Distributed Computing with HEP Cloud, GlideinWMS and HTCondor

Marco Mambelli

IF Computing School

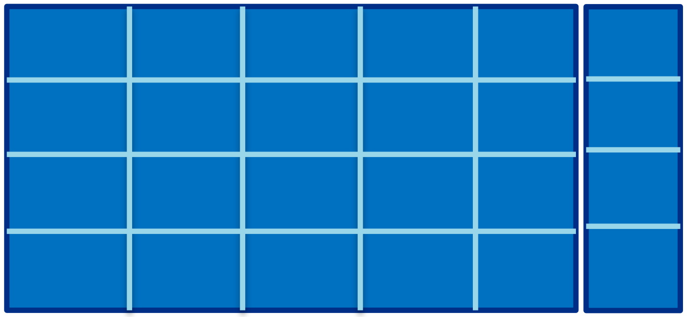
June 21 2021

Outline

- Distributed High Throughput Computing
- Pilot-based systems
- GlideinWMS and HEPCloud
- Storage and credentials
- HTCondor
- Resources and job requirements

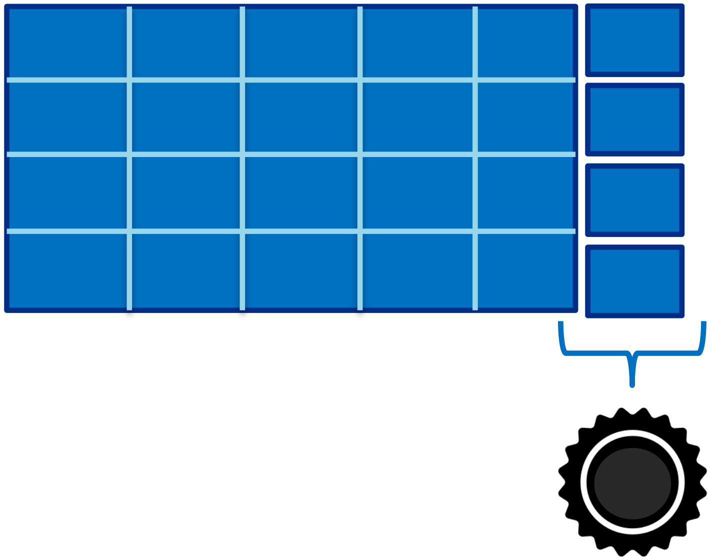
distributed High Throughput Computing (dHTC)

- Tasks split in small pieces (jobs)



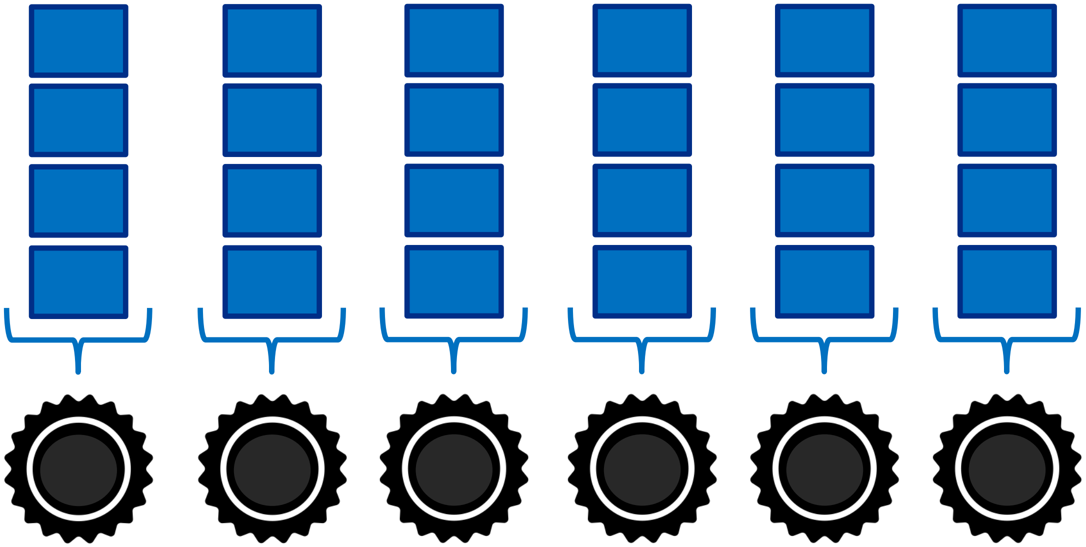
distributed High Throughput Computing (dHTC)

- Tasks split in small pieces (jobs)
- Resource processing queued jobs



distributed High Throughput Computing (dHTC)

- Tasks split in small pieces (jobs)
- Resource processing queued jobs
- Run many jobs in parallel to shorten completion



Where jobs run

- Your computer
  - Interactive
  - GUI
  - Your customization
  - Your software

- Institutional cluster
  - Batch queue (SLURM, PBS, HTCondor, SGE, …)
  - Terminal
  - Network access
  - Familiar environment
  - Local support

Where jobs run (2)

OSG Compute Federation
Open Science Grid
UCSD
Canada
North Atlantic Ocean
Norway
United Kingdom
Spain
Italy
Ukraine
Kazakhstan
Mongolia
Turkey
Iraq
Iran
Afghanistan
Pakistan
China
India
Thailand
Algeria
Libya
Egypt
Saudi Arabia
Mali
Niger
Chad
Sudan
Nigeria
Ethiopia
DRC
Kenya
Tanzania
Angola
Namibia
Madagascar
Indonesia
Venezuela
Colombia
Brazil
Peru
Bolivia
South
Indian Ocean

- Grid clusters
  - Borrowed resources
  - Network reachable
  - Unknown environment
  - Multi-institution support system

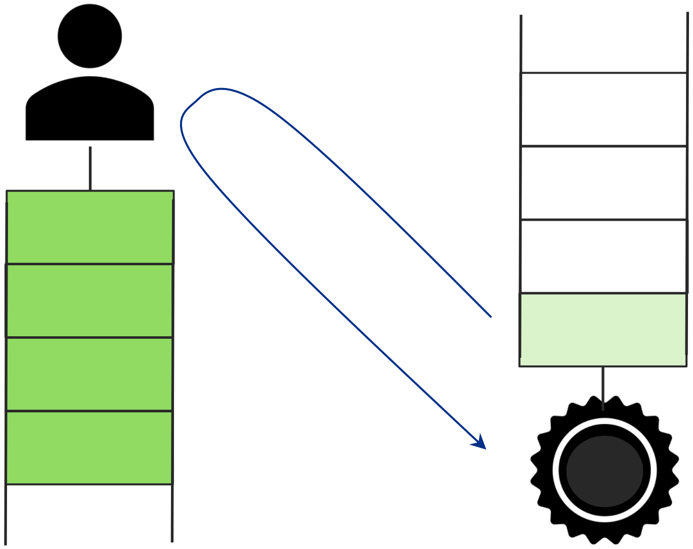
- (Commercial) Cloud
  - Rented resources
  - Virtual machines

Where jobs run (3)

- High Performance Computers (HPC)
  - Each is unique
    - Architecture
    - Network topology
  - Parallel and coupled jobs (MPI)
  - Allocations and long queue times

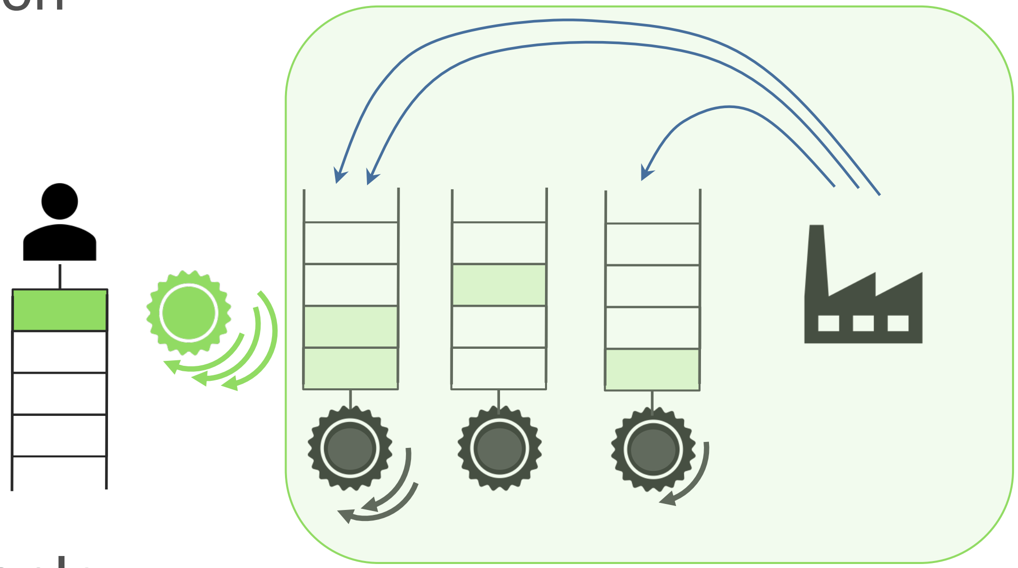
Pilot jobs (Glideins)

- Separation of tasks
  - Pilot job
    - Test
    - Set up
    - “Expendable”
  - User/real job
    - Science
- Late binding
- Flexible use of multiple resources



Overlay system

- Pilot layer
  - Distributed computing knowledge and troubleshooting
  - Reduce heterogeneity
  - Handle different speeds
  - Pressure-based submission
- Virtual cluster
  - Domain knowledge and troubleshooting
  - Elastic

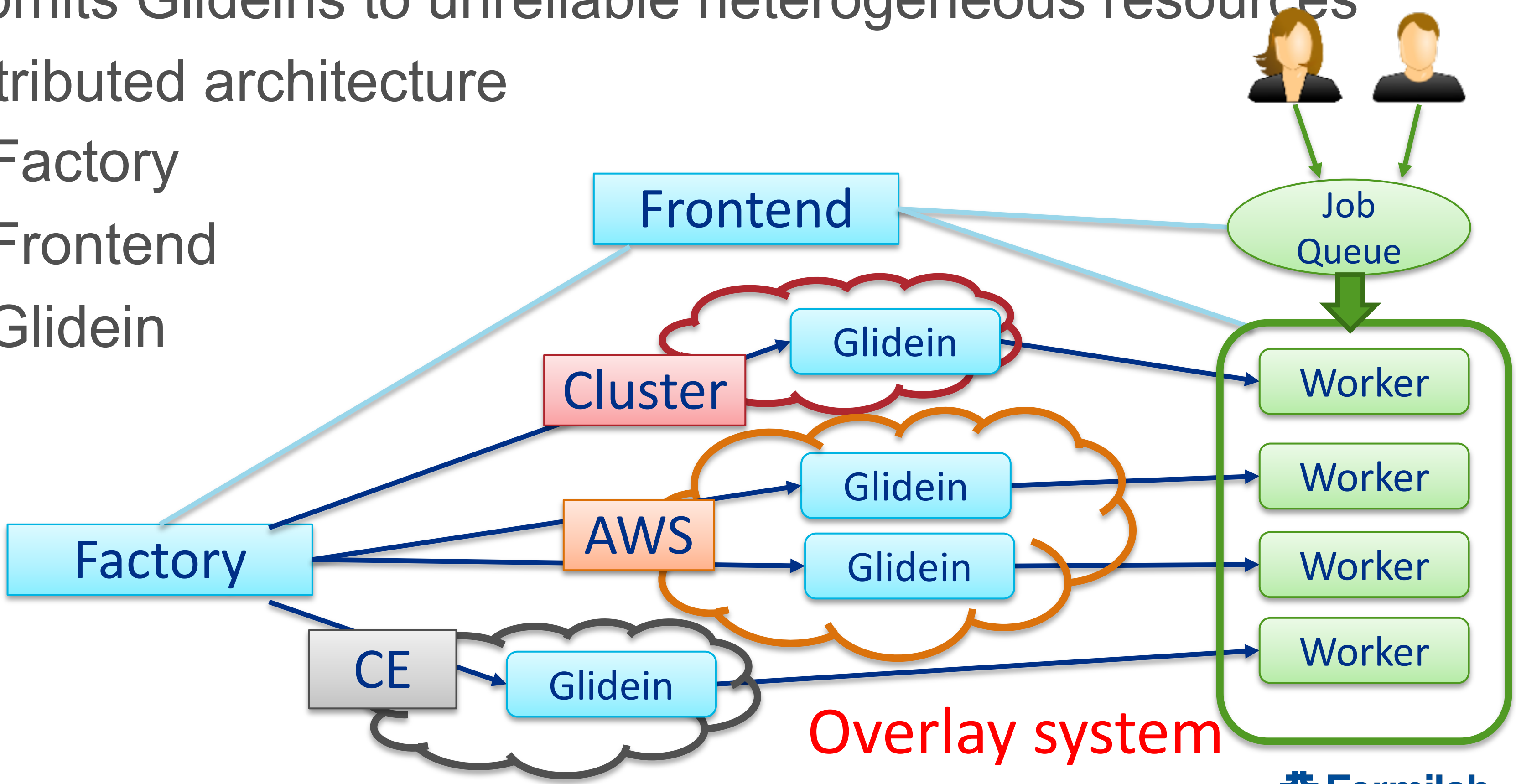


- Separation for software, systems and people

GlideinWMS

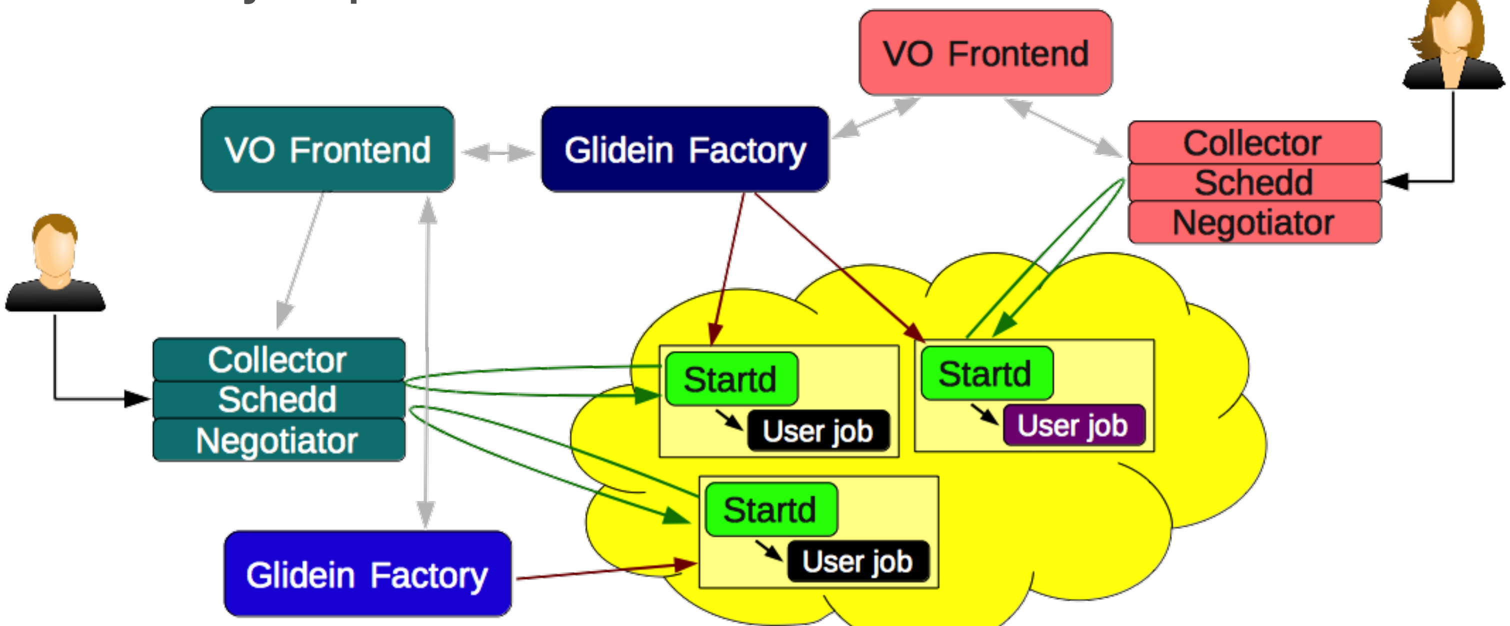
GlideinWMS is a pilot based resource provisioning tool for distributed High Throughput Computing

- Provides reliable and uniform HTCondor virtual clusters
- Submits Glideins to unreliable heterogeneous resources
- Distributed architecture
  - Factory
  - Frontend
  - Glidein



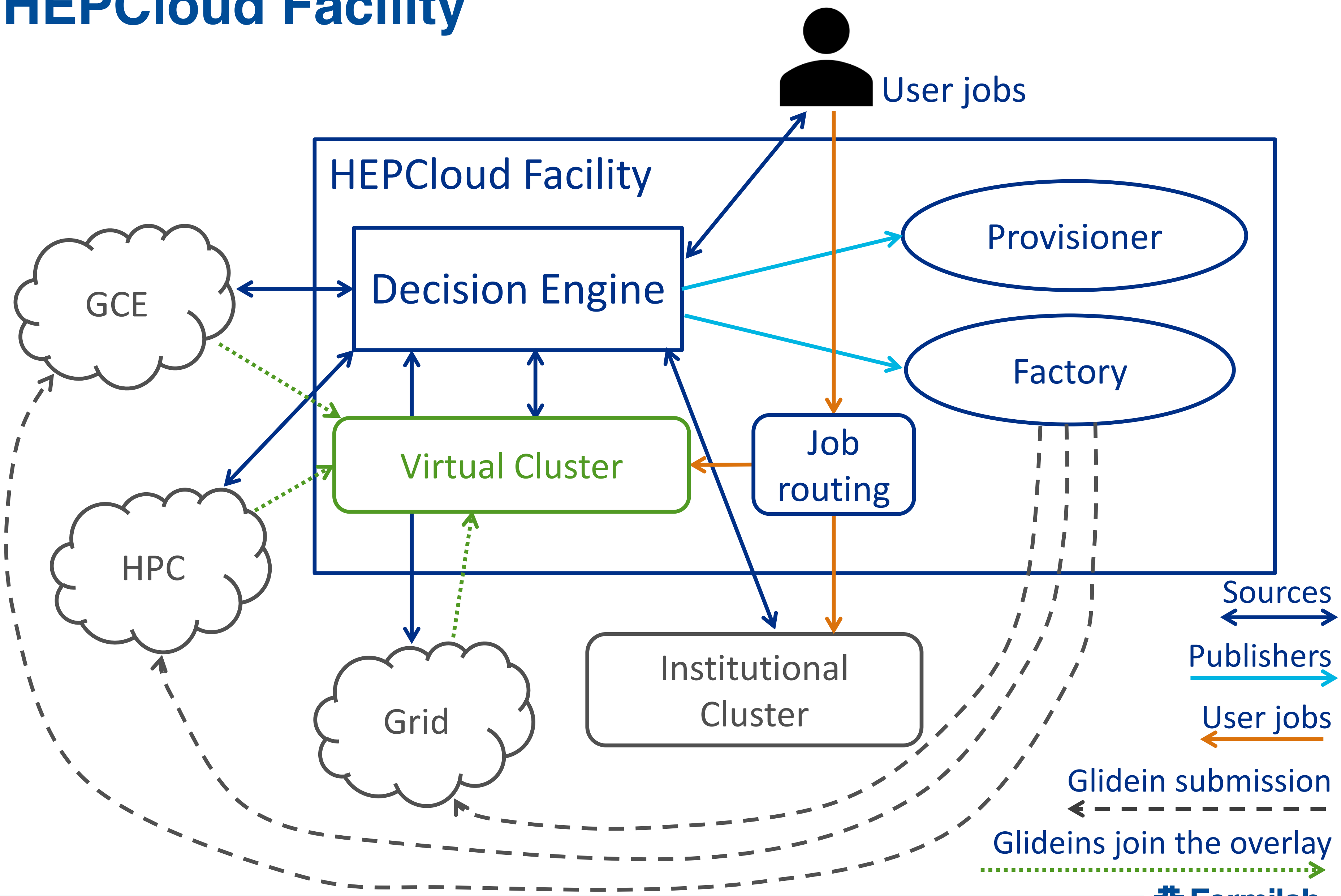
Distributed

- N-to-M relationship
  - Each Frontend can talk to many Factories
  - Each Factory may serve many Frontends
- Multiple User Pools
- High Availability replicas

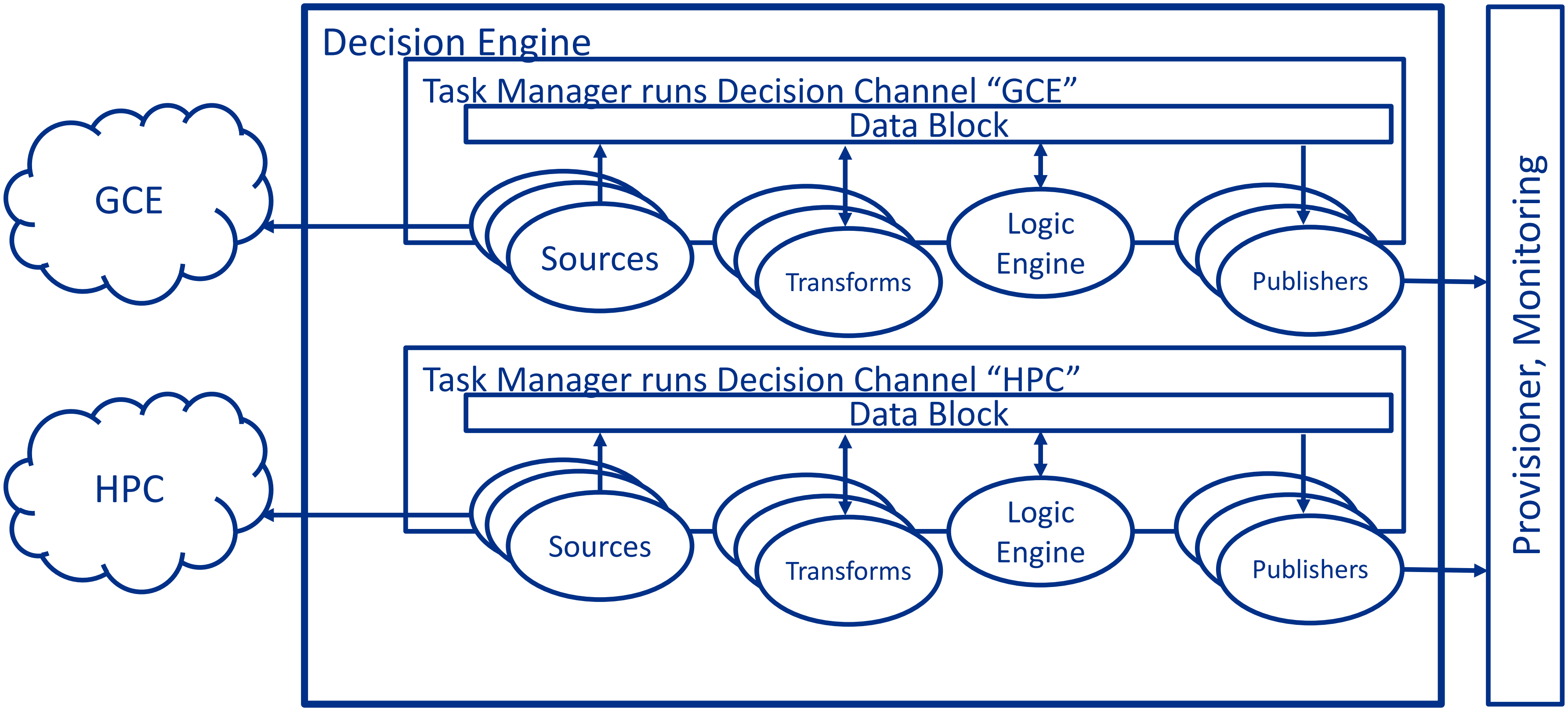


S.Timm - FNAL-UK Planning Meeting GlideinWMS

HEPCloud Facility



Decision Engine



Storage types summary

- System Volumes
  - Read only
- Locally Mounter Volumes (Local or RAM disk)
  - CWD (Current Work Directory)
  - TMP
- Interactive Storage Volumes (NAS - NFS, GPFS, Luster, …)
  - Shared file systems
  - Shared home directories
- Grid-accessible storage volumes
  - Distributed file system (HDFS, dCache, Xrootd)
  - Storage Element
- CernVM FS (CVMFS)
  - Write once read everywhere HTTP based distributed FS

Credential types

- X509 Certificate and Proxy
  - VOMS Extension
  - Identity based (you and your affiliations)
- JASON Web Token
  - SciToken
  - IDTOKEN
  - WLCG (IAM) token
  - Bearer token (capabilitybased)

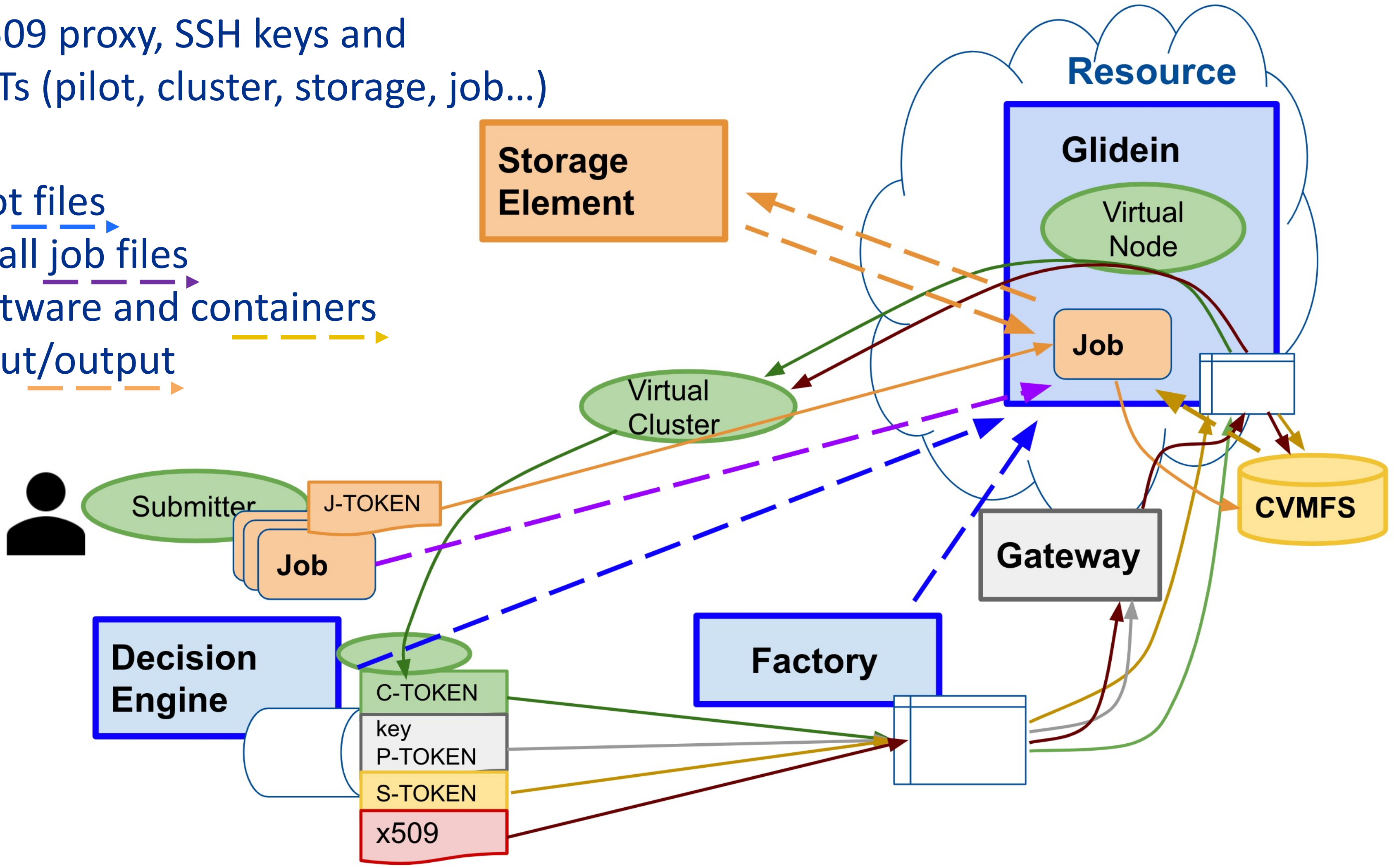
Credentials and data movement in a Glidein

Credentials

- x509 proxy, SSH keys and JWTs (pilot, cluster, storage, job…)

Data

- Pilot files
- Small job files
- Software and containers
- Input/output



HTCondor and ClassAds

- HTCondor is a Workload Management System (batch system)
  - Open source, robust, flexible, local (UW Madison)
- HTCondor principles: two parts of the equation
  - Jobs: quanta of work
  - Machines: available resources
- ClassAds is a language for objects (jobs and machines) to
  - Express attributes about themselves
  - Express what they require/desire in a match (similar to personal classified ads)
  - Structure
    - Set of attribute name/value pairs
    - Value : Literals (string, bool, int, float or an expression)

Example Match

Pet Ad

MyType = “Pet”

TargetType = “Buyer”

Requirements =

DogLover =?= True

Rank = 0

PetType = “Dog”

Color = “Brown”

Price = 75

Breed = "Saint Bernard"

Size = "Very Large"

...

Buyer Ad

MyType = “Buyer”

TargetType = “Pet”

Requirements =

(PetType == “Dog”) &&

(TARGET.Price <= MY.AcctBalance) &&

(Size == "Large"||Size == "Very Large")

Rank = (Breed == "Saint Bernard")

AcctBalance = 100

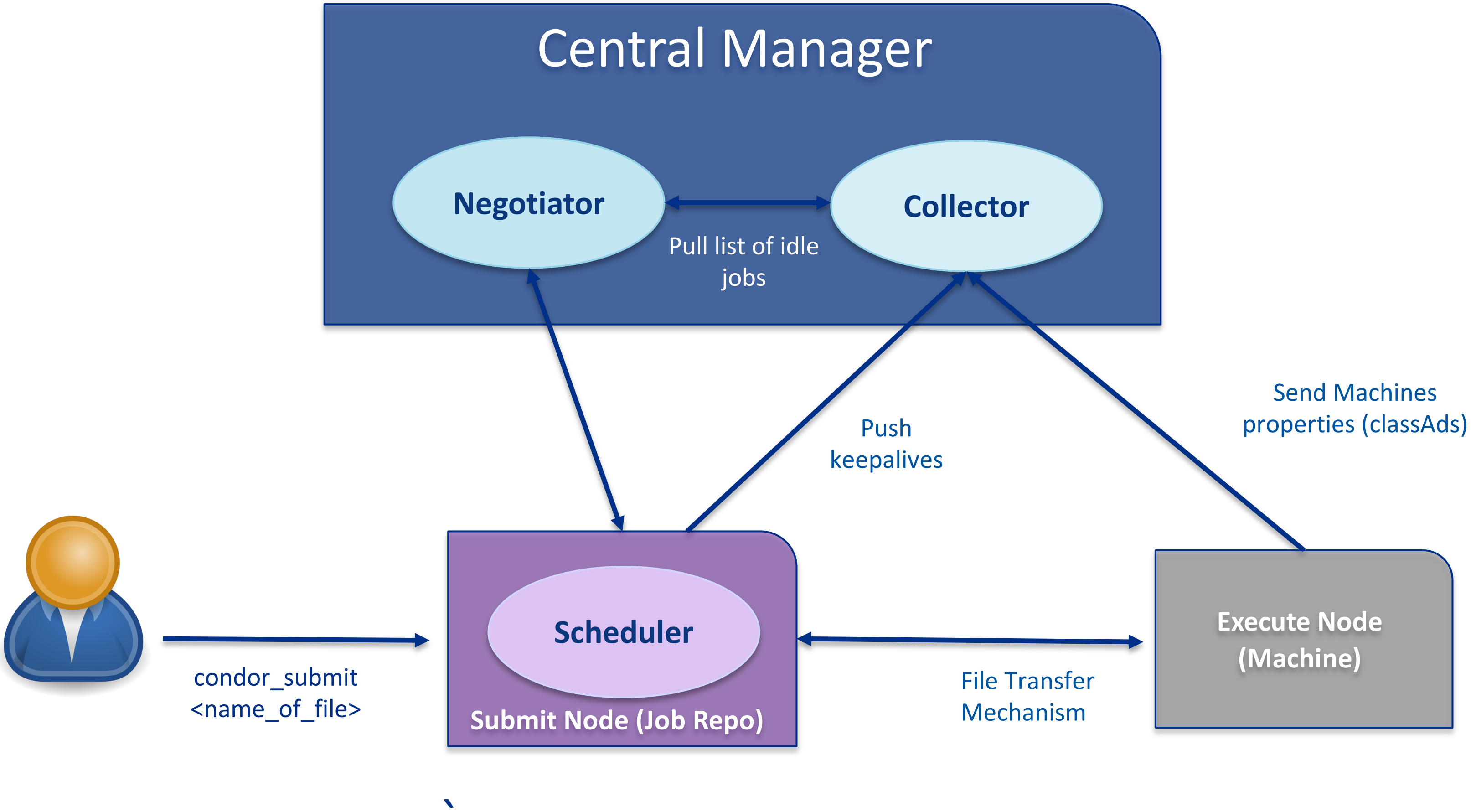
DogLover = True

. . .

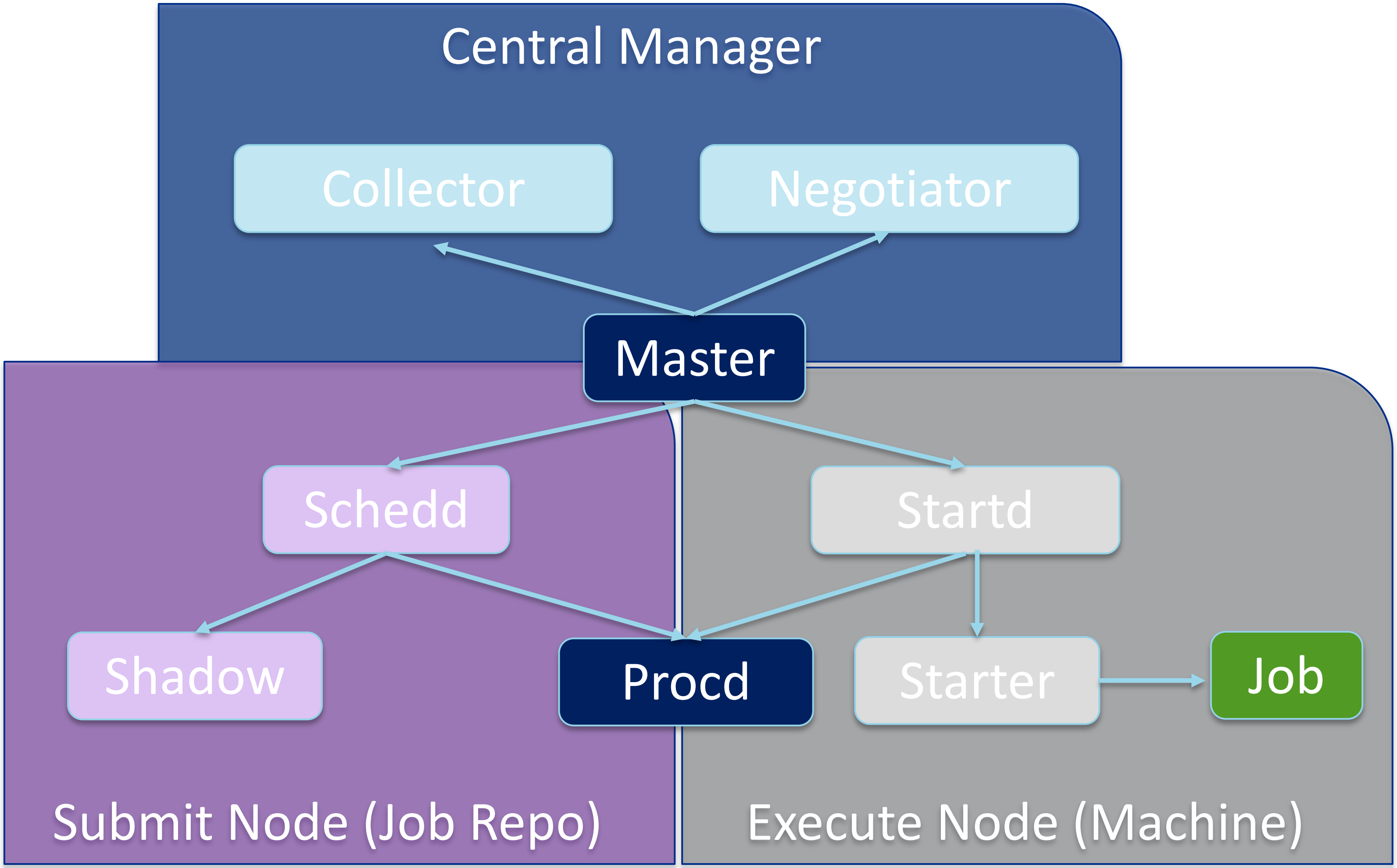
Dog == Resource ~= Machine

Buyer ~= Job

HTCondor components

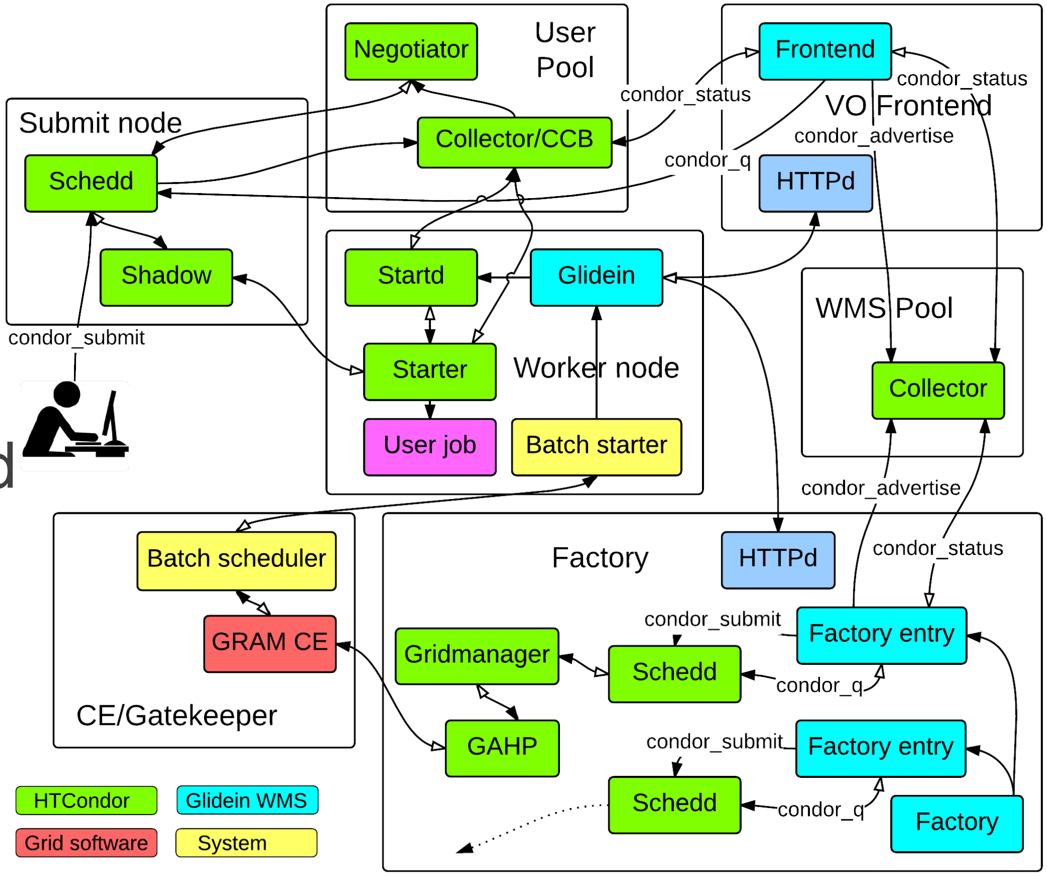


HTCondor components (daemons)



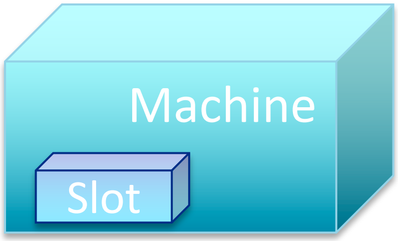
HTCondor building blocks in Glidein WMS

- The Factory works with an HTCondor pool, WMS pool, to submit Glideins to different resources
- The HTCondor Glideins are pilots that launch a startd that registers on a second HTCondor pool, User pool
- User jobs are matched and execute on the resources
- The Frontend monitors the user schedds and notifies the Factory about the need for more Glideins



Glideins run on Machines

- This is a machine (worker node, host, node, resource), managed by a (Local) Resource Manager
- More frequently virtual than not
- Characterized by its resources (dimensions):
  - CPUs (or total number of cores)
  - RAM (memory)
  - Disk



- There can be other special resources that the node provides: GPUs, access to devices, software, …
- The Glidein will receive all the node or part of it
- Sometime is not easy to identify everything used by a job

Job and Machine ‘dimensions’

- Job request
  - request_cpus: number of cores, integer, default 1.
  - request_disk: amount of disk space in Kbytes, default to sum of sizes of the job's executable and all input files (or image size)
  - request_memory: amount of memory space in Mbytes, default to executable size
- Machine
  - Cpus: number of cores, integer, by default the available cores
  - Disk: amount of disk space on this machine available for the job in KiB, by default the available space
  - Memory: amount of RAM in MiB in this slot

- Over and Under provision are possible

Summary

- Your jobs can run on many different resource types
  - Many have specific advantages/limitations
- GlideinWMS and HEPCloud help moving jobs around using Glideins
- HTCondor is used in many components
- Test your jobs locally
- Specify all the requirements