An OSG-based distributed Campus Computing Infrastructure

Christoph Paus, MIT
CHEP Meeting, San Francisco
Oct 10-14, 2016

Ray an Maria Stata Center at MIT

Introduction Campus Computing

Description of generic campus situation (*ex.* MIT)

- Large variety of research areas: engineering, maths, sciences, social sciences ....
- Most need large computing *at times* and have some computing resources
  - some resources shared but not widely, usage not 100% for all of them
  - no accurate inventory of all existing resources
- Most resources use some linux variant of similar versions

Some Issues

- Researches have peak demands that exceed their resources
- Often though the resources are not fully used
- Account management is work intensive

Introduction Campus Computing

Big computing centers do not work

Requirements for a new model

- Minimally expensive: money and human resources
- Technically feasible and attractive for most research efforts
- Use existing resources, but leave owners maximal control
- Reach beyond campus as needed/wanted

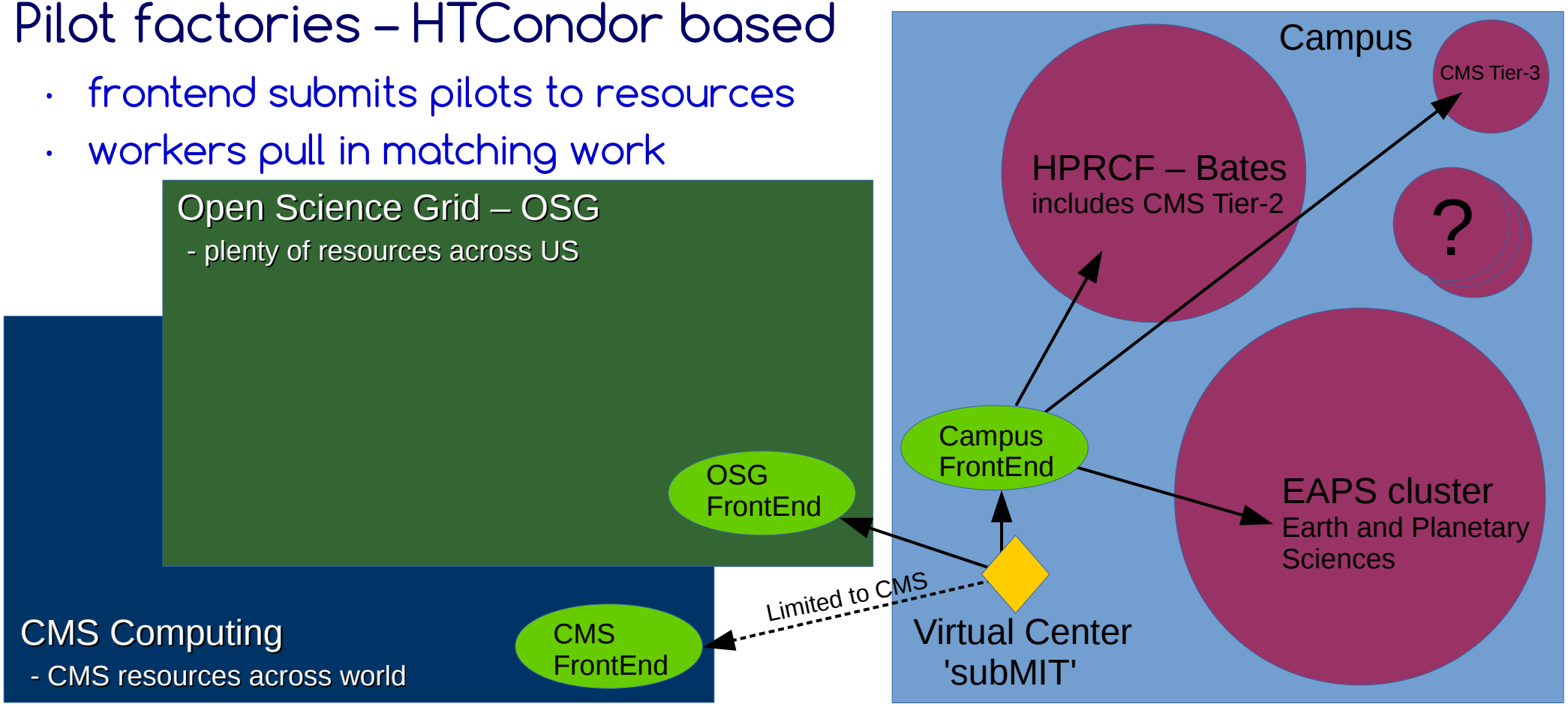
Virtual Computing Center: Pretend to be big

- Create a common login pool, big enough for tests
- Connect this pool to each campus computing resource
- Also connect to external resources: *ex.* OSG

Implemetation Details – MIT

Pilot factories – HTCondor based

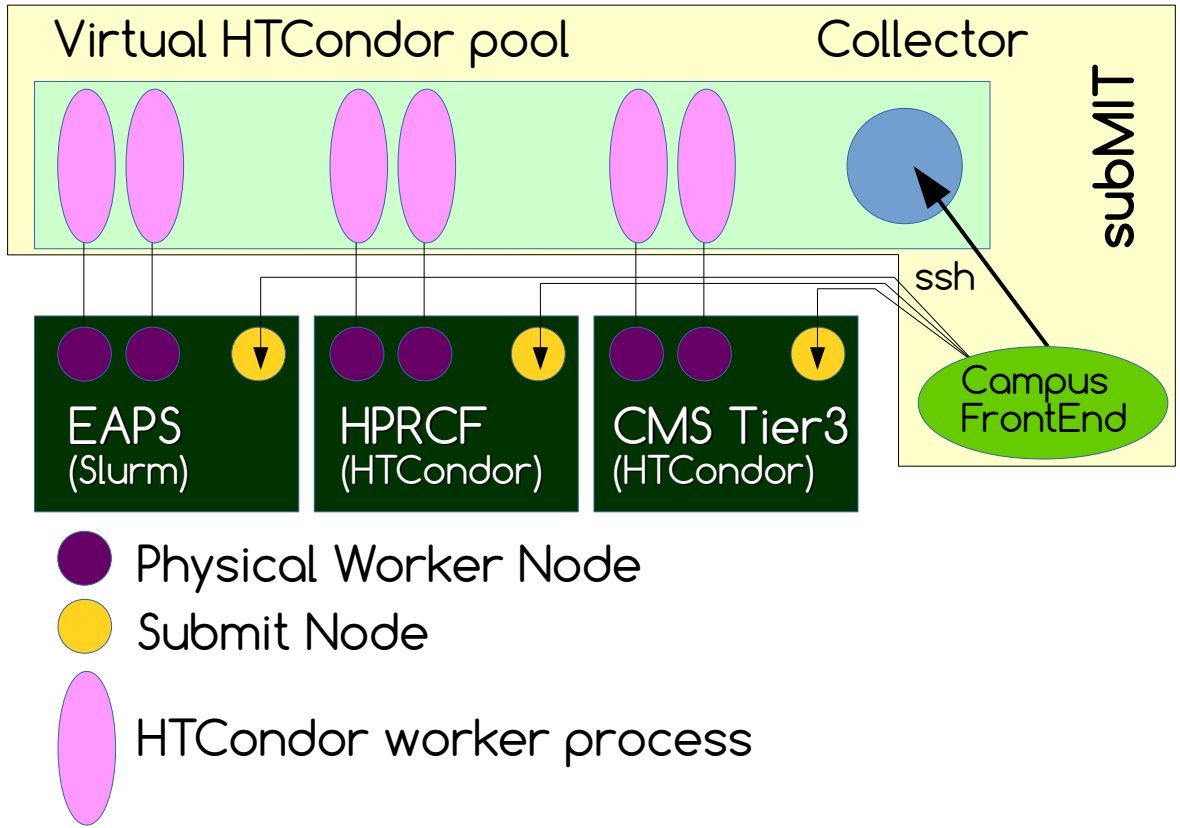
- frontend submits pilots to resources
- workers pull in matching work



Implementation Details – MIT

How are jobs running?

- FrontEnd submits glideIn pilots through BOSCO to the various resources submit nodes (local flavor)
- On subMIT user jobs get submitted to a HTCondor collector
- Physical workers are matched at subMIT and pull down their work
- subMIT becomes a huge virtual resource, real work is done at the physical worker
- At completion output is shipped as specified



Conclusions

The Virtual Computing Center

- Viable, pragmatic solution for generic campus computing
- Covers most use cases, *but not all*
- Allows maximal flexibility: all resources can be separately registered and used, but also controlled by owners
- Specific fully functional prototype implemented at MIT using OSG based tools: HTCondor, bosco, glideInWMS pilots
- Prototype provided over 30M CPU hours to MIT based researchers

What next?

- Some investment needed to establish infrastructure and support
- Users need to re-learn some, but win big
- Resource owners need to be convinced and have to adjust
- Need to find all other resources on campus and connect them