CRAB@CAF.cern.ch : status

Stefano Lacaprara

INFN LNL

Facilities Operation meeting
CERN, 1 February 2008

Intro

- Goal: to offer the CRAB user interface and functionalities to submit jobs to CAF (and LSF in general)
  - Status: beta ready
  - Still pre release CRAB_2_1_0_pre3
  - available via

    ```
    source
    /afs/cern.ch/cms/ccs/wm/scripts/Crab/CRAB_2_1_0_pre3/crab.sh
    | csh
    ```
  - Standard CRAB, with two additinal scheduler LSF and CAF
  - Waiting for beta tester and feedbak
  - expect a rapid development cycle: stay tuned!

What is available

- As for CRAB on grid, user should provide analysis code (any), declare what data he want to analyze, and CRAB does the rest
  - Job preparation, submission, monitoring, output handling
- For LSF submission you need to run from a node with LSF available.
- Don't need to be on a grid User Interface (does not harm, though), not to have a valid `proxy`
- Support for standard CMSSW job (any input data, including None)
- Run your job interactively to test the code, then fill `crab.cfg` to tell CRAB what to do
- Select CAF (or LSF) as scheduler
- create and submit the jobs

Configuration

- New LSF specific section in `crab.cfg`, possible to define the `queue`, `resources` other things.
- CAF define automatically the right queue and resources as well as other stuff in the wrapper before job execution
  - eg as castor dedicated stager
  - `export STAGE_SVCCLASS=cmscaf`
  - any additional stuff cen be put on request or we might allow user to define a `pre-execution` step to be performed (if needed)
- Data discovery is done by CRAB using DBS: just put the datasetpath into `crab.cfg`
- Of course, only dataset (data blocks) located at local site (`cern.ch`) will be selected
- Job splitting is done by CRAB according to user preferences as set on `crab.cfg` and on data availability

- User output has to be declared in `crab.cfg`.
- Two options for handling:
  - `return_data = 1`
  - the output (along with stdout/err and FJR) is returned on user working directory under `<crab_Working_Dir>/res` after `-get`
  - `copy_data = 1`
  - the output is copied at the end of the jobs to `castor` via `rfcp` to `$CASTOR_AREA` or where user decided
  - `storage_path = /castor/cern.ch/user/s/slacapra`
  - Stdout/err and FJR are returned (after `-get`) to `res/` directory
- Should we consider also copy to a remote storage element as well? (would need a `voms-proxy`)

- Checking job status `crab -status` gives you the status of all jobs as reported by LSF
- Other possible to kill, resubmit etc . . . via usual command `-kill <range>, -resubmit <range>`
- Monitoring is done autoamtically via DashBoard: still under test

crab.cfg example

```
[CRAB]
jobtype = cmssw
#scheduler = lsf
scheduler = caf
[CMSSW]
datasetpath=/GlobalNov07-A/Online-CMSSW_1_7_1/RAW
pset=DTAnalyzerGlobalRun_Nov07_Frontier.cfg
total_number_of_events=110
events_per_job = 10
#number_of_jobs = 1
output_file = DTAnalyzer.root, DTEffAnalyzer.root, STAnalyzer.ro
[USER]
#return_data = 1
copy_data = 1
storage_path = /castor/cern.ch/user/s/slacapra
[LSF]
#queue=1nh
#resource=cmscaf
```

Future

- Basic functionalities are there, need user feedback to add more (and fix problems)
- Now only basic workflow is implmented
  - we want to add a finale stage of processing to be done when all jobs finished
  - useful for merging or for collecting the output or ...
  - in principle can be used for anything, including user-defined step (might be interesting for alignement use case)

Future II

- Use of a CrabServer for LSF/CAF? Probably not needed.
- LSF rather reliable, errors are likely to be on user code or on data, not on infrastracture
- Can be useful if it can trigger more complex workflow (as merge-like jobs)
- Anyway, no real show stopper, just a matter of deciding
- Non cern sites have expressed interest in porting CRAB@lsf.cern.ch also elsewhere (lsf at CERN is different), and on other batch system (PBS)

Conclusion

- Beta version for CRAB@CAF.cern.ch is available
- Need beta tester to find bugs and problems
- Need user feedback to guide further development
- If you are interester, please contact CRAB feedback list ¡hn-cms-crabFeedback@cern.ch¿ or directly me <stefano.lacaprara@pd.infn.it>