

**McFarm: first attempt to build a practical, large
scale distributed HEP computing cluster using
Globus technology**

Anand Balasubramanian

Karthik Gopalratnam

Amruth Dattatreya

Mark Sosebee

Drew Meyer

Tomasz Wlodek

Jae Yu

University of the Great State of Texas



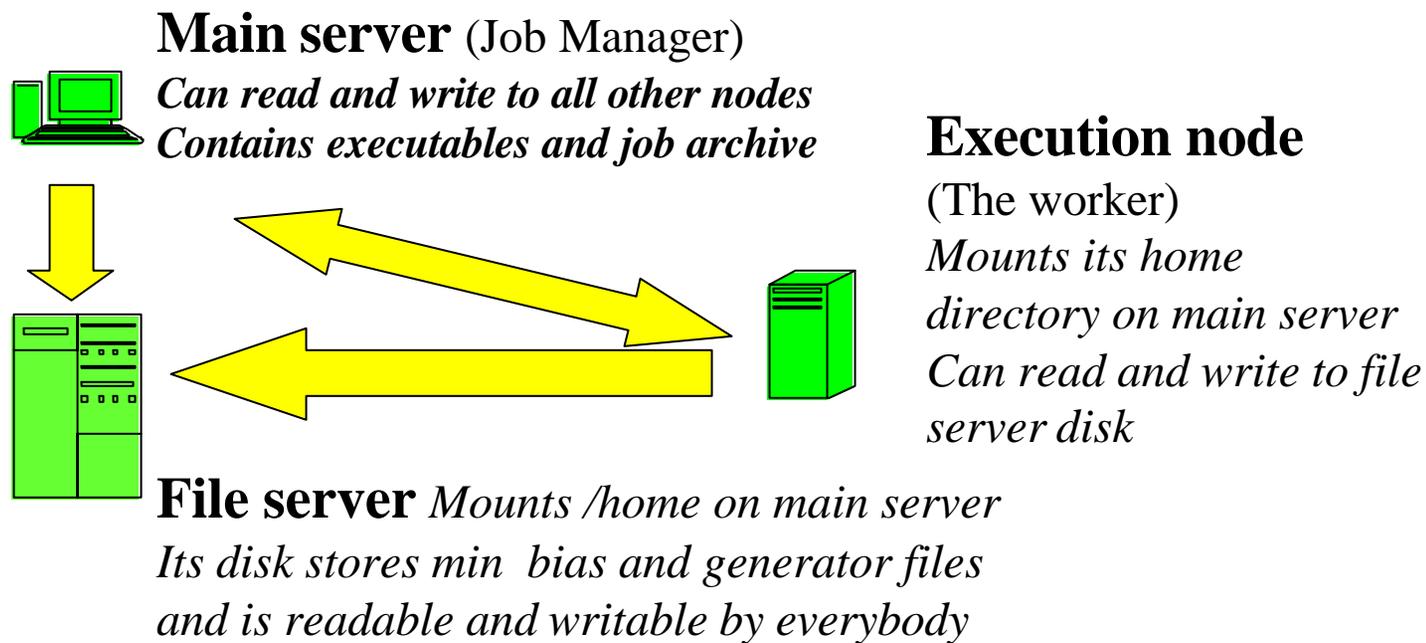
UTA D0 Monte Carlo Farms

- UTA operates 2 independent Linux MC farms: HEP and CSE

HEP farm: 25 dual pentium machines

CSE farm: 10 single 866MHz Pentium machines

- Control software (job submission, load balancing, archiving, bookkeeping, job execution control etc) developed entirely in UTA
- We plan to add several new farms from various institutions



Both CSE and HEP farm share the same layout, they differ only by the number of nodes involved and by the software which exports completed jobs to final destination

The layout is flexible enough to allow for farm expansion when new nodes are available

Several experimental groups in D0 have expressed interest in our software and plan to install it on their farms

- *LTU*
- *Boston*
- *Dubna*
- *Oklahoma*
- *Brazil*
- *Tata Institute, India*
- *Manchester*

The Oklahoma farm is already running, LTU has installed McFarm but is not yet producing data.

Tata institute is installing McFarm. We hope that others will follow.

How can you install the McFarm software?

- WWW page <http://www-hep.uta.edu/~d0race/McFarm/McFarm.html>
- You will find there a collection of notes and scripts for installation of farm server, file server, worker nodes, gather servers etc.
- Also you will find there additional information: how to install Linux, Globus, Sam, etc.
- **Software is available for download, but read documentation first!**

How do we intend to control MC production?

- When the number of farms is small (<2) it is possible to logon to each one of them and submit/kill jobs
- But when one has to manage a large number of farms (= 2 or more) this becomes impractical.
- We need a central site which will control the production at distributed locations.
- Here is where GLOBUS comes into picture...

Future of job submission and bookkeeping



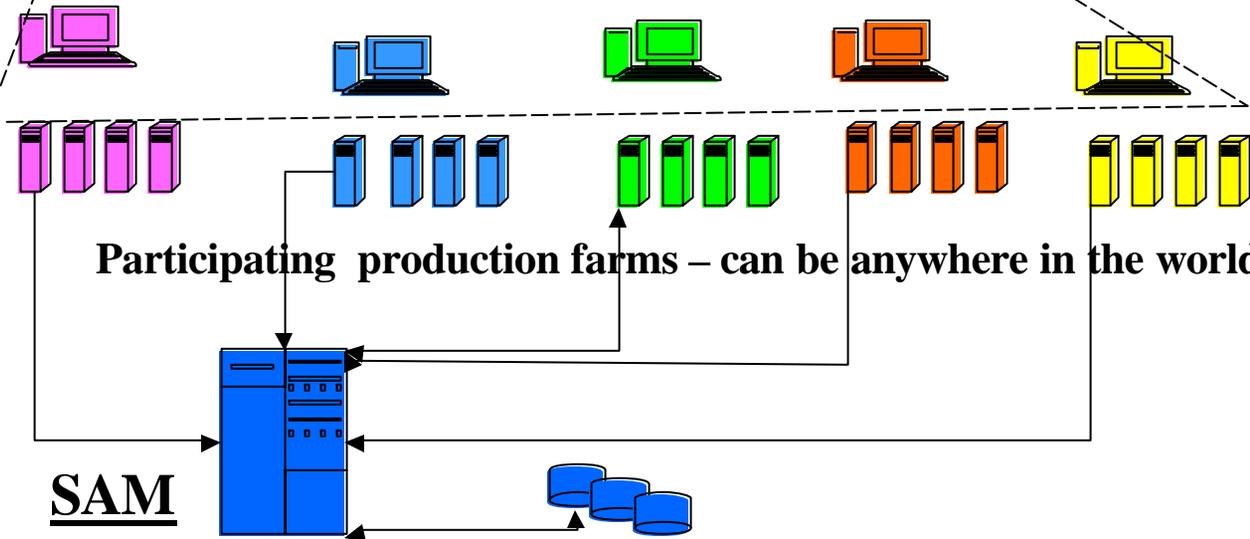
user

MC production server

*Only one machine takes care
of the job submission
and monitoring for all farms!*

**www server
(production
status)**

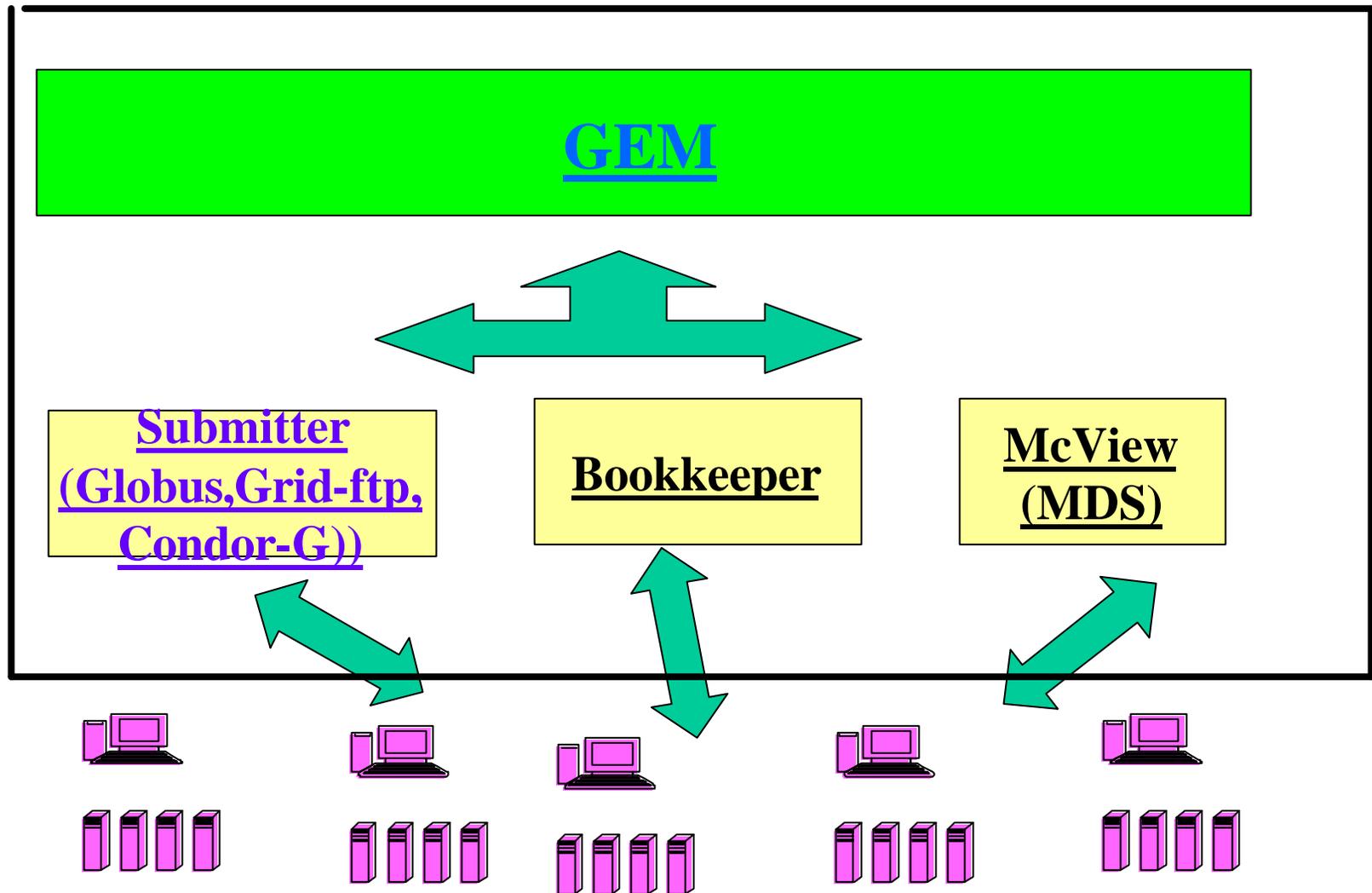
Job submission and control via Globus-tools



What do we need to run McFarm at a dozen of farms?

- We need three things:
- The ability to submit (and delete) jobs to remote farm
- The ability to monitor the production progress (The *bookkeeper*)
- The ability of obtain fast information about number of jobs waiting, running, errored, done on a given farm and the number of active nodes on that farm (the *intelligence provider, or McView*)

The plan

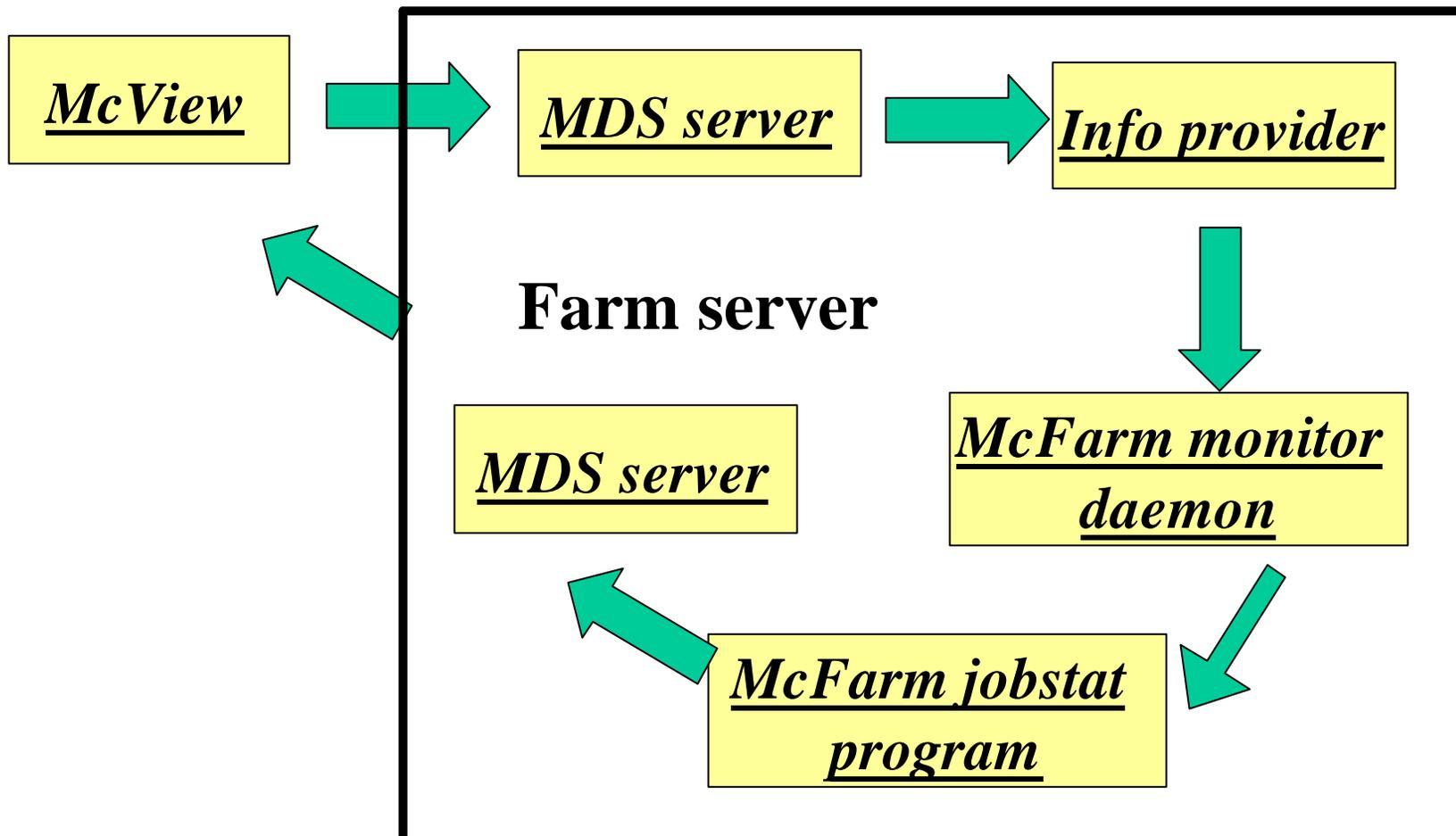


Future of bookkeeper

- **At present it stores run information in plain text files in a private format**
- **In the long run this is not a way to go**
- **I intend to split bookkeeper into two packages: one collects the run information and stores in MySQL, second does the WWW publishing**
- **I will design a good way of storing run information in MySQL database.**

McView data flow

Control station



GEM – the future of future

- **Globus Enabled McFarm**
- **The three components: submitter, bookkeeper and McView so far are supposed to be used in semi-automatic way by operator.**
- **Operator does the scheduling, based on what he sees in McView table and submits jobs to given farm**
- **GEM is going to connect those three packages and will do everything automatic. User will have to submit a run, and the rest will be left to GEM.**

Life is hard: Known Problems:

- **Sites tend to have various versions of executables, a bookkeeping of “which site has which version” is needed. (A potential nightmare)**
- **Possible solution (?): send executables with job**

Related problems:

- **Not all minitars work properly**
- **We had hard time with installing most recent ones in OU**
- **We will need (sooner or later) centralized system for automatic distribution and installation of minitars**

Another related problem:

- **Various places might have different kernels**
- **This did not hit us yet, but it will...**

Problems:

- **Various farms have different environment initialization (For example: to run “setup” command you have to source different script at different sites!)**
- **Solution: we lump the setup commands into “setup_farm” script which hides the problems**
- **Not an ideal solution!**

Problems:

- **Clocks are not synchronous at different farms!**
- **Result: Proxy identification can fail!**
- **Solution: use common time servers!**

Problems:

- **MDS information providers need root pwd to be installed. Possible maintenance nightmare!**
- **Solution: Anybody knows?**
- **Related: it would be good if the “Superserver” farm had the right to restart Globus daemons on “slave farms”. (I doubt farm administrators will like this idea...)**

Problems:

- **As mc_runjob evolves it needs constantly be patched.**
- **This (patching) should be made more automatic!**

Conclusion:

- **We do have a cluster of 3 McFarm sites (2 in UTA and 1 in OU) running and controlled via Globus**
- **First step towards Grid has been made!**
- **We expect to have LTU ready soon (few days, maybe even hours...), Tata should follow soon after**
- **More institutions are invited to join...**