HTCondor-CE Info Services

Brian Bockelman

HTCondor/Arc-CE Workshop, February 2016

Give Me Information!

- When the OSG started to look at the HTCondor-CE, we considered calling it the HTCondor-PE to focus on the fact we wanted a *provisioning element*.
    - In our mind, **HTCondor-CE does not run jobs**! Rather, it provisions resources for pilot factories!
- When we started to look into our information systems in 2014, we asked ourselves:
    - **What information does a pilot factory need?**
- and can we eliminate everything else?

Start simple

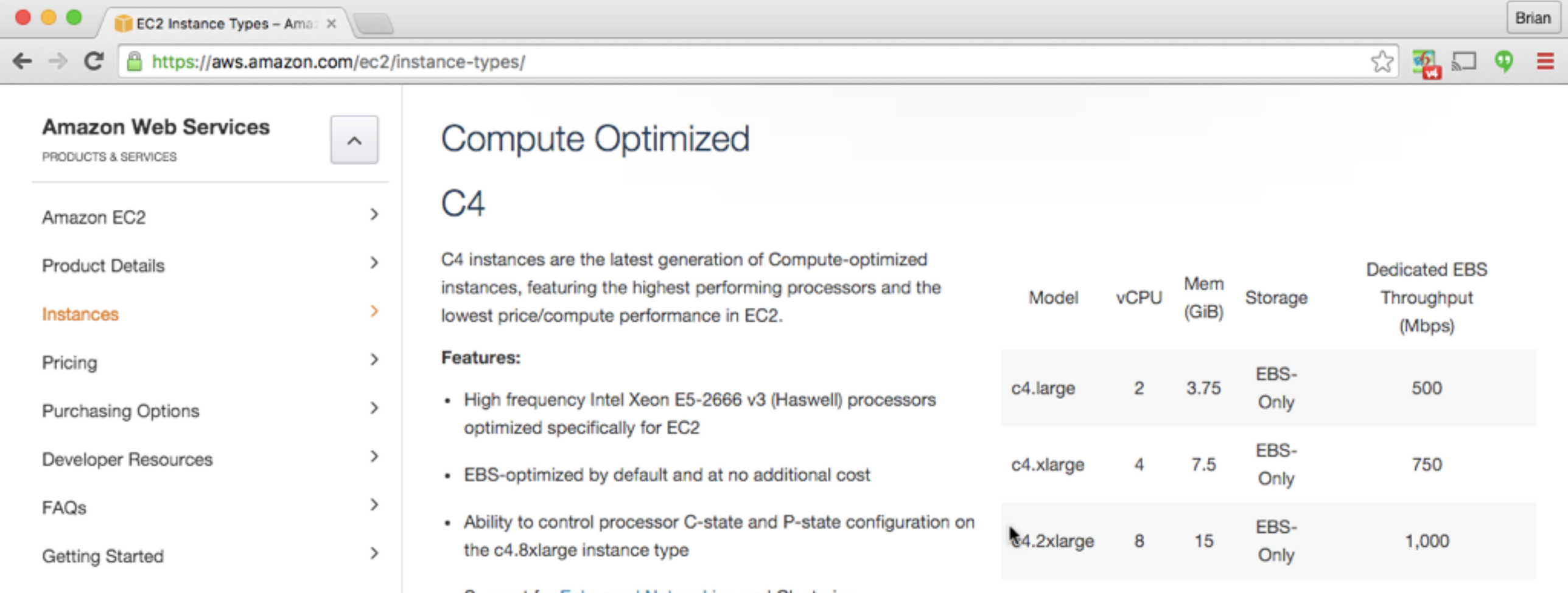
- For each endpoint providing services, we would like:
  - Contact information (**hostname / port**).
  - Access policy (**what VOs are supported?**). Optional - one can always probe for authorization.
  - **What resources** can be accessed? How? At what cost?
  - Debugging information: Site batch system name, site name, versions. Helpful for triage and **human purposes**.

What we **DON’T** want

- Things that shouldn’t get exposed by an endpoint:
  - Internal/private configuration (batch system queue names).
  - Description of site topology.
  - Site size / WLCG pledge info.

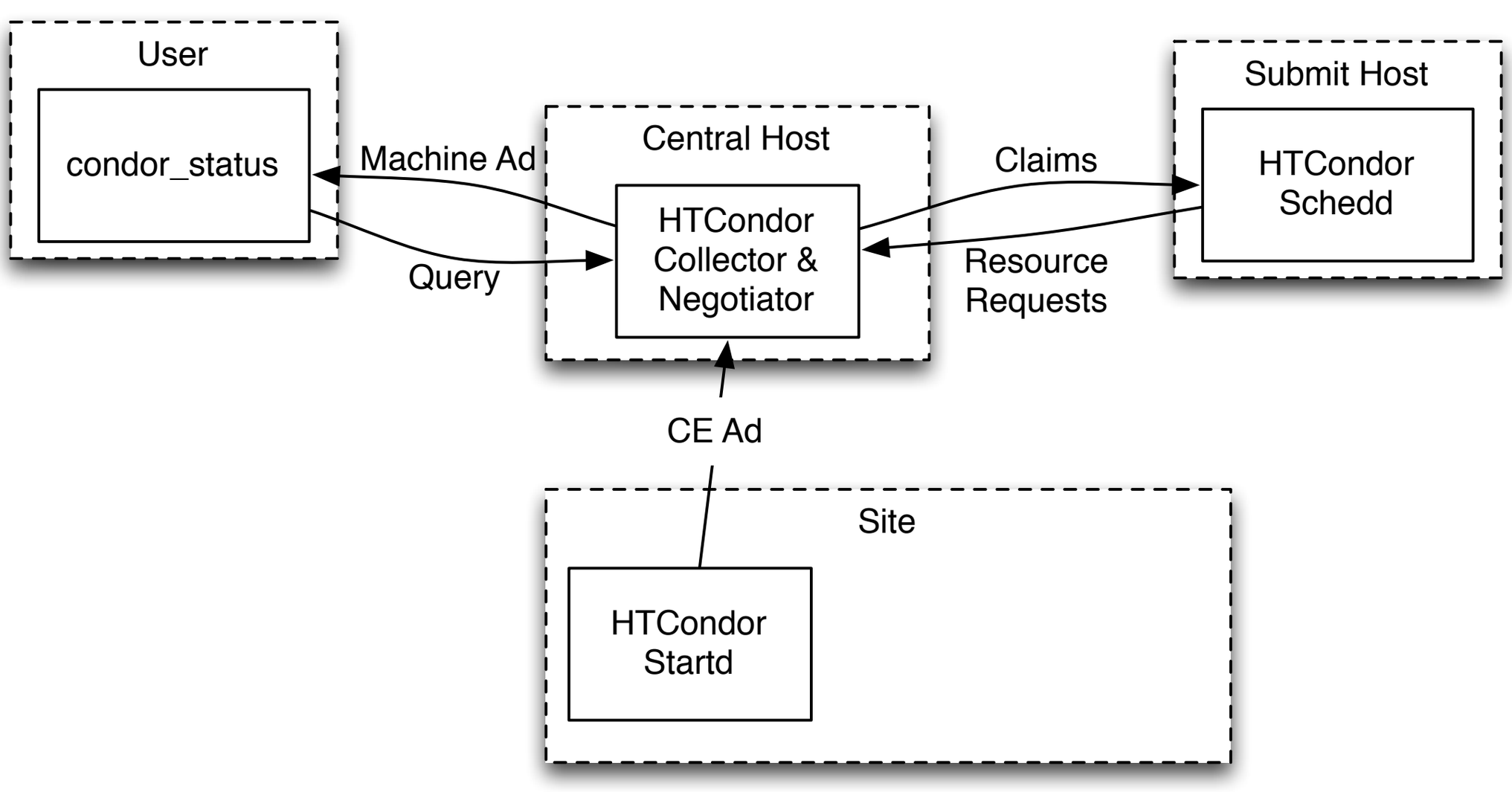
What resources?
Not how many!

- What's the most successful provisioning information service?



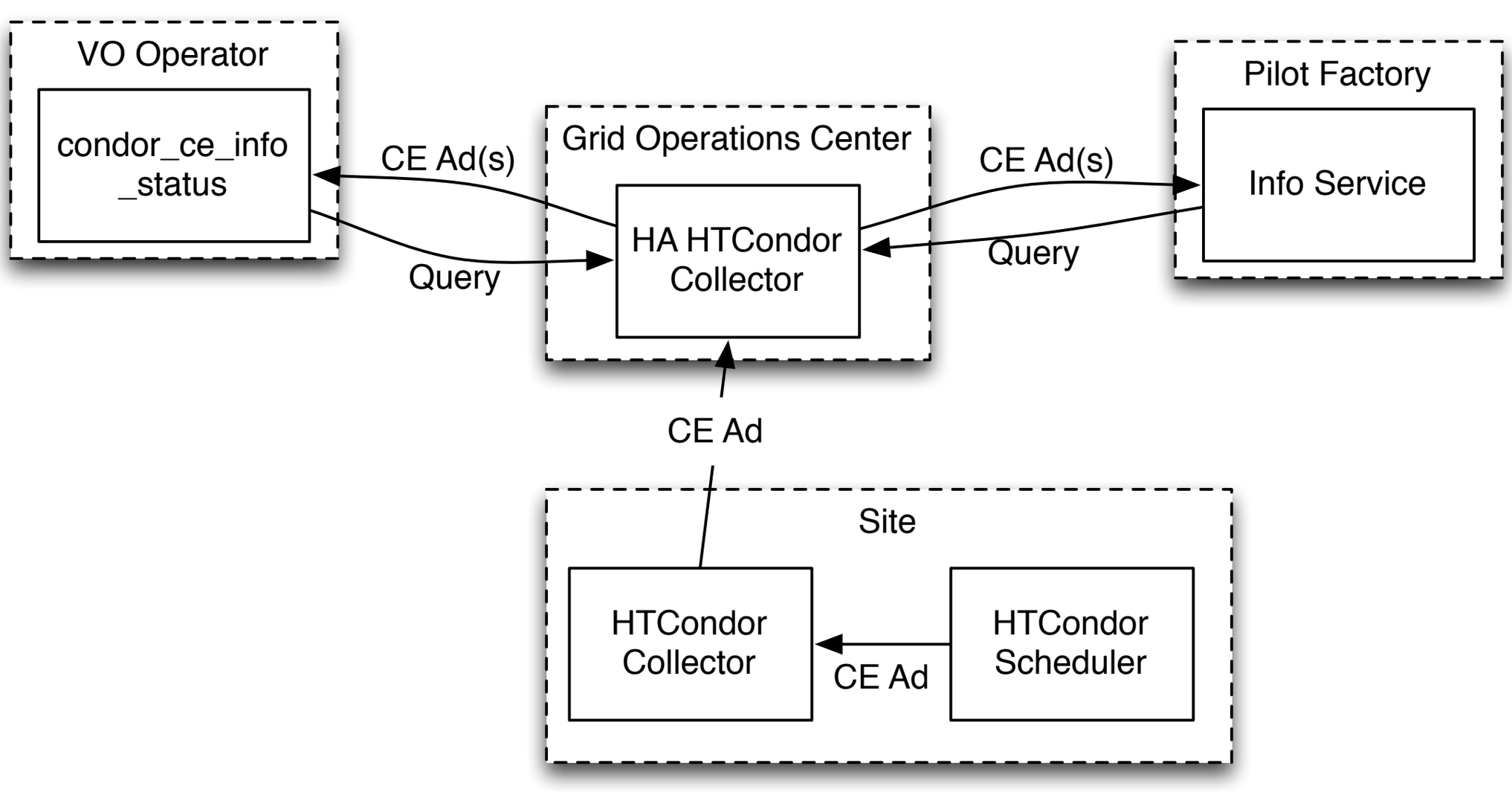
Basic Concept

- As with the CE itself, we borrow technology from the HTCondor team.



Basic Concept

- Run a central collector and aggregate a description of each CE that is operational.



The CE Ad

- Created by an HTCondor CE daemon (condor_schedd)
- Resource Catalog packaged up with other necessary information:
  - Address needed to submit jobs
  - Batch system in use on the CE

The CE Ad

- Created by the condor_sched
- Contains the `MyAddress` attribute for locating the daemon.
- Contains an ad expiration policy and generation timestamp.
- "Just" a HTCondor ClassAd. Hence, is **schema-free**: VOs can work with their sites to come up with domain-specific-extensions as needed.
    - Can express both values *and* policies (through expressions).

Resource Catalog 1

- An HTCondor CE classad attribute
- Generated by OSG Configure
- One entry per subcluster

```
OSG_ResourceCatalog = { \
   [  (Entry for blue)  ]; \
   [  (Entry for green)  ]; \
   [  (Entry for gray)  ]; \
}
```

Resource Catalog 2

```
[Subcluster Example_Gray]      [ \
name = Example_Gray  ------->   Name          = "Example_Gray"; \
cores_per_node = 12  ------->   CPUs          = 12; \
ram_mb = 32768       ------->   Memory        = 32768; \
                                Requirements = \
                                 TARGET.RequestCPUs   <= CPUs && \
                                 TARGET.RequestMemory <= Memory && \
                                Transform = [ \
                                 set_MaxMemory    = RequestMemory; \
                                 set_xcount       = RequestCPUs; \
                                ]; \
                              ] \
```

32GB, 12 core
Resource "gray"

Resource Catalog 3

```
[Subcluster Example_Blue]
name = Example_Blue
cores_per_node = 12
ram_mb = 32768
extra_transforms =
  set_WantRHEL7=1
```

32GB, 12 core
needs "WantRHEL7" set
Resource "blue"

```
[ \
Name          = "Example_Blue"; \
CPUs          = 12; \
Memory        = 32768; \
Requirements = \
 TARGET.RequestCPUs   <= CPUs && \
 TARGET.RequestMemory <= Memory && \
Transform = [ \
 set_MaxMemory    = RequestMemory; \
 set_xcount       = RequestCPUs; \
 set_WantRHEL7    = 1; \
]; \
] \
```

Resource Catalog 4

```
[Subcluster Example_Green]
name = Example_Green
cores_per_node = 24
ram_mb = 131072
max_wall_time = 720
allowed_vos = glow
```

128GB, 24 core
12 hour jobs
reserved for the glow VO
Resource "green"

```
[ \
Name         = "Example_Green"; \
CPUs         = 24; \
Memory       = 131072; \
MaxWallTime  = 720; \
AllowedVOs   = { "glow" }; \
Requirements = \
 TARGET.RequestCPUs   <= CPUs && \
 TARGET.RequestMemory <= Memory && \
 member(TARGET.VO, AllowedVOs); \
Transform = [ \
 set_MaxMemory    = RequestMemory; \
 set_xcount       = RequestCPUs; \
]; \
] \
```

Submit File Generation (WIP)

```
grid_resource = "condor ce.example.net ce.example.net:9619"
Transform = [
 set_MaxMemory    = RequestMemory;
 set_xcount       = RequestCPUs;
 set_WantRHEL7    = 1;
];
```



```
+GridResource = "condor ce.example.net ce.example.net:9619"
+MaxMemory    = RequestMemory
+xcount       = RequestCPUs
+WantRHEL7    = 1
```

CEView

- ClassAds are the lingua franca of the HTCondor ecosystem…
  - … but not the internet!
- Hence, we have developed a small webapp, **CEView**, for displaying CEs that are in a collector.
- Can be run in either **CE-local** or **grid-wide** mode.
  - In the latter case, exports contents of the collector in a JSON format optimized for AGIS.

CEView

sandhills-ce1.unl.edu

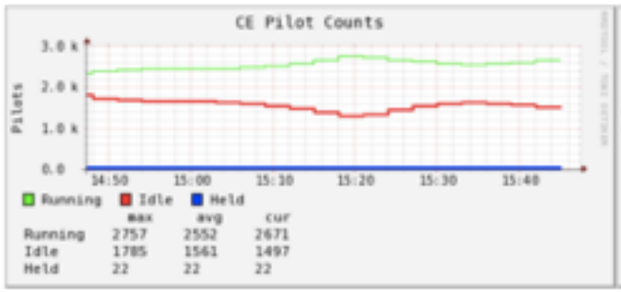
Brian

sandhills-ce1.unl.edu

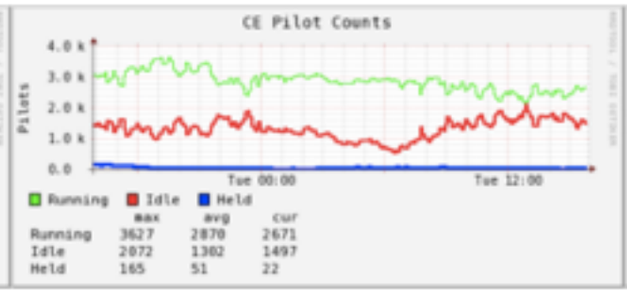
HTCondor-CE Overview | VOs | Pilots | Metrics | Health

Running	Idle	Held	Last Data Update
2686	1481	22	Tue Mar 01 2016 22:47:26 GMT+0100 (CET)

Last Hour



Last Day



Last Week

Pilots

Search Table

VO	VOMS	Jobs	Running	Idle	Held	DN
cms	/cms/Role=pilot	583	354	228	0	/DC=ch/DC=cern/OU=computers/CN=cmspilot02/vocms080.cern.ch
osg	/osg	2155	1414	733	6	/DC=com/DC=DigiCert-Grid/O=Open Science Grid/OU=Services/CN=pilot/osg-flock.grid.iu.edu
GLOW	/GLOW	58	56	2	0	/DC=com/DC=DigiCert-Grid/O=Open Science Grid/OU=Services/CN=pilot/glidein2.chtc.wisc.edu
cms	/cms/local/Role=pilot	4	4	0	0	/DC=ch/DC=cern/OU=computers/CN=cmspilot02/vocms080.cern.ch
fermilab	/fermilab/uboone/Role=pilot	24	18	6	0	/DC=com/DC=DigiCert-Grid/O=Open Science

Exploring other Resource Types

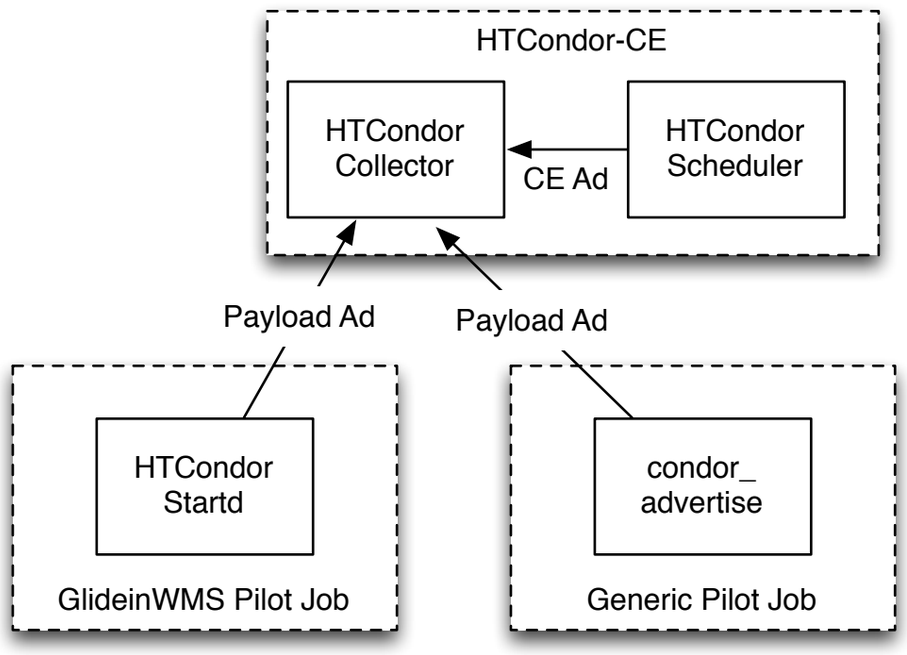
- OSG is starting to explore data access services - StashCache - for VOs who have cache-friendly data access patterns.
  - What kind of information services are needed for this type of resource?

- We currently produce minimal ads in the collector for these resources - but have not done a thorough treatment as with CEs.

- Should we expand wider? Collaborators are welcome!

CE Collector

- Each CE contains a condor_collector to help locate daemons.
  - We reuse this to forward the schedd ad
- Let’s reuse it again - allow pilots to advertise their payloads.



Wrapping Up

- By leveraging the combination of the ClassAds + condor_collector, we have bootstrapped an information system with a very different slant than the traditional GLUE/ BDII.
  - Does not aim to describe a grid site.
  - Describe resources and endpoints for provisioning.
  - Aims to provide visibility into payload jobs.
- **We are looking for partners** to invest in furthering these approaches!