

HTCCondor

Architecture

...or, who talks to what?

HTCondor Week 2019

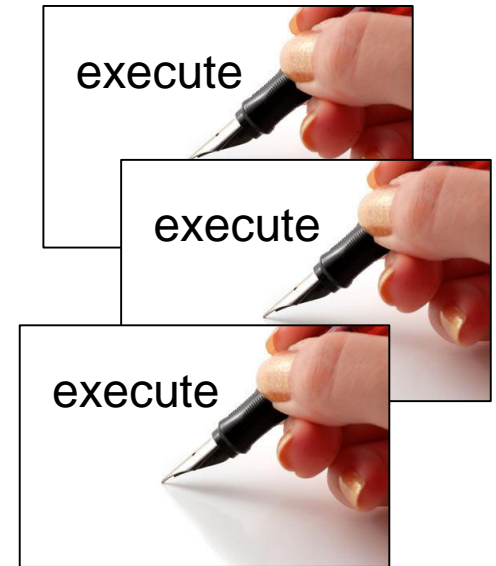
Todd Tannenbaum
Center for High Throughput Computing

HTCondor

› Jobs



› Machines



We are going to fill in the boxes!

Central Manager

Submit Machine

Execute Machine

ClassAds: The *lingua franca* of HTCondor



What are ClassAds?

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, where the value can be a literal **or an expression**. Semi-structured, no fixed schema.

ClassAd Types

- › HTCondor has many types of ClassAds
 - A "Job Ad" represents a job to Condor
 - A "Machine Ad" represents a computing resource
 - Others types of ads represent other instances of other services (daemons), users, accounting records.

Simple Example

Job Ad

```
Type = "Job"
Requirements =
  HasMatlabLicense
  == True &&
  Memory >= 1024
Rank = kflops + 1000000
  * Memory
Cmd= "/bin/sleep"
Args = "3600"
Owner = "gthain"
NumJobStarts = 8
KindOfJob = "simulation"
Department = "Math"
```

Machine Ad

```
Type = "Machine"
Cpus = 40
Memory = 2048
Requirements =
  (Owner == "gthain") ||
  (KindOfJob ==
  "simulation")
Rank = Department == "Math"
HasMatlabLicense = true
MaxTries = 4
kflops = 41403
```



Architecture & Job Startup

"LUNAR Launch" by Steve Jurvetson ("jurvetson") © 2006

Licensed under the Creative Commons Attribution 2.0 license.

<http://www.flickr.com/photos/jurvetson/114406979/>

<http://www.webcitation.org/5XIfTI6tX>

Quick Review of Daemons

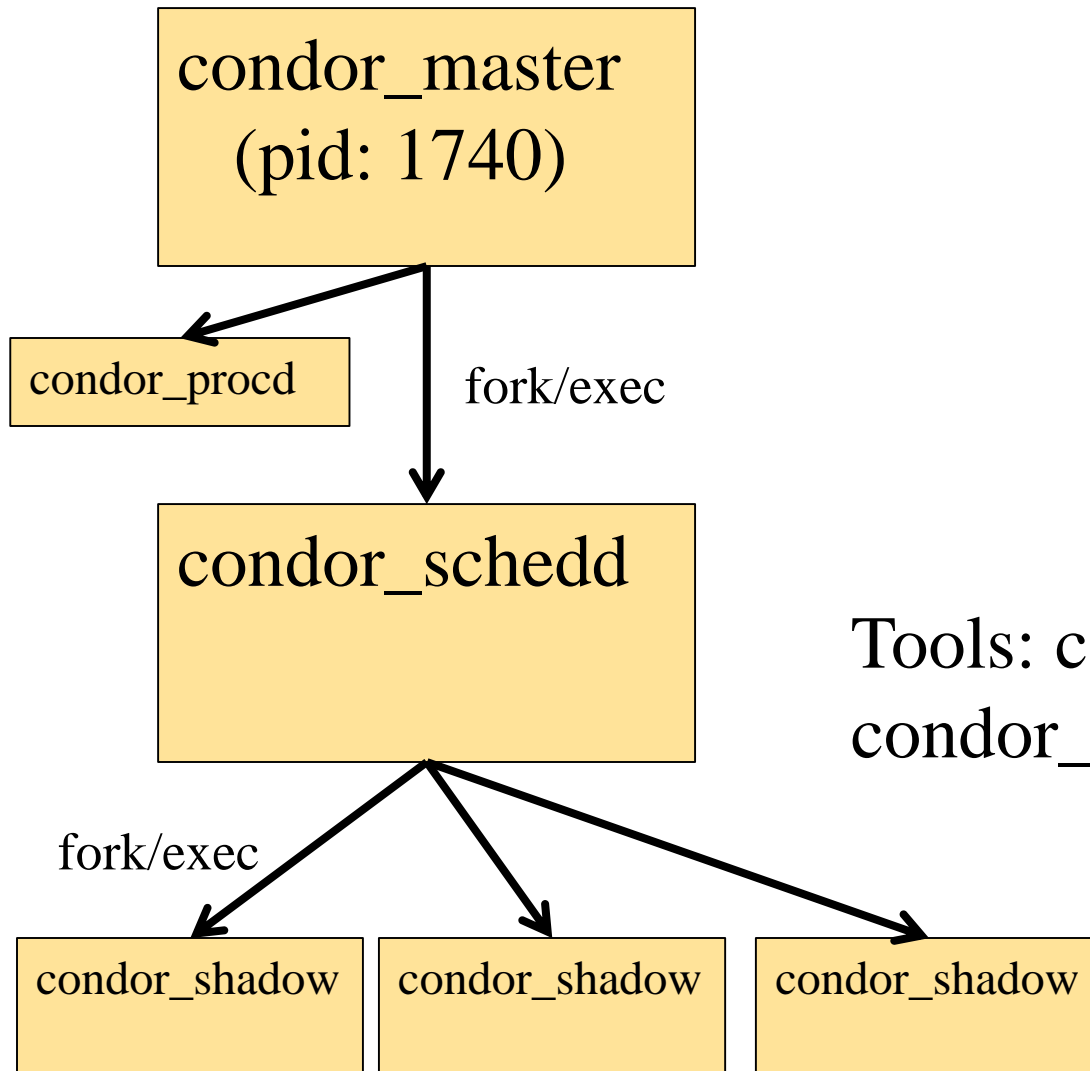
condor_master: runs on all machine, always plus a condor_procd, condor_shared_port

condor_schedd: runs on submit machine

condor_startd: runs on execute machine

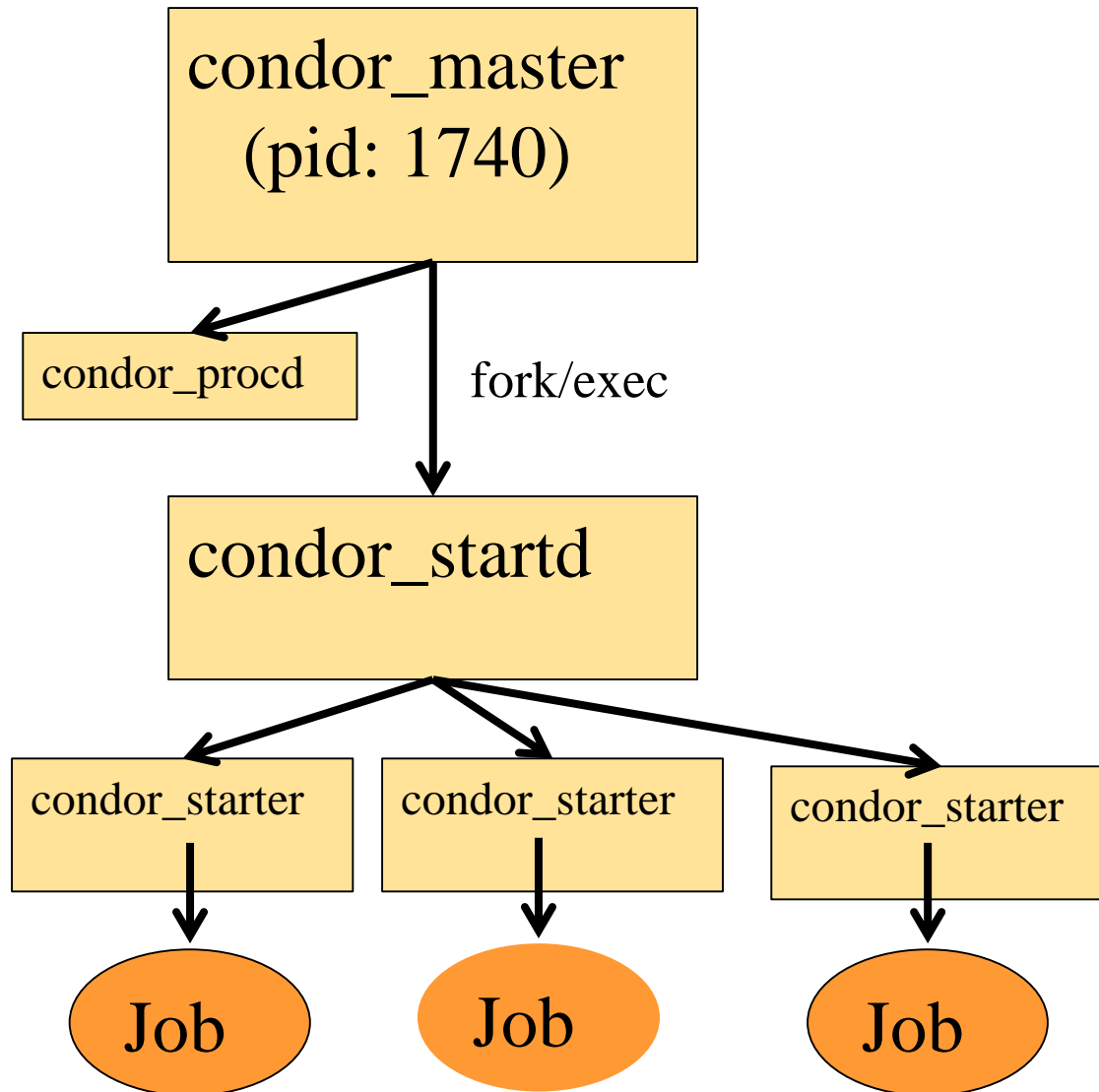
condor_negotiator, condor_collector: runs on central manager

Submit Machine Process View

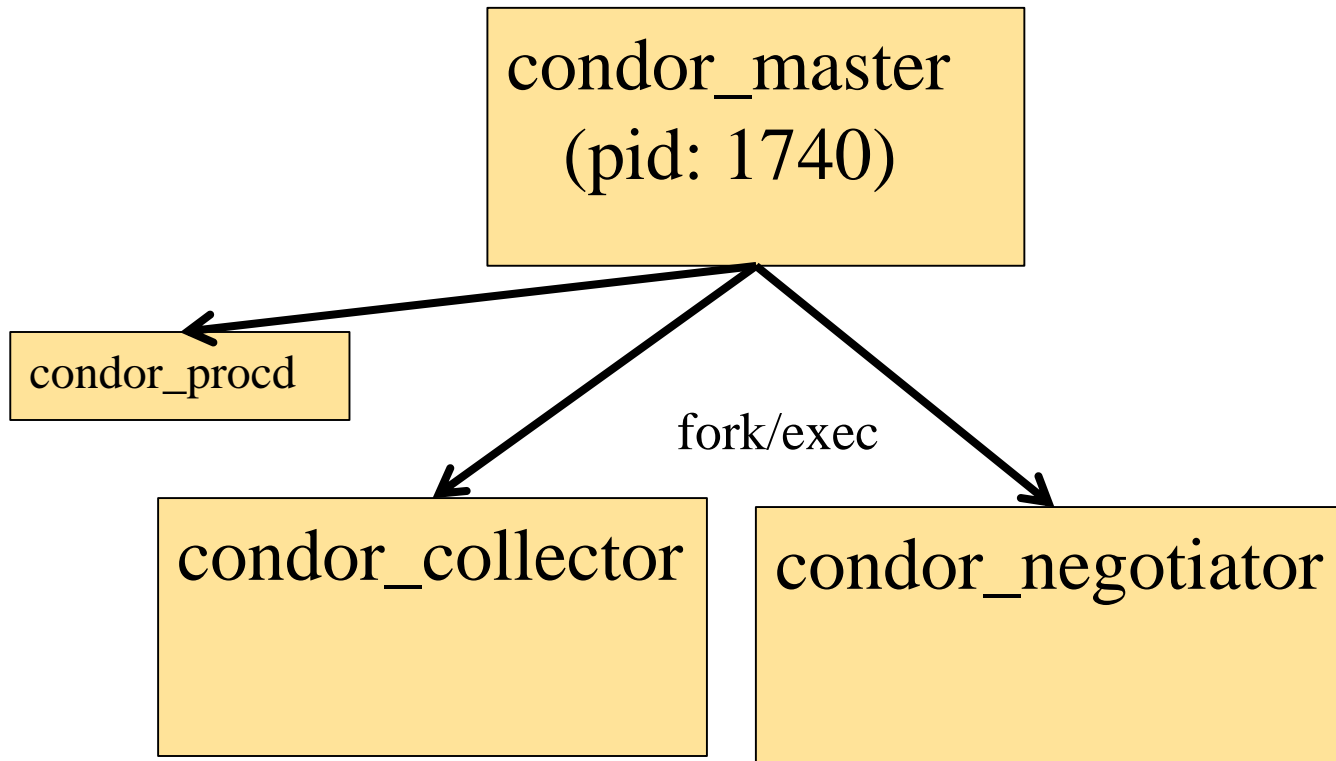


Tools: `condor_submit`, `condor_q`,
`condor_rm`, `condor_hold`, ...

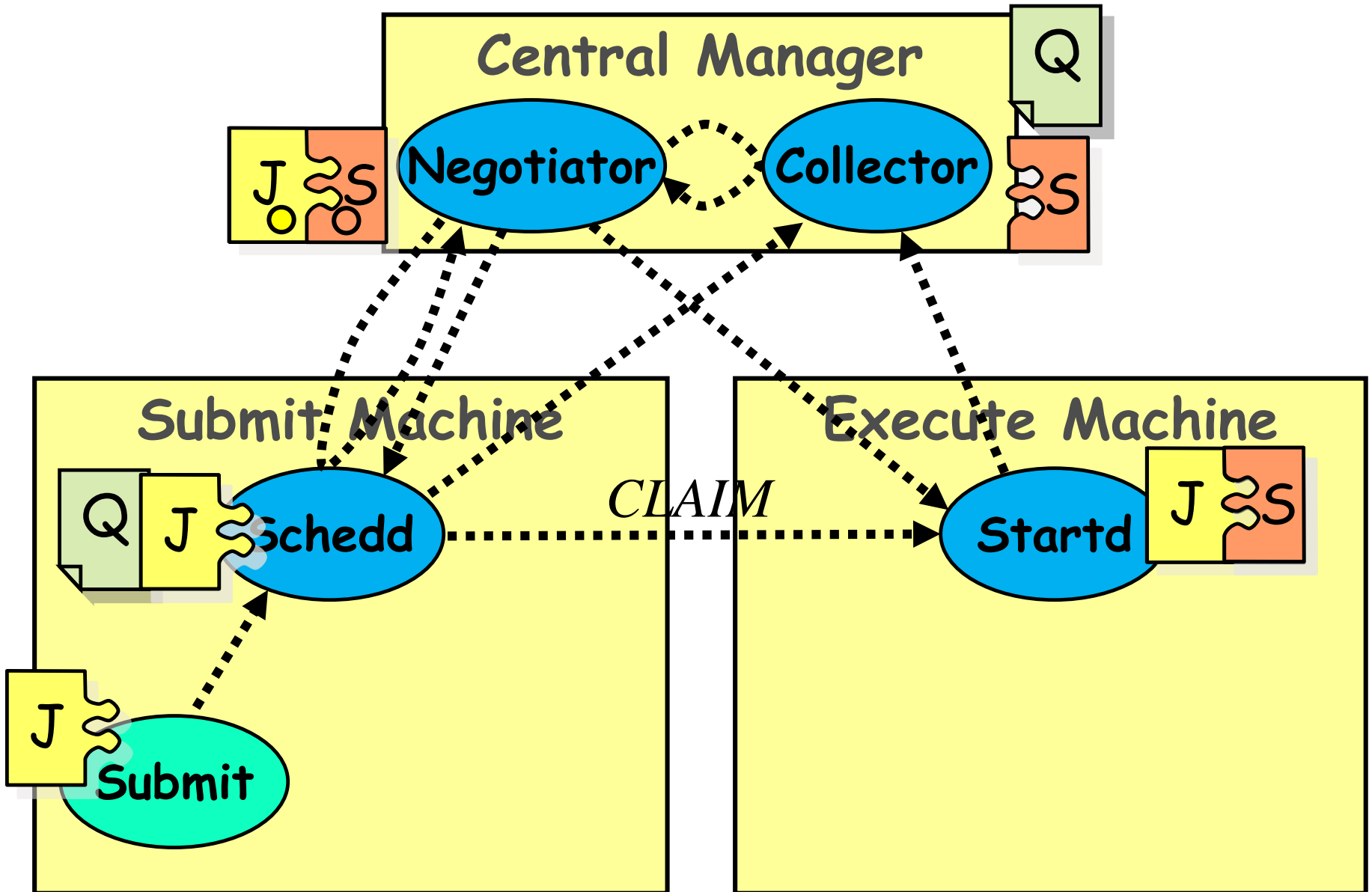
Execute Machine Process View



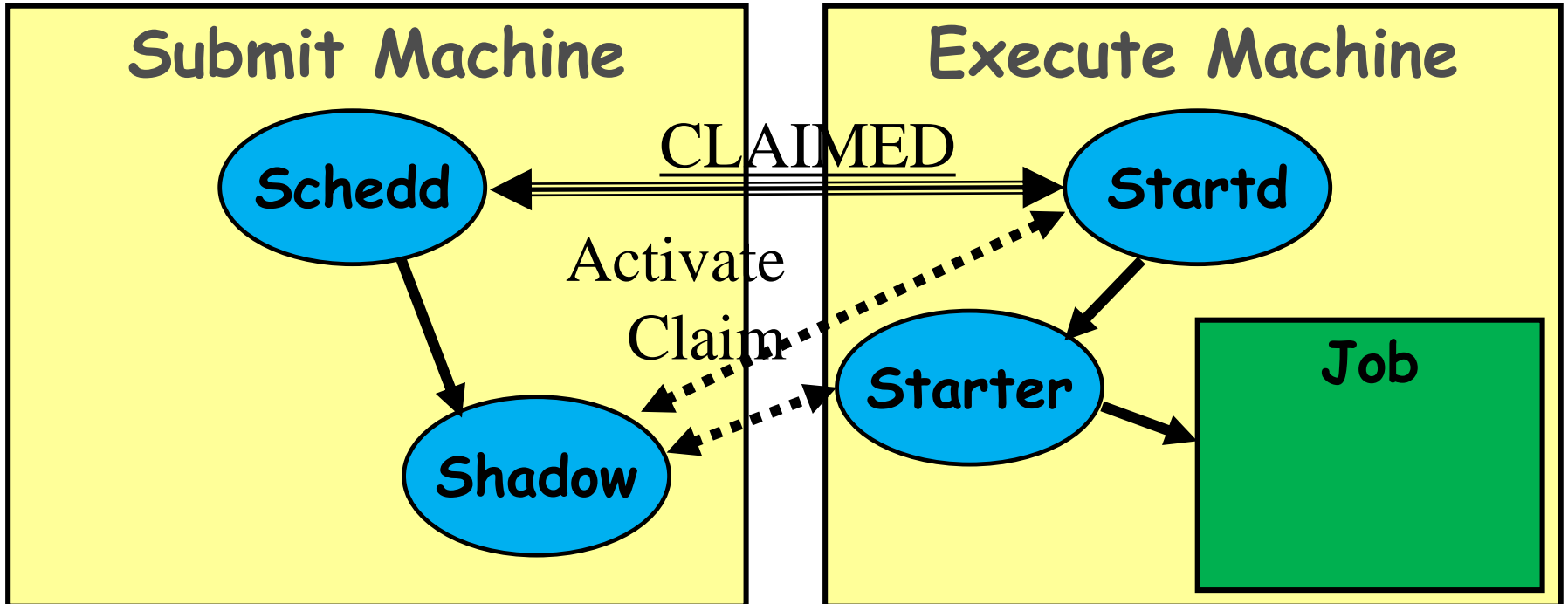
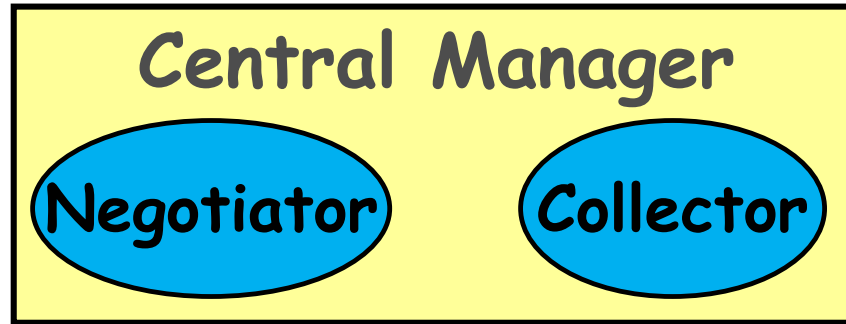
Central Manager Process View



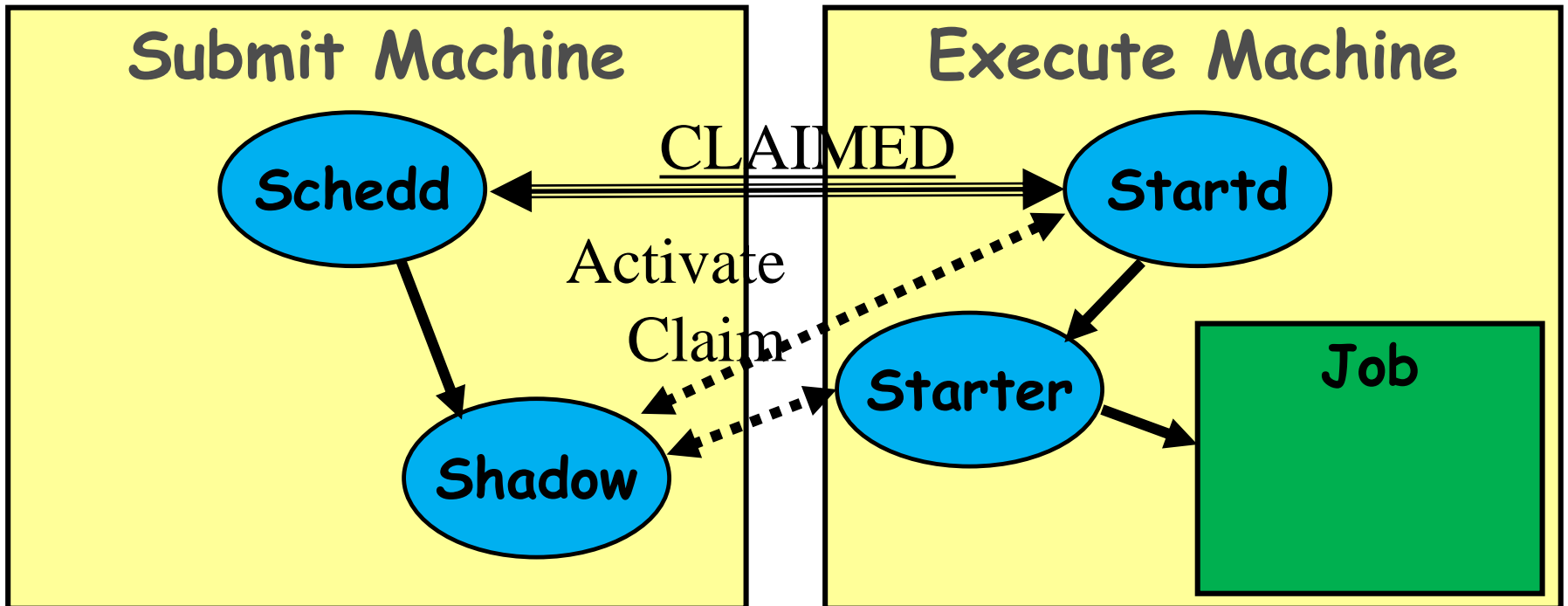
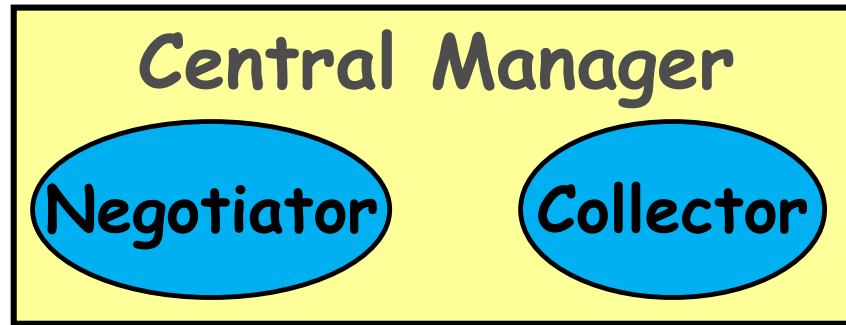
Claiming Protocol



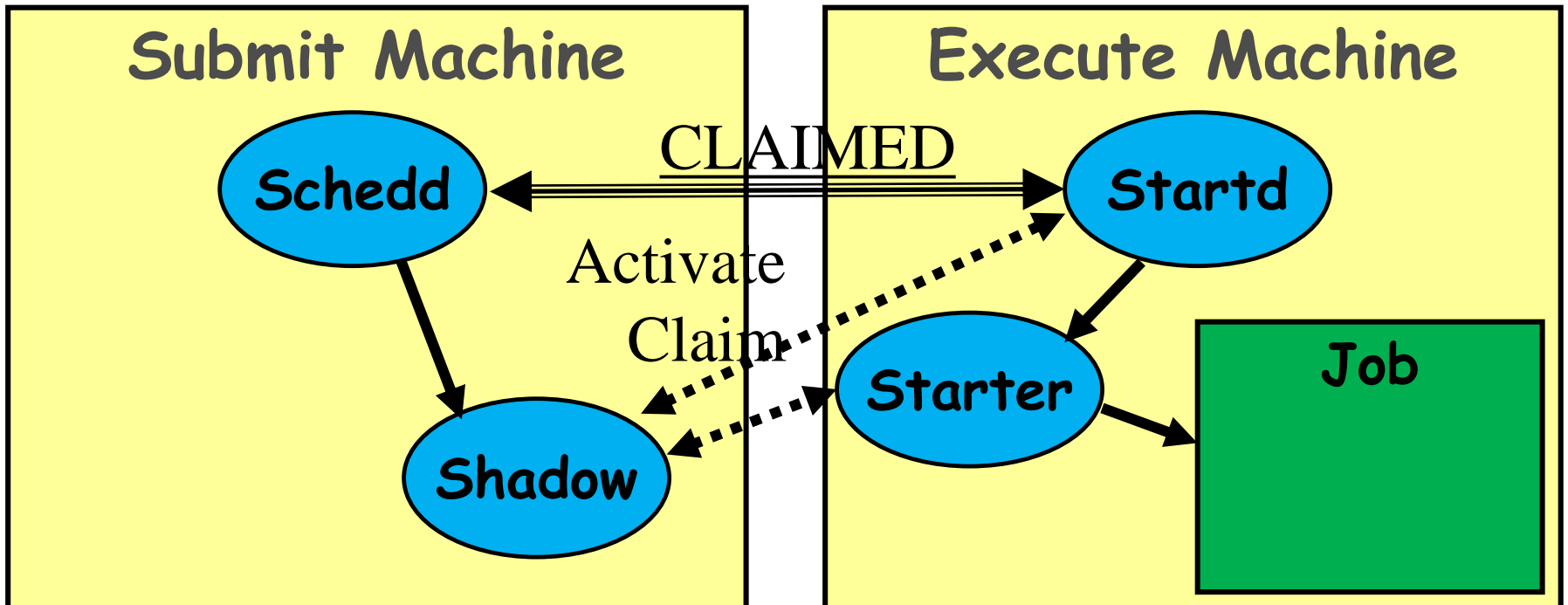
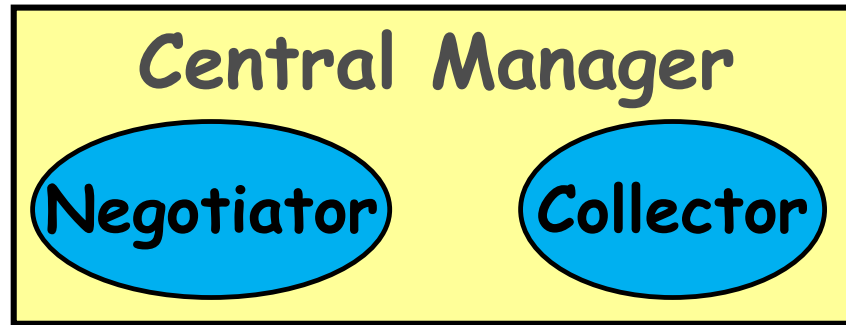
Claim Activation



Repeat until Claim released



Repeat until Claim released



When is claim released?

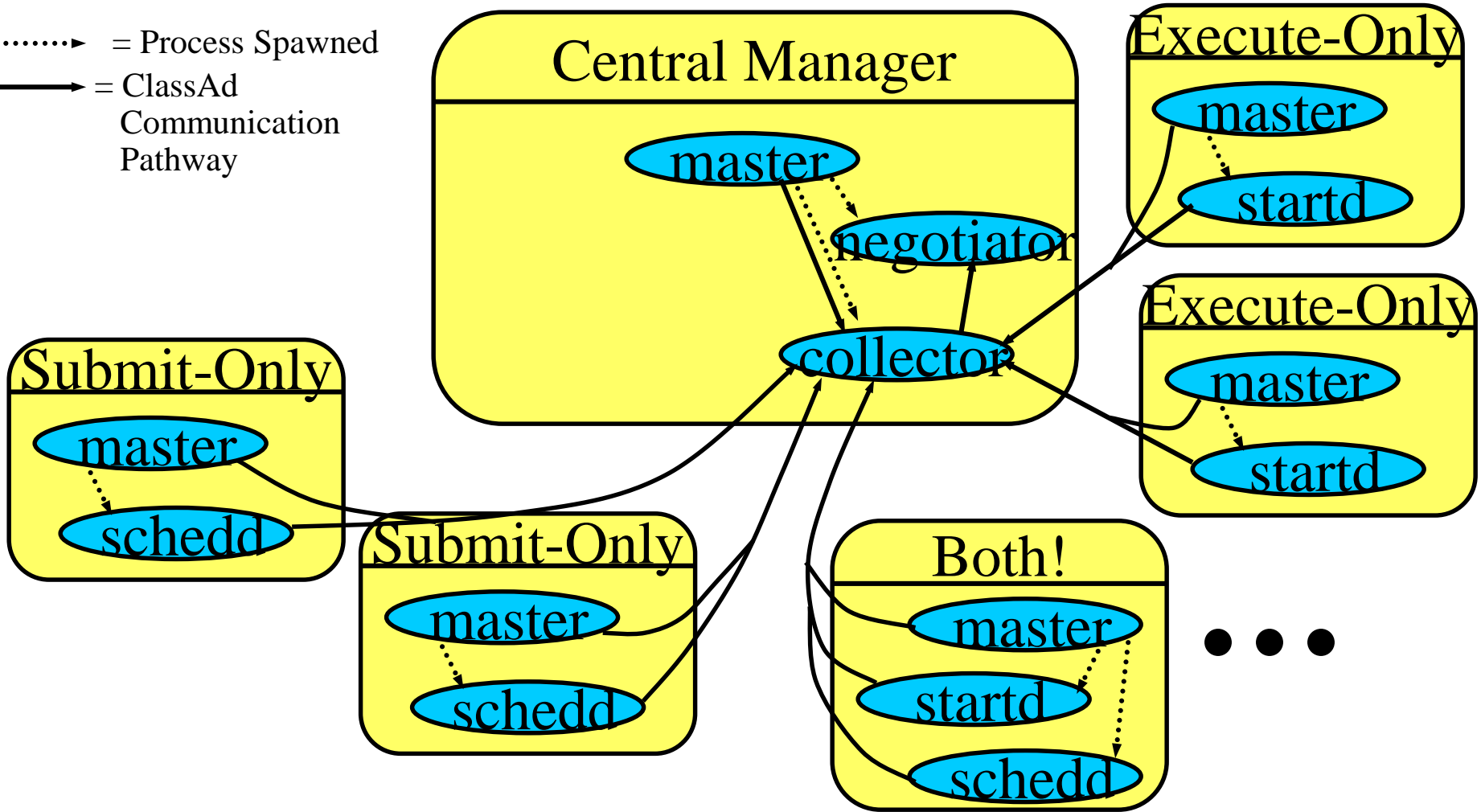
- › When relinquished by one of the following
 - lease on the claim is not renewed
 - Why? Machine powered off, disappeared, etc
 - schedd
 - Why? Out of jobs, shutting down, schedd didn't "like" the machine, etc
 - startd
 - Why? Policy re CLAIM_WORKLIFE, prefers a different match (via Rank), non-dedicated desktop, etc
 - negotiator
 - Why? User priority inversion policy
 - explicitly via a command-line tool
 - E.g. condor_vacate

Architecture items to note

- › Machines (startds) or submitters (schedds) can dynamically appear and disappear
 - Key for expanding a pool into clouds or grids
 - Key for backfilling HPC resources
- › Scheduling policy can be very flexible (custom attributes) and very distributed
- › Central manager just makes a match, then gets out of the way
- › Distributed policy enables federation across administrative domains
 - Lots of network arrows on previous slides
 - Reflects the P2P nature of HTCCondor

Layout of a General Condor Pool

.....▶ = Process Spawned
→ = ClassAd
Communication
Pathway



Thank You