

Thoughts on CRAB3- HTCondor Integration

Brian Bockelman

Talking about things we've done and
problems where we need advice.

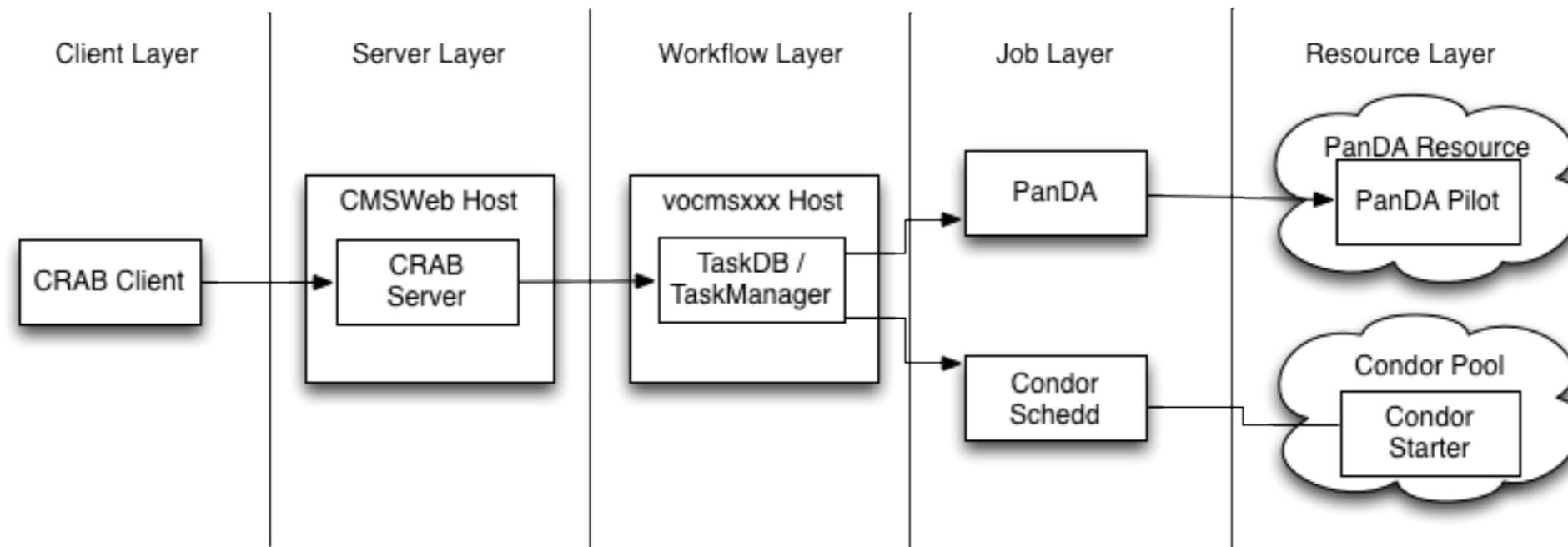
If you're looking for a finished product,
look elsewhere!

Big Picture

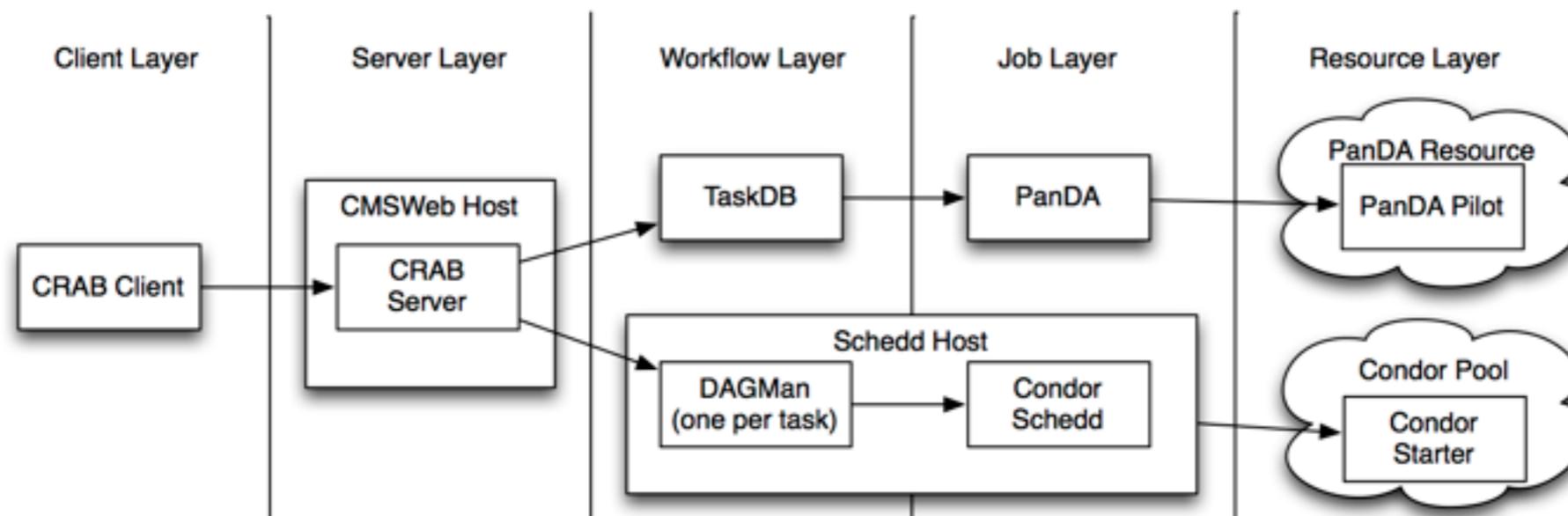
- We've been exploring two alternate designs in parallel:
 - TaskWorker submits DAGs to HTCondor.
 - CRABClient submits TaskWorker process to HTCondor. TaskWorker submits DAGs to HTCondor.
 - Note the former is a prerequisite for the latter.
- I'm going to focus on the CRAB3 / batch system interaction.

For visual learners

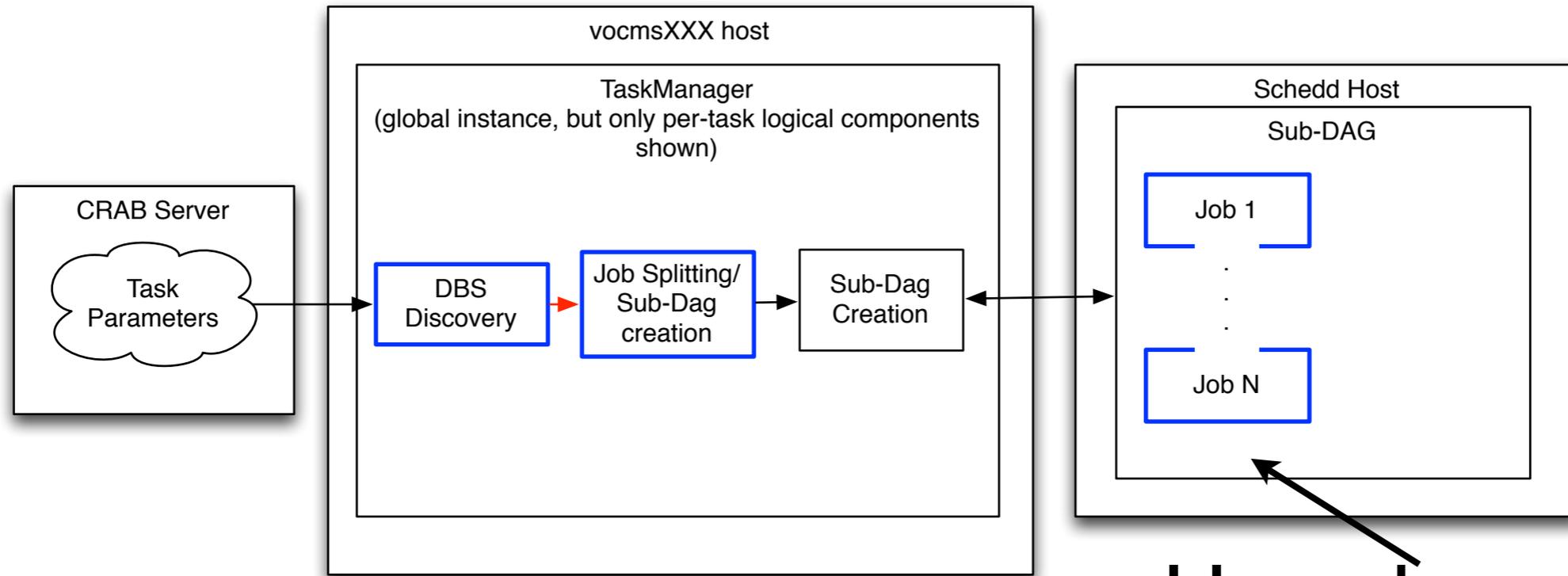
Option A



Option B - with CRABServer

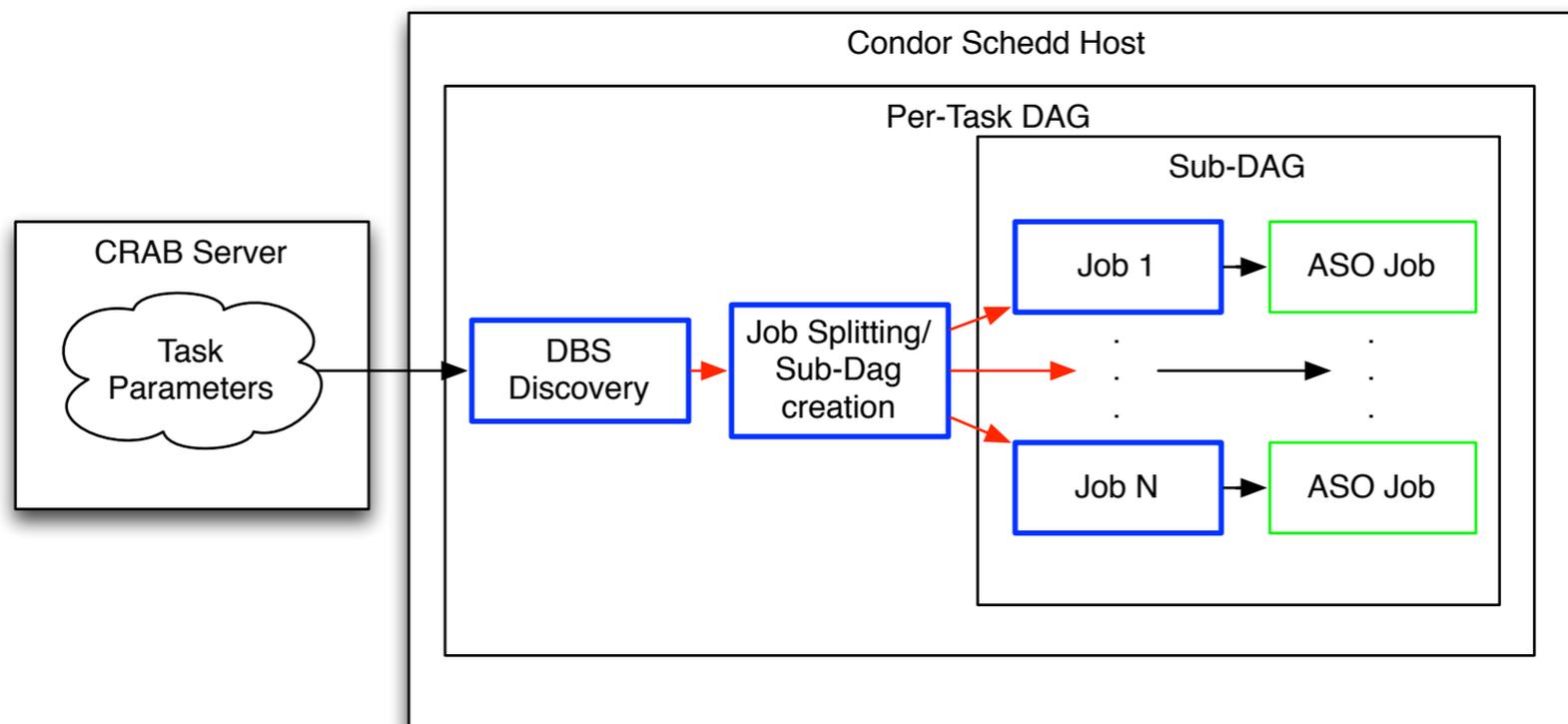


One Layer Down



VS

How does ASO fit in?



Why DAGs?

- DAGs - Directed Acyclic Graph of jobs. A set of jobs that have parent/child relationships. In HTCondor, managed by an external process, DAGman.
- For those involved with CRAB2, you may be surprised to see DAGs - why not straight HTCondor jobs?
- DAGs allow us to delegate state machine logic to an existing program. We can treat the jobs as a logical *task*, analogous to PanDA's *jobset*.
- DAGman will throttle the number of jobs in the queue; not submitting more jobs for a task if there are already *X* idle. This allows us to keep the number of jobs in the schedd approximately the same

Status

- Developed an initial version in April.
 - This version had various stubs and scaffolding for components we didn't understand or couldn't get working.
 - Since then, we've been working to replace the stubs with the appropriate pieces from the CRAB3 code.
 - In some cases, waiting for refactors of CRAB3 code so the HTCondor integration has a more natural place to land (see ticket CAF-17).
- The code should start landing upstream in a week or two. Will take a bit longer for all code to land. Need to re-merge against the upstream changes from the last month.
- Devs have performed a few end-to-end test tasks.
 - Does all the things HTCondor does - works with glexec.
 - Does not do the things HTCondor does not do - no PandaMon.
- Right now, we use the native HTCondor python bindings - working on remote gsissh submission so we can use the existing AnaOps pool.

Things that keep me up at night

- Many CRAB3 devs have been working together at CERN - or having been working on the project - for awhile.
 - The process of onboarding a new, remote person is a bit choppy.
 - Special thanks to Mattia and Daniele for helping me out.
 - I think we've hit our groove as long as everyone works in JIRA.
 - Starting to get the big picture of where our work fits in and *how it gets in* to the source code.
- **Unit tests, unit test, unit tests:** CRAB3 has several components, but there's currently no way to work on one component without having several working. Unit tests would allow us to better modularize and only work on one module at a time.
- **Documentation:** The design document tends to be "Daniele's last presentation". We need something that goes deeper - a TDR would be great!
 - What assumptions are made about runtime environments? What APIs exist?
 - A lot of time our slides are simplifications - the communication between modules may be represented by a single arrow.
 - A map between modules and python files would be great.
 - It would greatly help the understanding of contributors and future maintainers.

Other Late Night thoughts

- Technical challenges / opportunities:
 - We really need help with ASO. Haven't been able to figure out how PanDA communicates with the ASO server.
 - I think there are some PanDA assumptions living in the resubmit code. I've not stared at the code enough to give actionable feedback. Will come in a month or so.
 - Framework job reports are returned to the schedd and retrievable by the CRABServer. Not sure if this model reasonably matches CRAB3-PanDA.

Final Late Night Thoughts

- I would love to see a CRAB3 hammer cloud. The more sites we use, the more we uncover any underlying faulty assumptions.
- For example, we currently assume storage elements have umask 000 - assumes sub-directories of world-writable directories are world-writable. This is a big difference from normal practice!
- CRAB3 commissioning starts in August.
 - That's the start line, not the finish line. We should set expectations appropriately -- the real finish line is the start of Run II.
 - I hope, in between, we can fill out some of the documentation and testing pieces I mentioned.