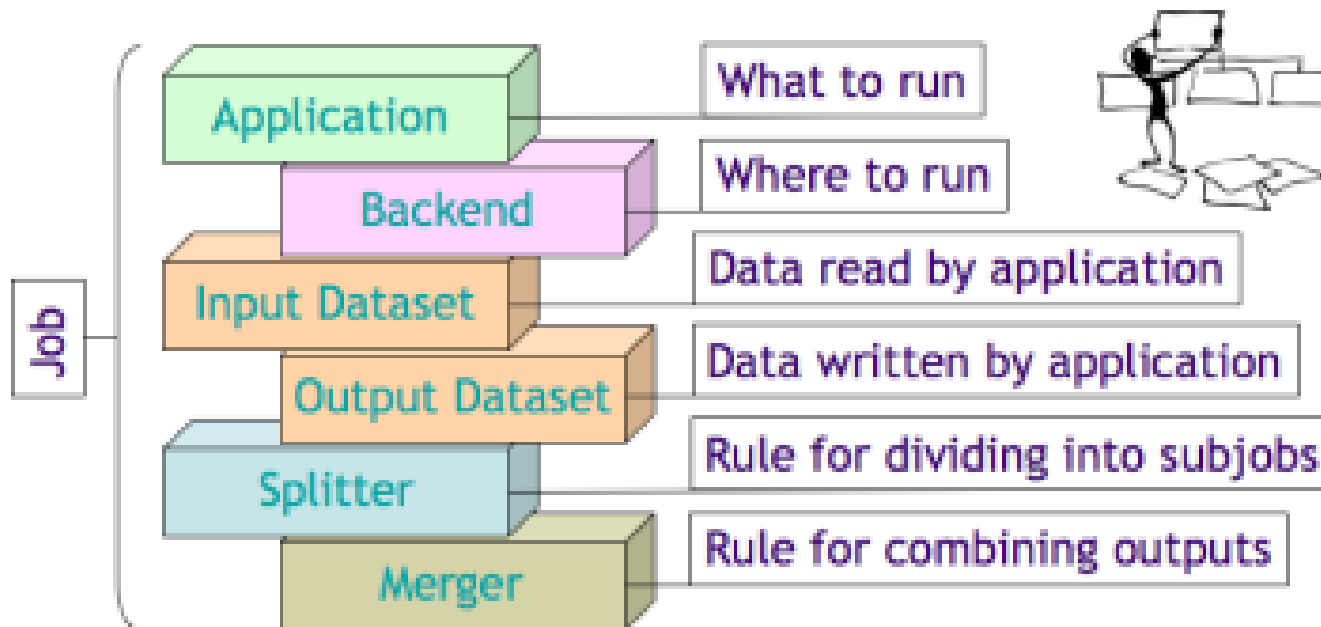
- The **local machine** (interactive or in background)
- **Batch** systems (LSF, PBS, SGE, Condor)
- **Grid** systems (LCG, gLite, NorduGrid)
- **Workload management** systems (**DIRAC**, PanDA)

Jobs look the same whether they run locally or on the Grid



# Ganga Job Object

- Ganga is developed in Python with a modular architecture
- A "Job" is composed of six modules (some of which are optional):



# Ganga applications

- In [7]:plugins('applications')
- Out[7]: ['Autodock', 'Executable', 'Root']



# LHCb Plug-ins applications

- In [3]:plugins('applications')
- Out[3]: ['Boole', 'Executable', 'GaudiPython', 'Brunel', 'Moore', 'DaVinci', 'Panoptes', 'Gauss', 'Autodock', 'Gaudi', 'Francesc', 'Bender', 'Vetra', 'Root']



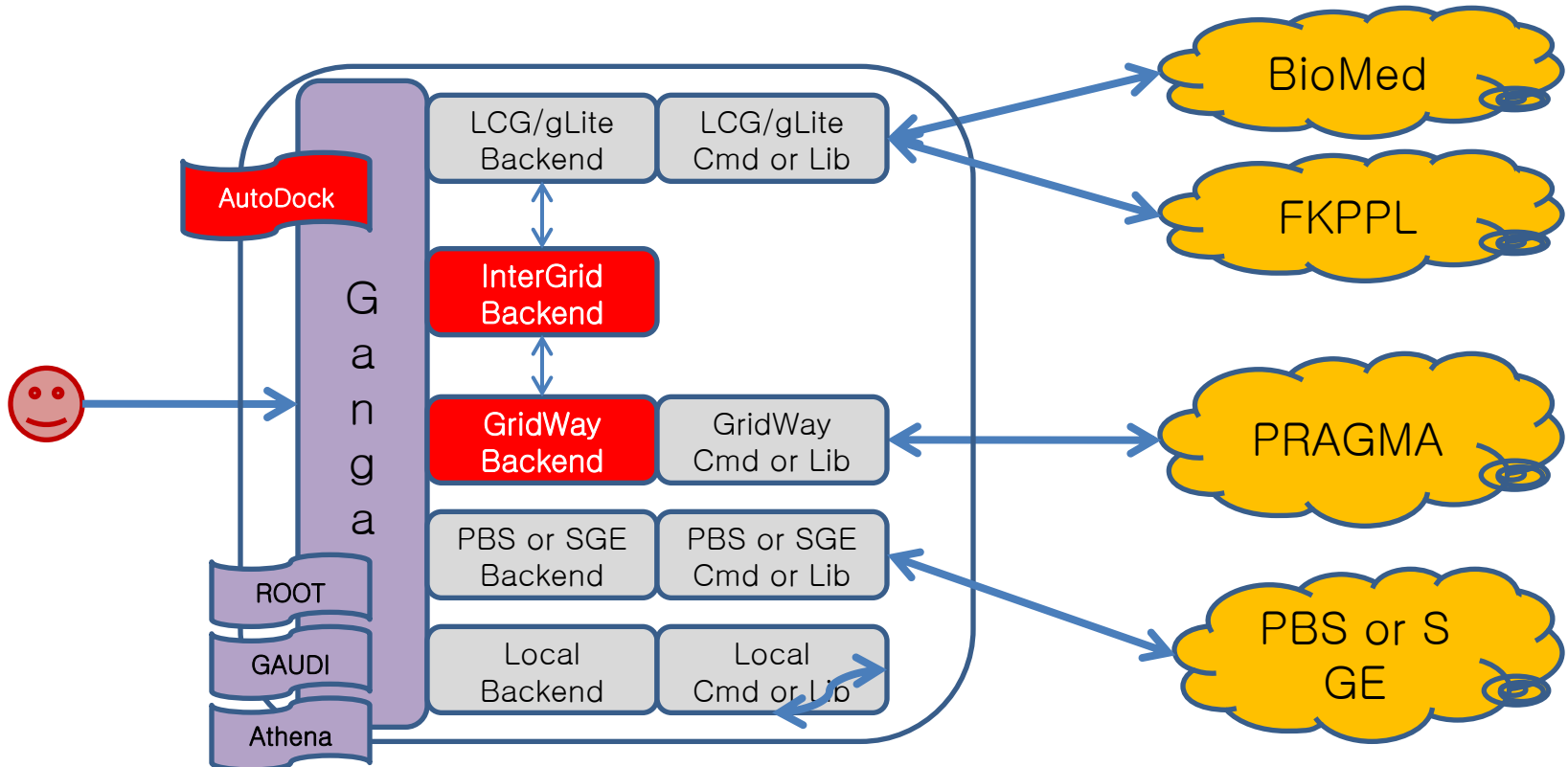
# Ganga backends

- In [4]:plugins('backends')
- Out[4]: ['LSF', 'Gridway', 'Remote', 'PBS', 'Condor', 'SGE', 'Batch', 'LCG', 'Dirac', 'InterGrid', 'Local', 'Interactive']





# InterGrid Backend



# Command Line Interface

- Ganga supports submission from the command line:

```
ganga athena \  
  --inDS data10_7TeV.00152409.physics_MinBias.merge.AOD.f238_m427 \  
  --outputdata histos.root \  
  --lcg \  
  AnalysisSkeleton_topOptions.py
```



# IPython Interface

- Start ganga without arguments and you get the IPython prompt:

```
j = Job()

j.application=Athena()
j.application.option_file=['AnalysisSkeleton_topOptions.py' ]
j.application.prepare()

j.inputdata=DQ2Dataset()
j.inputdata.dataset="data10_7TeV.00152409.physics_MinBias.merge.AOD.f238_m427"

j.outputdata=DQ2OutputDataset()

j.splitter=DQ2JobSplitter()

j.backend=Panda()

j.submit()
```



# Ganga: Help+Debug

```
# ganga has online help
```

```
help('index') # See all the ganga objects
```

```
j = Job(backend=Dirac(), application=Root())
```

```
# How do I set the CPU time for my job?
```

```
help(j.backend) # I use CPUTime!
```

```
# ganga (unlike python) is typesafe
```

```
j.application.version = 51800 # WILL FAIL!!!
```

```
j.application.version = '5.18.00' # That's better!
```

```
# What methods?
```

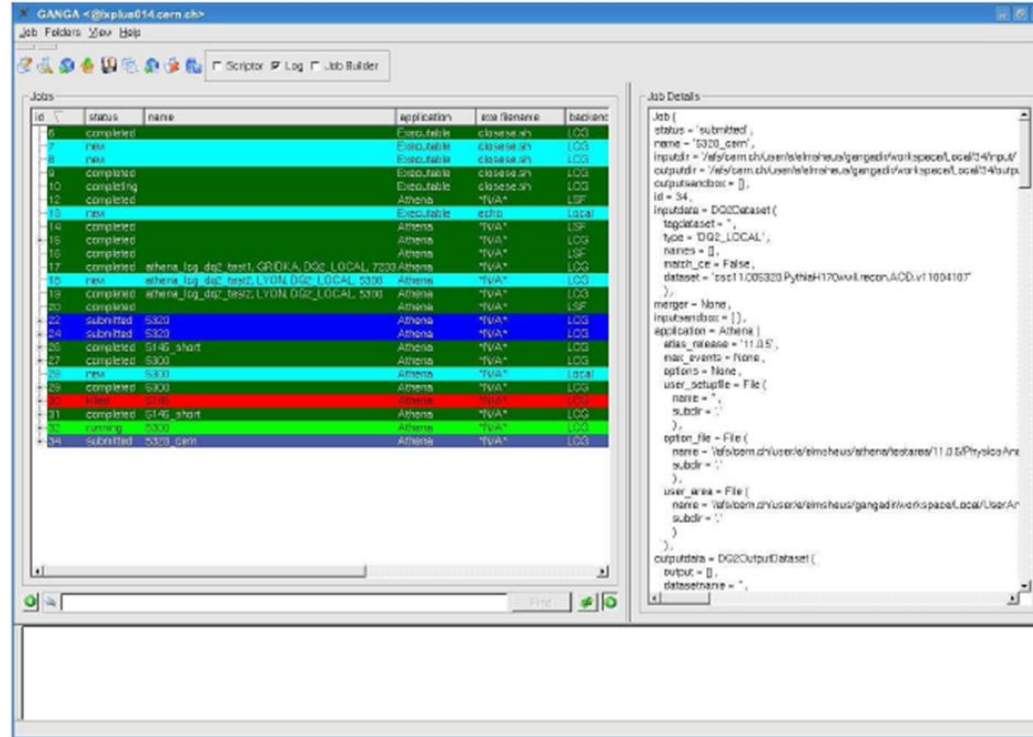
```
dir(j.backend)
```

```
help(j.backend.getOutput) # That's how I use it!
```



# Graphical Interface

- Run **ganga --gui** to get the graphical interface:



The screenshot displays the Ganga graphical interface. The main window is titled "GANGA <@xp104014.com>". It features a menu bar with "Job", "Folders", "View", and "Help". Below the menu bar are icons for "Scriptor", "Log", and "Job Builder". The central area is divided into two panes. The left pane, titled "Jobs", contains a table with columns for "id", "status", "name", "application", "job filename", and "basename". The right pane, titled "Job Details", shows a JSON-like structure representing the job configuration.

id	status	name	application	job filename	basename
7	completed		Eliso.fable	eliso.fable	LOG
7	fail		Eliso.fable	eliso.fable	LOG
8	fail		Eliso.fable	eliso.fable	LOG
9	completed		Eliso.fable	eliso.fable	LOG
10	completing		Eliso.fable	eliso.fable	LOG
12	completed		Athens	*TVA*	LOG
13	fail		Eliso.fable	eliso.fable	LOG
14	completed		Athens	*TVA*	LOG
15	completed		Athens	*TVA*	LOG
16	completed		Athens	*TVA*	LOG
17	completed	athens_lpg_dq2_test1_CRDJK_002_LOCAL_7230	Athens	*TVA*	LOG
18	fail	athens_lpg_dq2_test1_LYUJL_002_LOCAL_5330	Athens	*TVA*	LOG
19	completed	athens_lpg_dq2_test1_LYUJL_002_LOCAL_5330	Athens	*TVA*	LOG
20	completed		Athens	*TVA*	LOG
24	submitted	5320	Athens	*TVA*	LOG
24	submitted	5320	Athens	*TVA*	LOG
25	completed	5145_short	Athens	*TVA*	LOG
27	completed	5320	Athens	*TVA*	LOG
28	fail	5330	Athens	*TVA*	LOG
29	completed	5330	Athens	*TVA*	LOG
30	fail	5330	Athens	*TVA*	LOG
31	completed	5145_short	Athens	*TVA*	LOG
32	running	5330	Athens	*TVA*	LOG
34	submitted	5330_cant	Athens	*TVA*	LOG

```
Job Details
Job {
  status = 'submitted',
  name = '5320_cant',
  inputdir = '/afs/cern.ch/users/elm/haus/gangadriver/workspace/local/4/inputs',
  outputdir = '/afs/cern.ch/users/elm/haus/gangadriver/workspace/local/4/outputs',
  id = 34,
  inputData = DQ2Dataset(
    topdataset = '',
    type = 'DQ2_LOCAL',
    names = [],
    match_ce = False,
    dataset = 'test1.005320.Pythia4170wv1.recem.ACD.v1.1004107'
  ),
  merger = None,
  inputandbox = [],
  application = Athens |
  atlas_release = '11.2.5',
  mail_events = None,
  options = None,
  user_setupfile = File(
    name = '',
    subdir = ''
  ),
  ),
  optionfile = File(
    name = '/afs/cern.ch/users/elm/haus/athens/testarea/11.0.5/Physics/An',
    subdir = ''
  ),
  user_area = File(
    name = '/afs/cern.ch/users/elm/haus/gangadriver/workspace/local/userAr',
    subdir = ''
  ),
  ),
  outputdir = DQ2OutputDataset(
    output = [],
    datasetname = ''
  )
}
```



# Conclusion

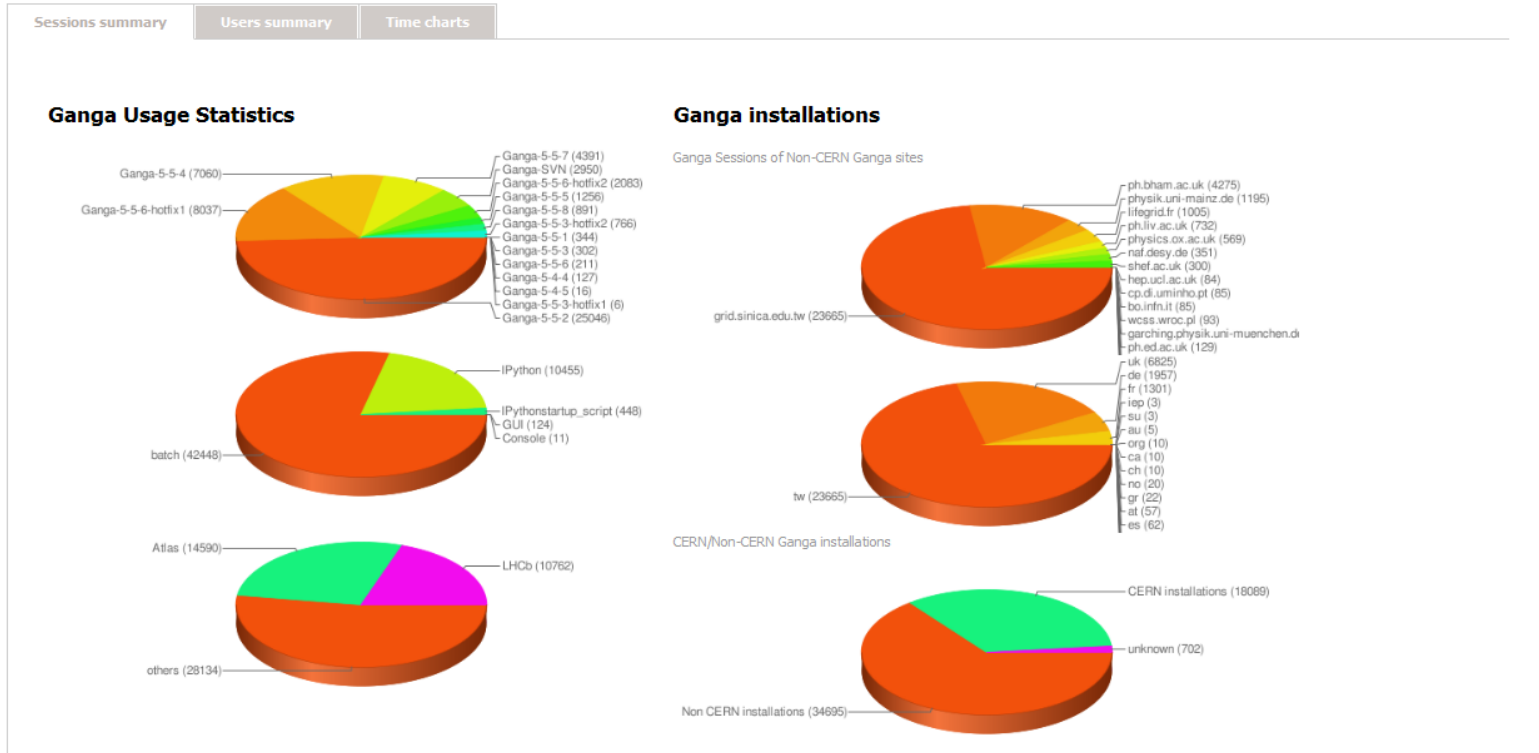
- Ganga may be used as a fronted to DIRAC and other Grid and batch technologies, including Gridway to OpenNebula through GangaKISTI plugin
- Ganga allows Belle users to be shielded from any particular technology and achieve interoperability at the application level between many different systems if needed (e.g. seamless integration of local and Grid resources from a user perspective)
- It is heavily used, production-quality tool



# Ganga Monitoring System

Quick search  Advanced search  
 From:    To:    Experiment:

**From: 2010-05-27**      **To: 2010-06-17**      **Experiment: All**  
**Total number sessions: 53486**      **Number unique users: 507**      **Number of sites: 60**



# Need more information?

- <http://cern.ch/ganga/>
  - User manual, FAQ
- [hn-atlas-dist-analysis-help@cern.ch](mailto:hn-atlas-dist-analysis-help@cern.ch)
- <https://twiki.cern.ch/twiki/bin/view/Atlas/DistributedAnalysisUsingGanga>
- <http://ganga.web.cern.ch/ganga/presentations/index.php>
  - Additional presentations material

