CRAB: status and perspective

Stefano Lacaprara

INFN Legnaro National Laboratory

CMS Italia, 13 July 2006

Current status

- CRAB CMS Remote Analysis Builder
  user oriented tool to grid submission and handling of analysis jobs

Goal of the project

provide an user friendly front-end for end user interaction with the grid for CMS, including interaction with data management, middleware, remote computing element, basic monitoring functionalities, etc.

Status

In production. Used since more than a year by end users.
Only computinig tool in CMS used by physicist.

Current status II

- **Support DBS/DLS Done** Available both for old (ORCA) and new EDM (CMSSW). Fully implemented and available to users (not yet as default - see after). Side remark: it takes a lot of time to switch between two implementations of a component from operation point of view.

- **Support for CMSSW Done** Complete workflow works, using DBS/DLS, site local config, trivial catalog etc . . . (to be released soon - see after)

- **Integration with BOSS** CRAB makes full use of BOSS (version 3.6.4). Integration with BOSS4 is basically completed.

- **gLite3.0** Standard job submission tested and working. Bulk submission tested by BOSS team, still some problem with 3.0, works well for 3.0+fix (to become 3.1, when?) Should therefore be available out-of-the-box for CRAB.

Current status III

- **Inter-operability with LCG - OSG** Fully supported submission, via Resource Brocker to OSG sites (as well as LCG ones, of course). Transparent to final user, needed some work on CRAB side. Cannot be 100%, due to lack of some services in OSG (mainly data management-related, *e.g.* file catalog as LFC)

- **Dashboard** Full set of information about task, job, application, status, etc . . . sent to Dashboard, both from User Interface and from Worker Node.

- **Multiple job type (FAMOS)** Supported: meant mainly as an exercise to understand how difficult is to introduce a different job type wrt ORCA. Very useful for CMSSW integration.

- **Pythia job with CMSSW** New in CRAB_1_2_0: working and already used to produce events with success (see after)

CRAB
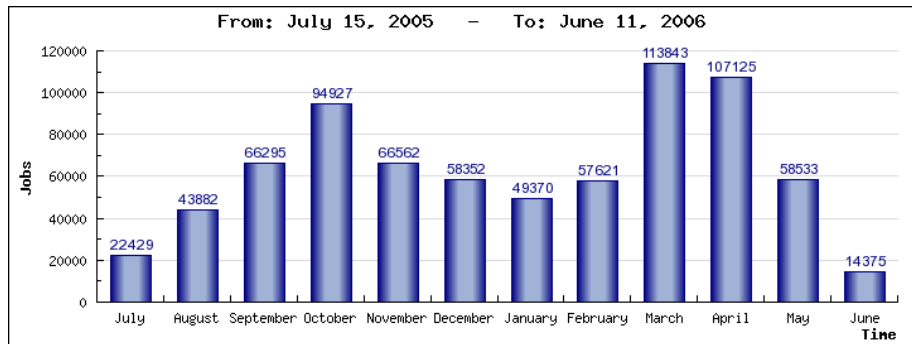
reached 100'000 jobs/month: daily record is about 10'000 jobs.

Current problems

- It took more than 3 weeks to issue the 1_2_0 release, and it is not out yet...(it has a lot of new stuff, true)
- The core developers have been 100% busy with test and fixes: limited help from other people
- core developers have limited time left for *real development*
- development is slowed down by lack of dedicated resources (eg test-bed with exclusive or privileged access)
- In principle VOMS + work on T-2 (or similar) can provide this. It's just a matter to do it: who? not the core developers...
- Where (and how) to put user output? Copy to Storage Element is working but file copy is not reliable!

Current problems II

- Limited experience with CMSSW usage: again PRS feedback would be valuable
- A good example is pythia use case, asked by two users and implemented very quickly.
- Now: limited usage by user, so user-support not a burden, but was in the past, and will be in future. Again, it mostly relies on core developers (not only, luckily, experienced user helps more and more)
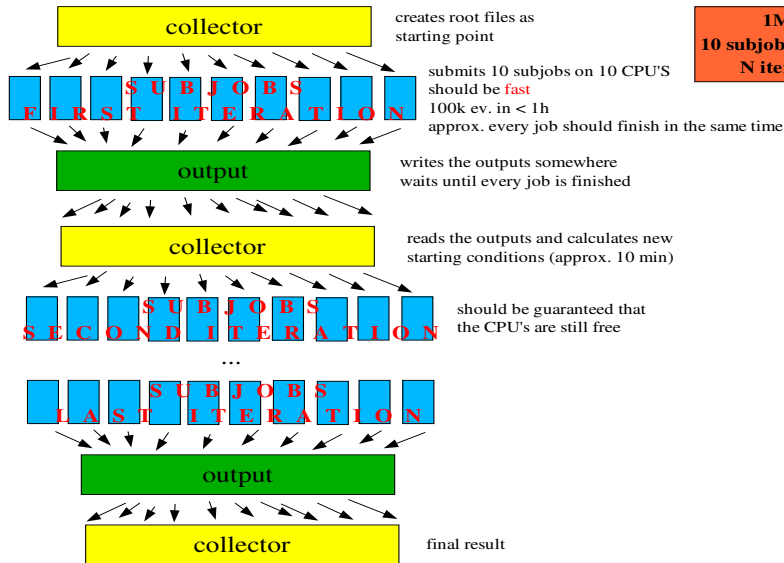- Many, many time a real user-support has been asked, but so far...

Future: Use cases

- So far CRAB worked more or less well for current (PTDR use cases)
  - difficult to bring home all jobs: an automatic (and smart) monitoring and resubmission mechanism could be very helpful
  - scalability.
  - more complex workflow: good example is alignment
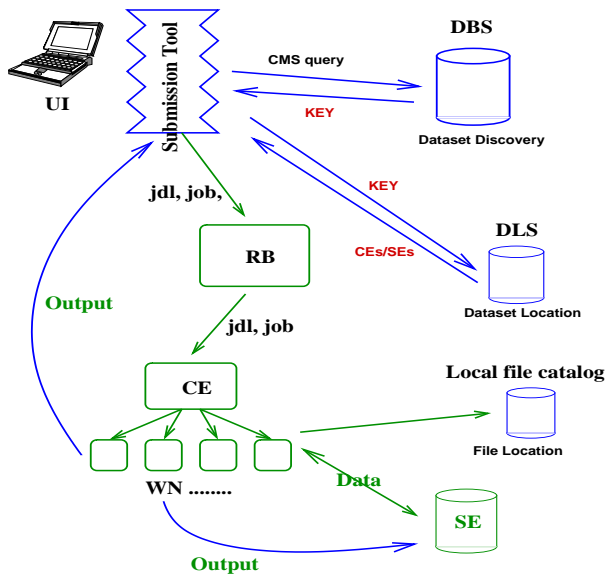 - make available to a group what have been produced by someone (eg pythia) publication in *private* DBS/DLS

Alignment use case (by Laura Edera)

collector — creates root files as starting point

S U B J O B S
F I R S T I T E R A T I O N — submits 10 subjobs on 10 CPU'S
should be fast
100k ev. in < 1h
approx. every job should finish in the same time

output — writes the outputs somewhere
waits until every job is finished

collector — reads the outputs and calculates new
starting conditions (approx. 10 min)

S U B J O B S
S E C O N D I T E R A T I O N — should be guaranteed that
the CPU's are still free

...

S U B J O B S
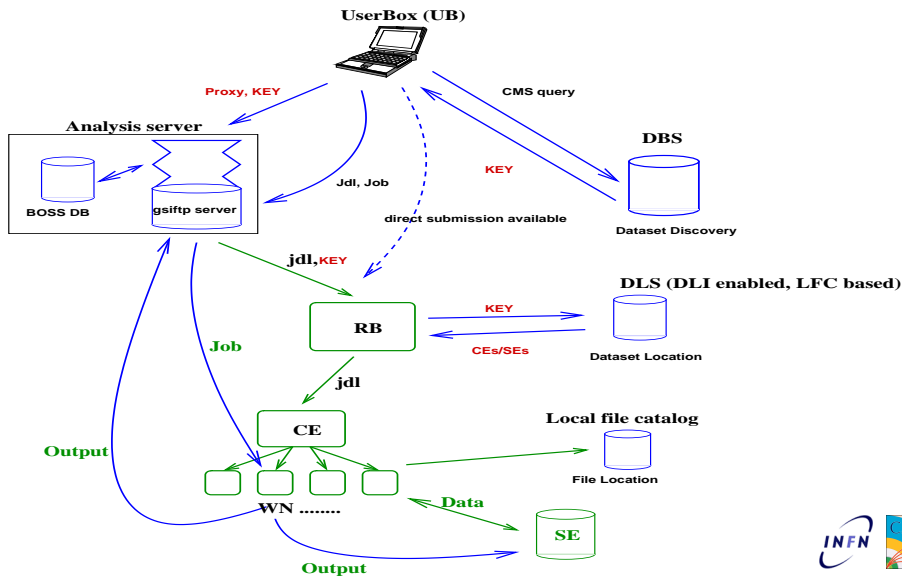L A S T I T E R A T I O N

output

collector — final result

1M ev.
10 subjobs (100k ev.)
N iterations

A Proposal for CRAB Analysis Assistant (CRAAAB)

A Proposal for CRAB Analysis Assistant (CRAAAB)

Workplan

- Goal of the exercise:
  - ▶ Start playing with existing tool: how hard is to put something in place?
  - ▶ Once a toy is there, we can start to evolve the AA to its real goal, which is automatize task handling (check status, resubmission, black-list of black-hole, etc...)
- Start with existing tools: CRAB + JobRobot
- CRAB: split creation and submission
- JobRobot: for automatic submission and retrieval
- First exercise done with limited effort
- Gaining real experience with JobRobot: is a good starting point? Is it already usable as is? What is missing?
- A scalable architecture for the Assistant is still under discussion

Problems

- Technical: many!
- keep the server as simple as possible, start with something working and iterate: Beware!! The tool MUST be user-oriented
- handling of user output will be a serius thing (will be in any case, with or w/o AA)
- Question: where do we place the AA? How many of them?
  - Number depends on scalability. Must allow any user to setup one on a machine anywhere easily.
  - Natural place is probably CMS T2's: require a commitment at T2 to support *directly* analysis activity with something which is CMS dedicated and not a "general" LCG middleware.
- Socio-political: is it an antagonist of ProductionAgent?
- IMHO not really: it's a *natural* upgrade of CRAB. Unless PA is defined to be *the only way* to do *production-quality* activity.
- Provided people is able and focused to work on it, we can have a prototype very soon
- On the other hand, CRAB as is today is not able to cope with *use cases* cited above.

Publication of results

Use case very simple and often presented:

User A produced a huge dataset of some kind, and he want to share it with its co-workers

- In principle is foreseen by CMS model:
- Have a "private" (local-scope, local can be as "global" as one wish) DBS and DLS
- publish private data on private DBS/DLS
- send DBS/DLS URL to co-workers
- that's it.
- Note: the very same mechanism might apply also for MTCC data, or TB data, or "private" production
- It is a very interesting integration test: are we interested in pursuing it?

Summary

- CRAB is up and alive
- More interaction with PRS will add a significant plus to the future of the project
- A proposal for CRAB evolution with a dedicated server is under study
- A complete exercise of publication and access of data produced in MTCC is possible and can be done: require agreement and focused effort

Concluding remarks

- Where is the humble physicist in CMS computing model?
- Where does he work? CERN? My PC?
- Where does he put his results? Castor? His Storage Element (which is "my" SE?) ? My PC?