



# Farm Batch System (FBSNG)

J.Fromm, K. Genser, T. Levshina, I. Mandrichenko  
Computing Division, Fermilab



# What is FBSNG ?

FBSNG is a batch system designed as a resource management tool for farm architecture.

Major concepts of FBSNG are:

- **Job/Section/Processes**
  - A job consists of sections
  - Each section is an array of identical processes
  - Section processes start simultaneously on one or more nodes
  - The section dependencies define the sequence of section execution
- **Abstract Resources**
  - An resource is characterized by its name and integer capacity
  - A global resource - available on the entire farm
  - A local resource - available on particular node
  - An attribute - local resource with unlimited capacity
  - Section and process specify required resources
  - Resources are allocated at the beginning of section/process execution and released at the end
- **Load Balancing**
  - FBSNG does resource counting not load measuring



# FBSNG main features

- Job Control:
  - Submission
  - Monitoring
  - Killing/canceling
  - Holding/releasing
- Customizable Scheduler
  - fair-share
  - guaranteed scheduling
- API (Python binding)
  - Allows to develop custom tools to manage FBSNG jobs.



# Who is using FBSNG?

- Fermilab users
  - Experiment dedicated Farms
    - CDF Farm – 154x2CPU nodes
    - D0 Farm – 90x2CPU nodes
    - CMS Farm – 40x2CPU nodes
  - “Common Use” Farm
    - Fixed Target Farm (BTeV, KTeV, NuMI, SDSS...) – 90x2CPU nodes
- NIKHEF (D0 collaborators)
- Northwestern University (D0 & CDF collaborators)
- Other farms at HEP as well as non-HEP organizations

# Current Status



FBSNG version v1\_3 has been released on June 1, 2001

- Current version is v1\_3d. This is a production version at Fermilab
- FBSNG and all required products are available from fermitools (<http://www.fnal.gov/fermitools>)

# Requirements



- FBSNG is written primarily in Python
  - There are a few C /C++ sources
- FBSNG is currently supported under UNIX (IRIX and OSF1 platforms) and Linux operating systems
- Software/hardware requirements:
  - Python v1.5 – v2.1
  - FCSLIB v2\_0a or higher
  - GUI requires python built with Tcl/Tk support and Tcl/Tk
  - Configuration files synchronized on all farms nodes (NFS could be used)



# New Features (I)

- Kerberos authentication support (optional)
  - configurable usage of kerberos v5
  - creation of kerberos credentials for batch processes
  - support of "proxy list" concept
- Dynamic farm re-configuration
  - creation/modification/deletion of queues, process types, resources, etc... on a fly
- Customizable processes placement schemas
  - round-robin
  - maximizing number of processes per node
  - one process per node

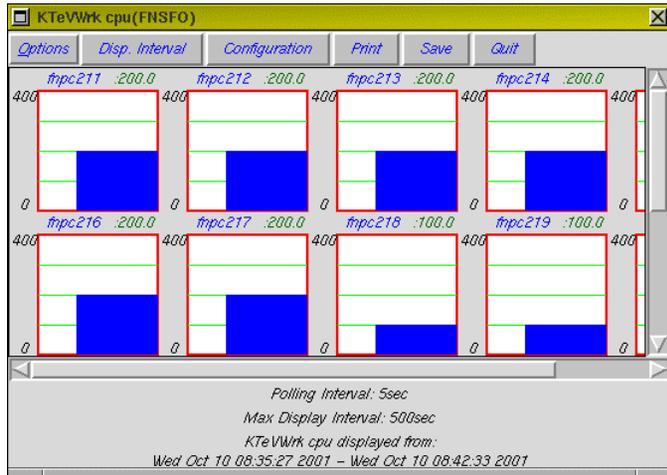


## New Features (II)

- Ability to run batch processes in interactive mode
- Direct targeting of farm nodes
  - user can specified (in Job Description File) a set of nodes for section to run on
- Object ownership
  - Customizable list to authorized user that can access certain objects (such as queue, process type) and perform administrative action
- Asynchronous job status changes notification of API client.
  - FBSNG API client can subscribed to receive notifications about batch job section or process status changes



# GUI New Features

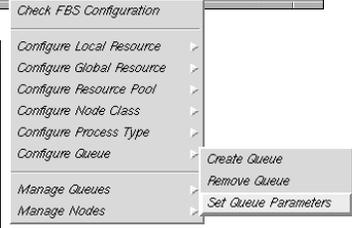


GUI monitor provides functionality to monitor abstract resource utilization



The Set Queue Parameters(FNSFO) dialog box contains the following sections:

- Available Queues:** A tree view showing categories like KTeV\_Short, NuMI, CMS, SDSS, and TEST, with sub-items like KTeV\_Long, BTeV, NuTeV, E998, E781, and IOQ\_H. The 'KTeV\_Medium' queue is selected.
- Configuration Fields:** A list of parameters with values:
  - default process type(string): KTeV\_Medium
  - authorized users(-user name>,... or <\*>): /ktevdb
  - queue priority gap(integer): 200
  - queue priority increment(integer): 0
  - queue priority decrement(integer): 20
  - maximal queue priority(integer): 10000
  - minimal queue priority(integer): 0
  - section priority gap(integer): 1000
  - maximal section priority(integer): 100
  - priority(integer): 0
  - real time limit(integer): 57600
  - cpu time limit(integer): -1
- Available Process Types:** A list including BTeV\_Gen, BTeV\_Worker, CMS\_Worker, E781\_Worker, E871\_Worker, E898\_Worker, IO, IO\_H, and IO\_0.
- Acceptable Process Types:** A list containing KTeV\_Medium.



Gui monitor allows dynamic farm reconfiguration

# Web interface (FBSWWW)



### FBSNG on the web

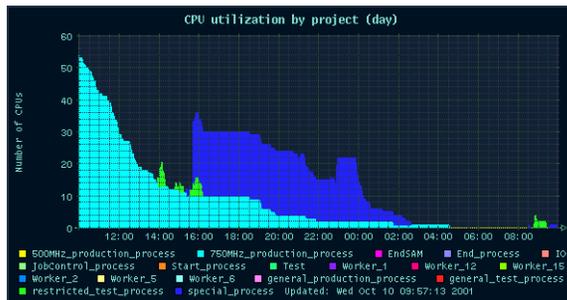
Farm: fnsfo  
Time: Wed Oct 10 20:34:50 2001  
Report: List of queues

[Queues](#) [Jobs](#) [Nodes](#) [Process Types](#) [Graphs](#)

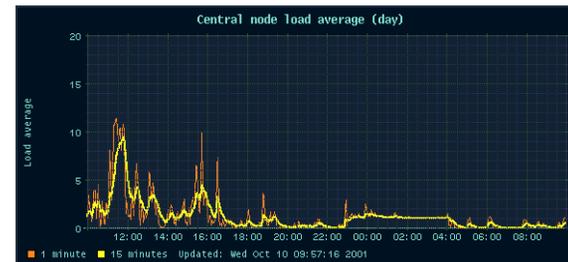
Other farms:  
[DD](#)  
[CDE](#)  
[fnsfo](#)

Name	Status	Default Process Type	Share	Prio	Waiting	Ready	Running	Total
BTeV	OK	BTeV_Worker	3.00	0	0	0	0	0
CMS	OK	CMS_Worker	1.50	0	0	0	0	0
E781	OK	E781_Worker	3.00	0	0	0	0	0
E871	OK	E871_Worker	3.00	0	0	0	0	0
E898	OK	E898_Worker	1.50	0	0	0	0	0
IOO_H	OK	IO_H	(inf)	500	0	0	0	0
IOO_O	OK	IO_O	(inf)	500	0	0	0	0
KTeV_Long	OK	KTeV_Long	1.00	0	0	0	0	0
KTeV_Medium	OK	KTeV_Medium	1.50	0	0	0	0	0
KTeV_Short	OK	KTeV_Short	3.00	0	0	0	1	1
NuMi	OK	WorkerNuMi	3.00	0	0	0	6	6
NuTeV	OK	WorkerNuTeV	3.00	0	0	0	0	0
SDSS	OK	SDSS_Worker	1.50	0	0	53	32	85
TEST	OK	Worker	(inf)	100	0	0	0	0

Web interface is build on FBSNG API. It is using [RRD tool](#) (developed by Tobi Oetiker). It is not included into FBSNG distribution.



CPU utilization per project



Central node load average

# Other Farm Tools



- Farm Remote Copy Utility (FCP)
  - allows to limit the number of concurrent data transfers
  - performs transfer only when disk space is available to complete it
- Farmi Inter-Process Communication toolkit (FIPC)
  - provides means to establish synchronization and communication between processes running in a distributed batch environment.
- Distributed farm disk storage (Disk Farm)
  - manages distributed disk space on the farm nodes
  - provides “global virtual file name space”
  - utilizes Unix-like command for file manipulation
  - allows data replication

# Summary



- FBSNG was used quite intensively during the last several years at Fermilab
- It has proven itself to be a production quality resource management tool for farm architecture
- FBSNG is portable and easily configurable
- It provides API, GUI and Web Interface
- FBSNG is supplied with comprehensive documentation (<http://www-isd.fnal.gov/fbsng>)
- Several additional packages have been developed that allows for inter-process synchronization, balancing network load, and disk space utilization