Introducing the HTCondor-CE

CHEP 2015

Presented by Edgar Fajardo
Introduction

• In summer 2012, OSG performed an internal review of major software components, looking for strategic weaknesses.
  • One highlighted area of concern was the gateway software (GRAM).
  • Decided to diversify; internally evaluated CREAM, HTCondor-CE, and staying with GRAM. Decision was to use HTCondor long-term.

• During 2013, major integration and beta testing was done.

• In 2014, large scale testing and transition started at largest, most complex sites.
What is HTCondor-CE?

• A **Compute Element (CE)** is OSG's entry point to a site's local cpu resources

• The core of a CE is the **gateway**, which provides:
  
  • **Routing**: Translation of a resource request to a batch system job (includes initial job submission, state tracking, and cleanup).

  • **Remote management**: A protocol for managing resource requests from remote clients.

  • Authentication and **Authorization**: Determining identity of the client and what actions they may perform.

• **NOTE**: In OSG, we do not submit jobs to CEs. We submit *resource requests*. At most CEs, these become jobs in the batch system.
What is HTCondor-CE?

- The HTCondor-CE is a gateway implemented using a special configuration of the HTCondor software. Everything is:
  - HTCondor daemons.
  - HTCondor configurations.
  - HTCondor plugins.
- Authentication is done with GSI; authorization is done with LCMAPS.
- Remote submit protocol is HTCondor-C.
- The ‘heart’ of customizing jobs is the JobRouter, a declarative transform language.
  - Interface with local batch system is blahp / Condor-G.
HTCondor-CE in a slide

Submit Host
HTCondor Schedd
Job (grid universe)

HTCondor Case
HTCondor-CE Schedd
CE Job

Job Router Transform
HTCondor Schedd
HTCondor Job (vanilla)

Submit Host
HTCondor Schedd
Job (grid universe)

PTBS Case
HTCondor-CE Schedd
Grid Job
Routed Job (grid uni)

Job Router Transform
blahp-based transform
PBS
PBS Job
Transformation Details

- **Job Router**: Transform requests to jobs (localize at CE)

- **BLAHP**:  
  - Submit jobs to non-HTCondor batch systems (PBS, SGE, SLURM, etc.)
  - blahp is the executable which then calls, for example, qstat / qsub / qdel.
  - blahp has another layer of customization if, for example, you need to tweak qsub arguments. Most useful things can be done via the JobRouter transform.
Job Route Example

```
JOB_ROUTER_ENTRIES = \
[ \
  GridResource = "batch pbs"; \
  TargetUniverse = 9; \
  name = "Local_PBS_cms"; \
  default_queue = "cms"; \
  Requirements = target.x509UserProxyVOName == "cms"; \
] \
[ \
  GridResource = "batch pbs"; \
  TargetUniverse = 9; \
  name = "Local_PBS_other"; \
  default_queue = "other"; \
  Requirements = target.x509UserProxyVOName != "cms"; \
]
```

More details/recipes for the routes:

Hate declarative languages? Script-based callout also available!
Why HTCondor-CE?

STRATEGIC

- **Local expertise** - HTCondor developers are within “shouting distance” from OSG software team.

- **Unify** submission, CE, and opportunistic computing platforms - Condor-G, Condor-CE, and BOSCO - **onto a single software stack.**
  
  - At least one HTCondor piece - the **blahp** - overlaps with CREAM CE.

- **Minimize** OSG effort needed. HTCondor-CE is a *configuration* of HTCondor: all non-configuration bugs are a problem for the HTCondor team.

- **No new external dependency**. We depend on the HTCondor team regardless of the CE; with any other solution, we have an additional external dependency.
Why HTCondor-CE?

TECHNICAL

- Management: CE layer is visible and available for interaction. “condor_*” toolset works with CE.

- Packaging: HTCondor provides many customization hooks - and doesn’t overwrite changes on upgrade!

*Limit in the test is the batch system memory, which maxes out at 16k.

<table>
<thead>
<tr>
<th></th>
<th>HTCondor-CE</th>
<th>GRAM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Best max running jobs</td>
<td>16k*</td>
<td>10k</td>
</tr>
</tbody>
</table>
| Network Port usage   | 2           | 4    | (per running job)
CE layer visible!

Can query the CE itself with condor_ce_q!
Provides visibility to CE status information and errors.
Running Daemons

HTCondor CE (condor-ce.example.com)
condor-ce
- condor_master
- condor_schedd
- condor_job_router
- condor_collector:9619
- condor_shared_port:9620

CONDOR_HOST = \condor-ce.example.com
JOB_ROUTER_SCHEDD2_NAME = \condor-ce.example.com
JOB_ROUTER_SCHEDD2_POOL = \pool.example.com

submitter
- condor_master
- condor_schedd
- condor_shadow
- condor_shadow
- condor_shadow
- condor_shared_port:<###>

CONDOR_HOST = pool.example.com
QUEUE_SUPER_USER_MAY = .*
IMPERSONATE = .*

HTCondor (pool.example.com)
Central Manager
- condor_master
- condor_collector
- condor_negotiator

CONDOR_HOST = pool.example.com
Information Service

• Prior information services in OSG were not useful for provisioning. Other services describe the state and configuration of batch queues. We want a provisioning information service that describes:

  • **Resource types.** “I can allocate you a 8-core resource with 16 GB of RAM.”

    • Note we don’t try to enumerate the number of resources of each type - utilized or available.

  • **How to access resources.** “To get at resource type A, set attribute foo=‘bar’ in request.”

    • Want to include both standard attributes (Cores, RAM, Disk) and VO-custom ones (IO intensity, VO queue name, job type).

  • To meet these requirements, we needed to start from scratch.
HTCondor-CE Collector

- Basic idea: reuse HTCondor components to provide the info service.
- Describe provisioning information in a single HTCondor ClassAd.
- Setup a condor_collector centrally. Use built-in forwarding capabilities from site collector to central one.
- Use standard condor querying tools to interact with central collector.

For more information, see https://indico.fnal.gov/getFile.py/access?contribId=19&sessionId=8&resId=0&materialId=slides&confId=8580
Upcoming Activity

• A few strategic directions:
  
  • Flesh out the blahp support for LSF & SGE when used via HTCondor.  
    • Getting blahp to work with LSF has been an **epic battle**.
  
  • Make (HTCondor-CE) - (HTCondor) = smaller.
    • Goal is always to keep the HTCondor-CE “config-only”.  Still Mostly True.
  
  • Take better advantage of existing HTCondor features; get HTCondor team to implement new ones (**Docker universe**).
    • Continue refinements - especially in terms of ease-of-configuration and ease-of-customization.
HTCondor-CE and Docker Universe

• JobRouter allows you to inject arbitrary attributes into the routed job for HTCondor sites.
  • This allows admins to control which HTCondor features or options are turned on for a given user’s job.
  • At Nebraska, we’ve been very interested in containerization efforts; one observation are chroots are hard to create!
    • Docker provides similar container features but provides tooling for easy-to-create environments.

• Next HTCondor release will allow the site to launch a Docker container.
  • Sites can easily modify HTCondor-CE routing to have pilots launched inside Docker containers.
  • We can then share a single Docker image for worker nodes. “One less thing” for sites to maintain.
HTCondor-CE (Local) Collector

• We’ve always wanted more information about payload jobs.
  • Who’s running? What are they running? Are they using CPU efficiently?

• In the next HTCondor-CE release, the CE will allow pilots to send startd ads (representing the payload jobs). The CE admin can view the payload activity with `condor_status`.

• In the next gWMS release, the pilot will send these ads automatically.
Parting Thoughts

- HTCondor-CE is a fresh approach on gateway technology - focusing solely on **resource provisioning** - built on top of the **foundation of HTCondor**.

  - Reason for switching include both organizational and technical.

  - Tries to re-envision both the concept of a gateway and how information services interact.

- Transition has been ongoing within OSG for the larger sites. In 2015, we will start tackling smaller ones.

- Moving forward, we would like the HTCondor-CE to shrink in size (becoming more and more “just HTCondor”) and continue to benefit from new features in the base software.