PRAGMA/GLEON Expedition Technology overview

Renato Figueiredo
Associate Professor
Electrical and Computer Engineering
University of Florida

A bit of background

There’s also beauty in computing environments

Tools that enable great science

Complexity needs to be managed - collaboration

PRAGMA + GLEON

- Brings together computer science and lake ecology researchers and students
- Catalyze interactions between the two domains, focusing on scenario of complementary interest
- A people network *and* a computer network - common vocabulary, common resources, and trust
- Advancing the science within both disciplines

Our ‘expedition’ thus far

- Team has collaborated on the conceptualization of a computing infrastructure
  - Identify gaps early on, build shared knowledge
  - Iterate throughout implementation
  - Build it with them vs. build it and they will come
- Computing infrastructure itself lends to collaboration - sharing of resources, models, data

Expedition Goals

- A collaborative, distributed computing infrastructure capable of serving, in a sustainable manner, the increasing data processing demands of lake ecology scientists
- Understanding the opportunities and challenges in designing, implementing, and managing such infrastructures, and generalizing to other domains
- Building a common vocabulary to catalyze research synergies for both domains

Requirements

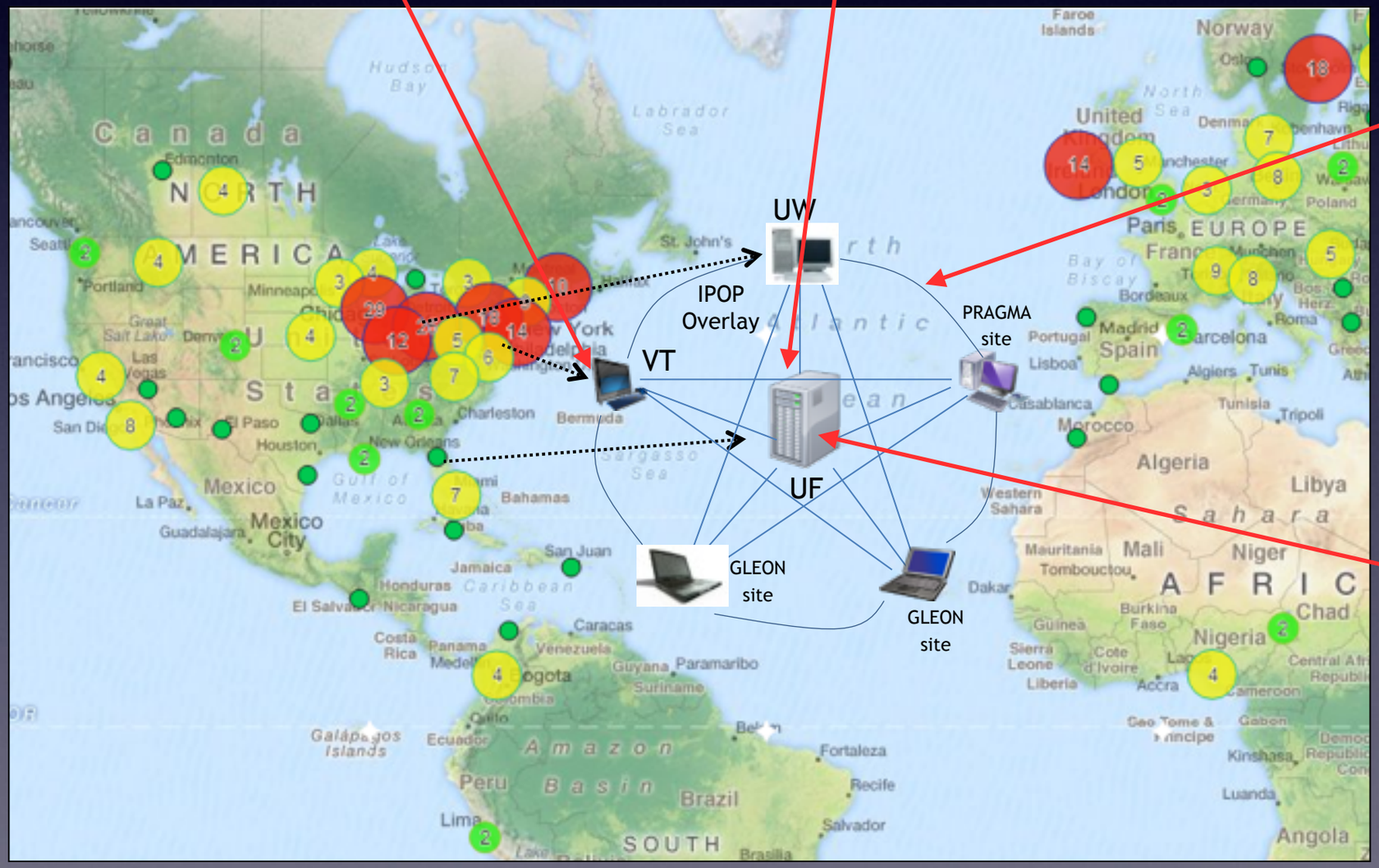
- Usability
  - Ease of use is crucial - but not easy to achieve
    - Meet halfway
- Capacity and scalabilty
  - More models, simulations, data: virtuous cycle
- Robustness
  - Build upon existing, proven systems

Technology overview

Desktops (near you)

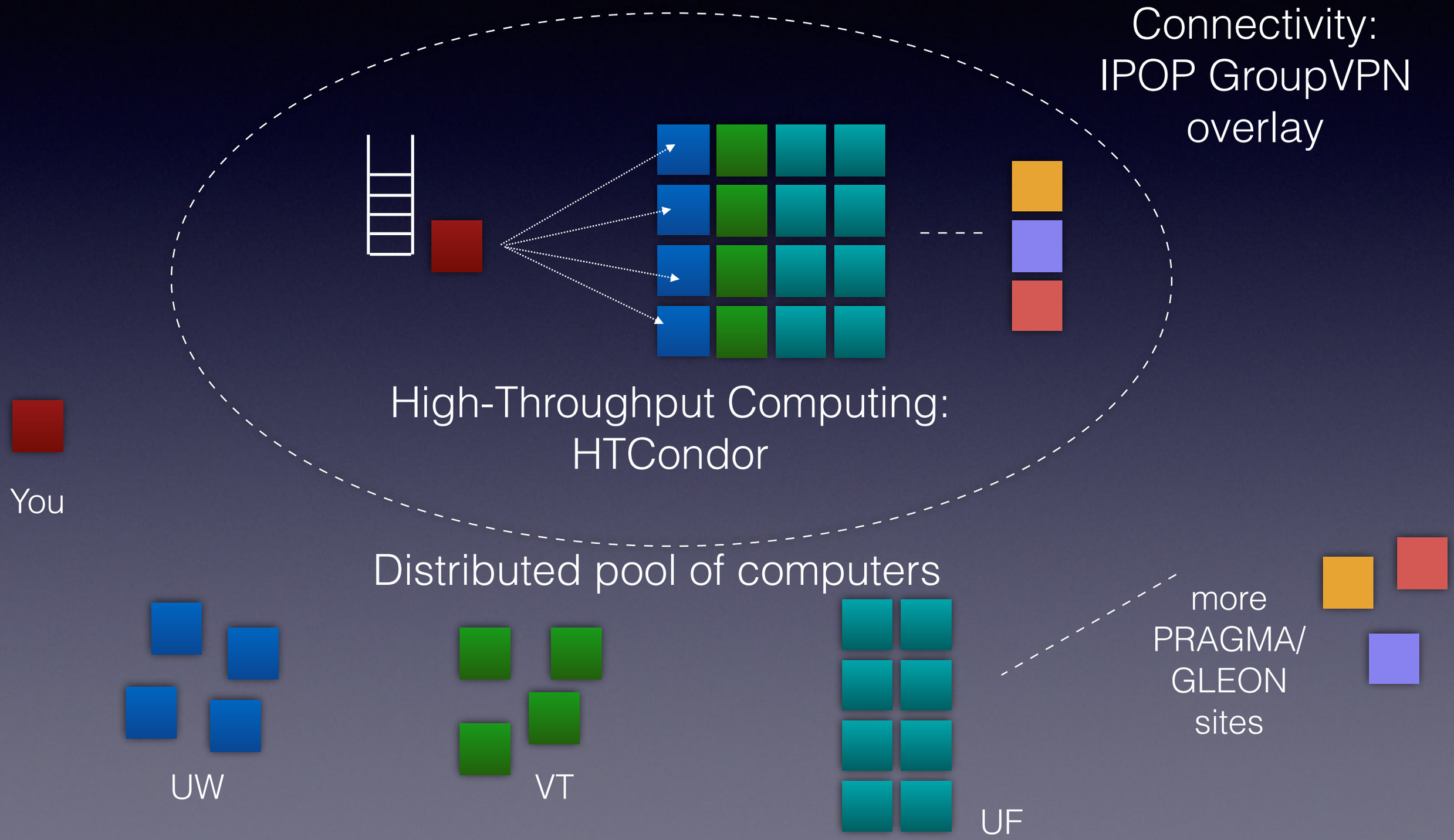
Servers (distributed)

Overlay software (IPOP)



Job scheduler software (HTCondor)

Overview


Connectivity:
IPOP GroupVPN
overlay
High-Throughput Computing:
HTCondor
You
Distributed pool of computers
UW
VT
UF
more
PRAGMA/
GLEON
sites

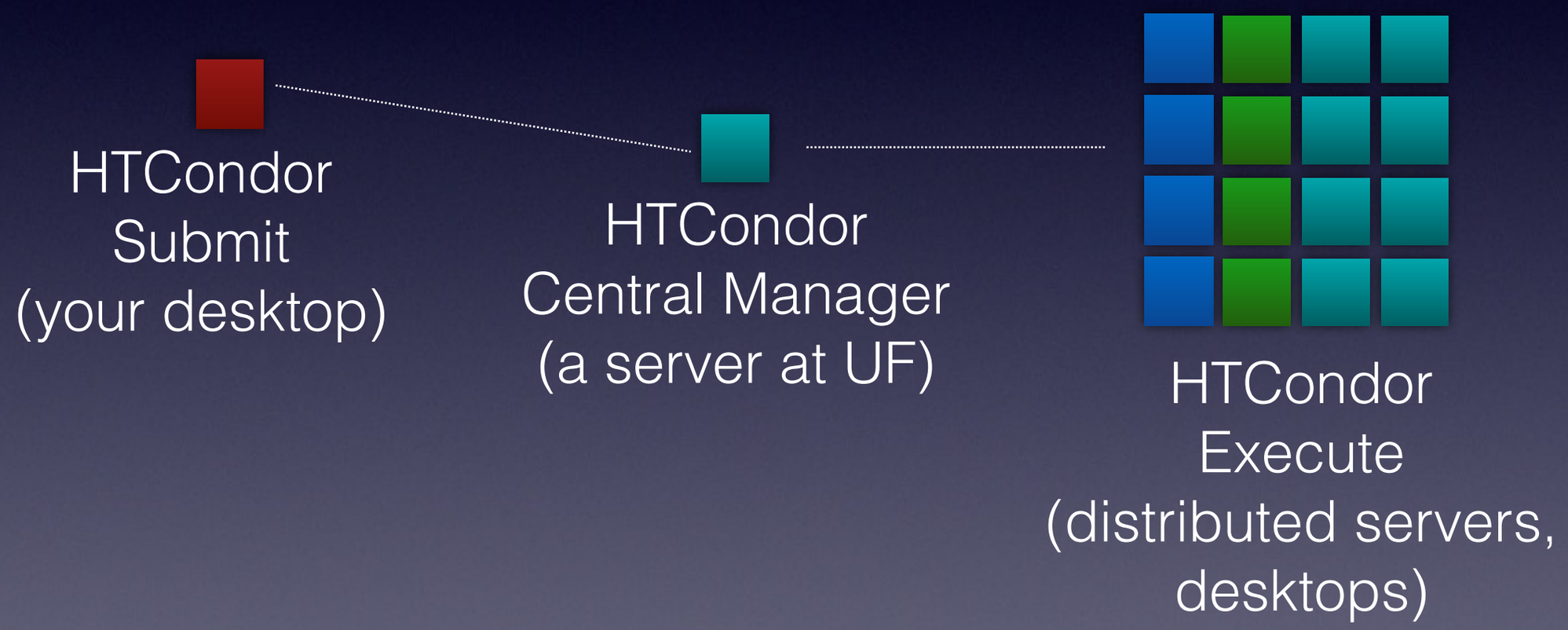
HTCondor: High-Throughput Computing

- Run large batches of independent, long-running simulation jobs across distributed computing resources
- HTC versus HPC/supercomputing
  - Support existing code (e.g. GLM model)
    - Parallelism: multiple jobs vs. parallel program
    - E.g. parameter sweeping, Monte Carlo simulations
  - Key metric: # of jobs completed per hour (day, …)

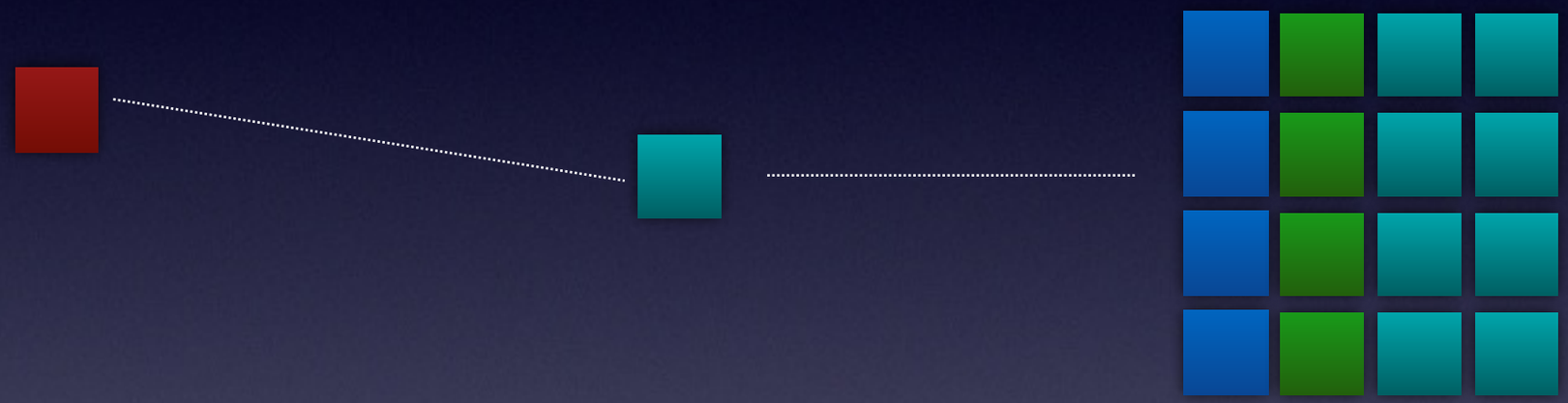
User Workflow

- Install IPOP+HTCondor software on your PC desktop(s)
- Configure IPOP to “join” the GroupVPN overlay
  - Gain access to number of resources
- Prepare and submit HTCondor job batches
  - Job: model’s executable, inputs, outputs
  - Batch: 10s, 100s, 1000s, … of independent jobs

HTCondor Workflow

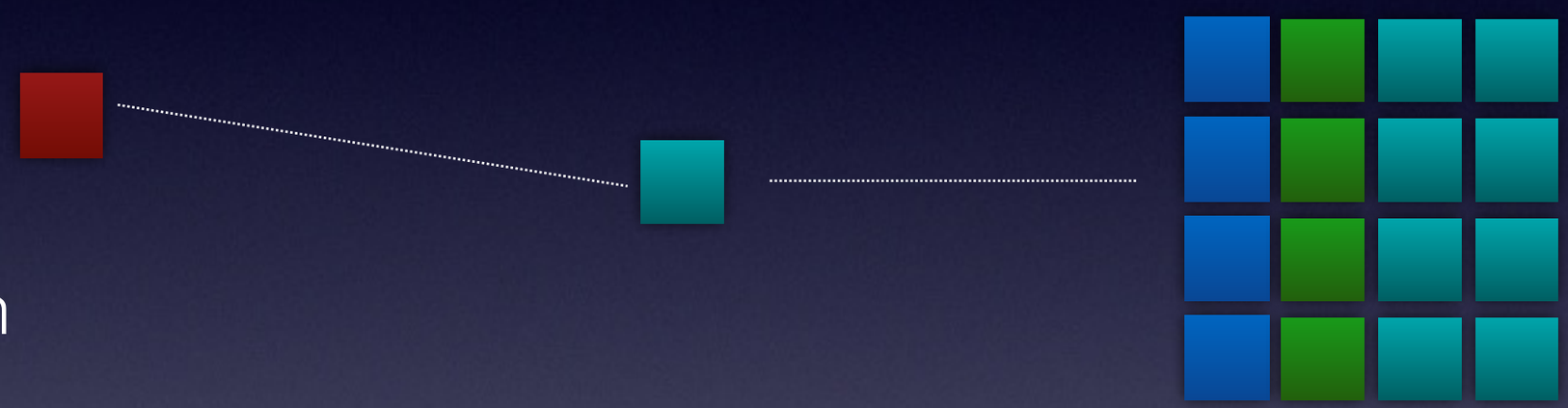


Prepare jobs



Files in your computer;
each folder
defines a job

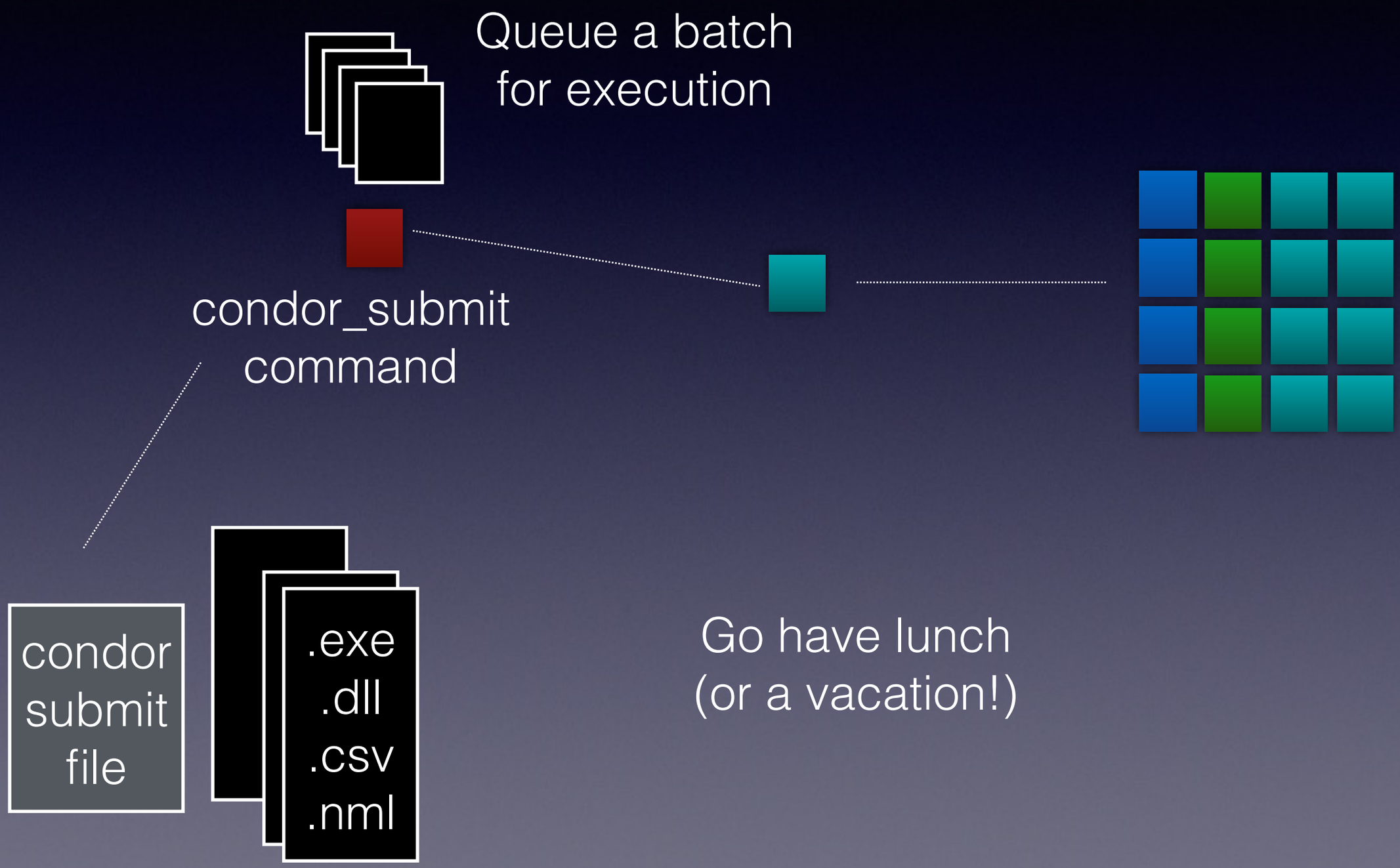
Prepare batch



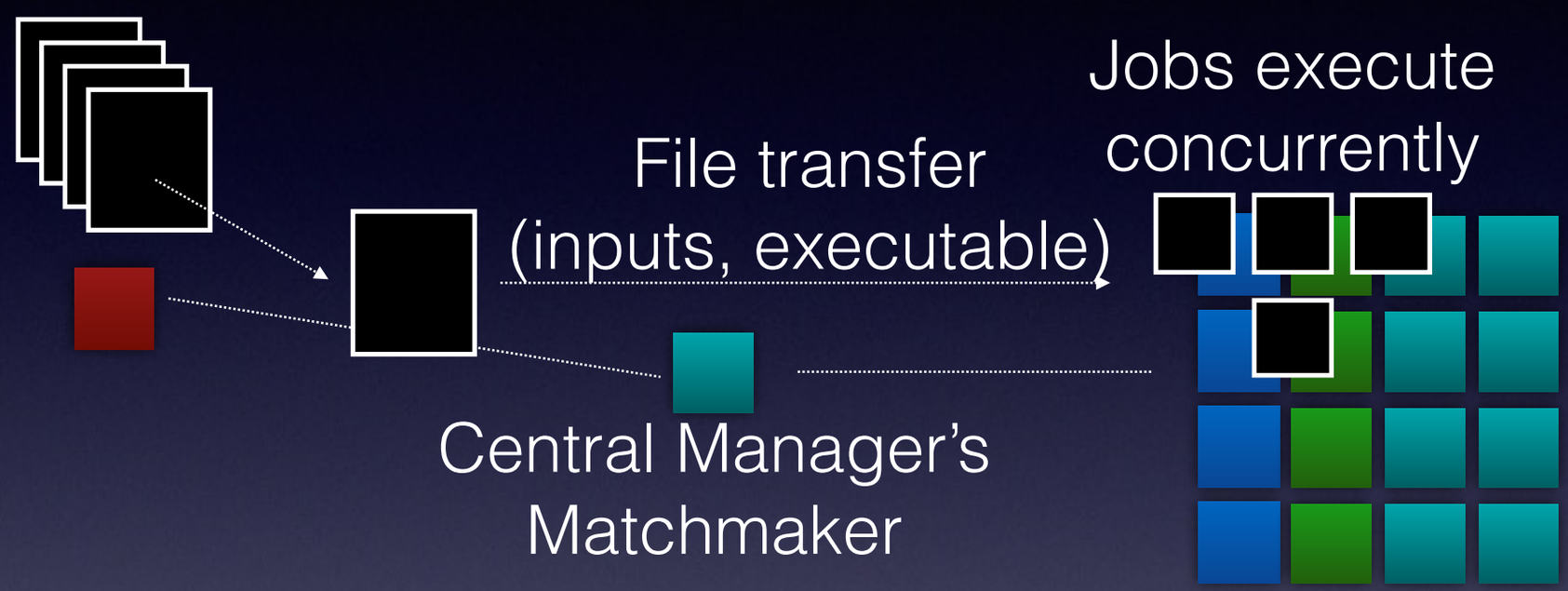
Configuration
file specifies
a batch of jobs

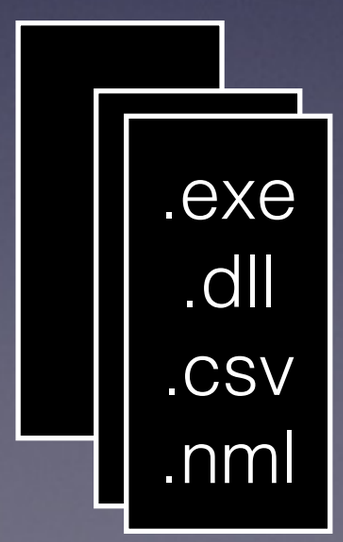
A key part of user experience
Collaboration between CS
and lake ecology teams
Goals: simplify and automate

Submit batch



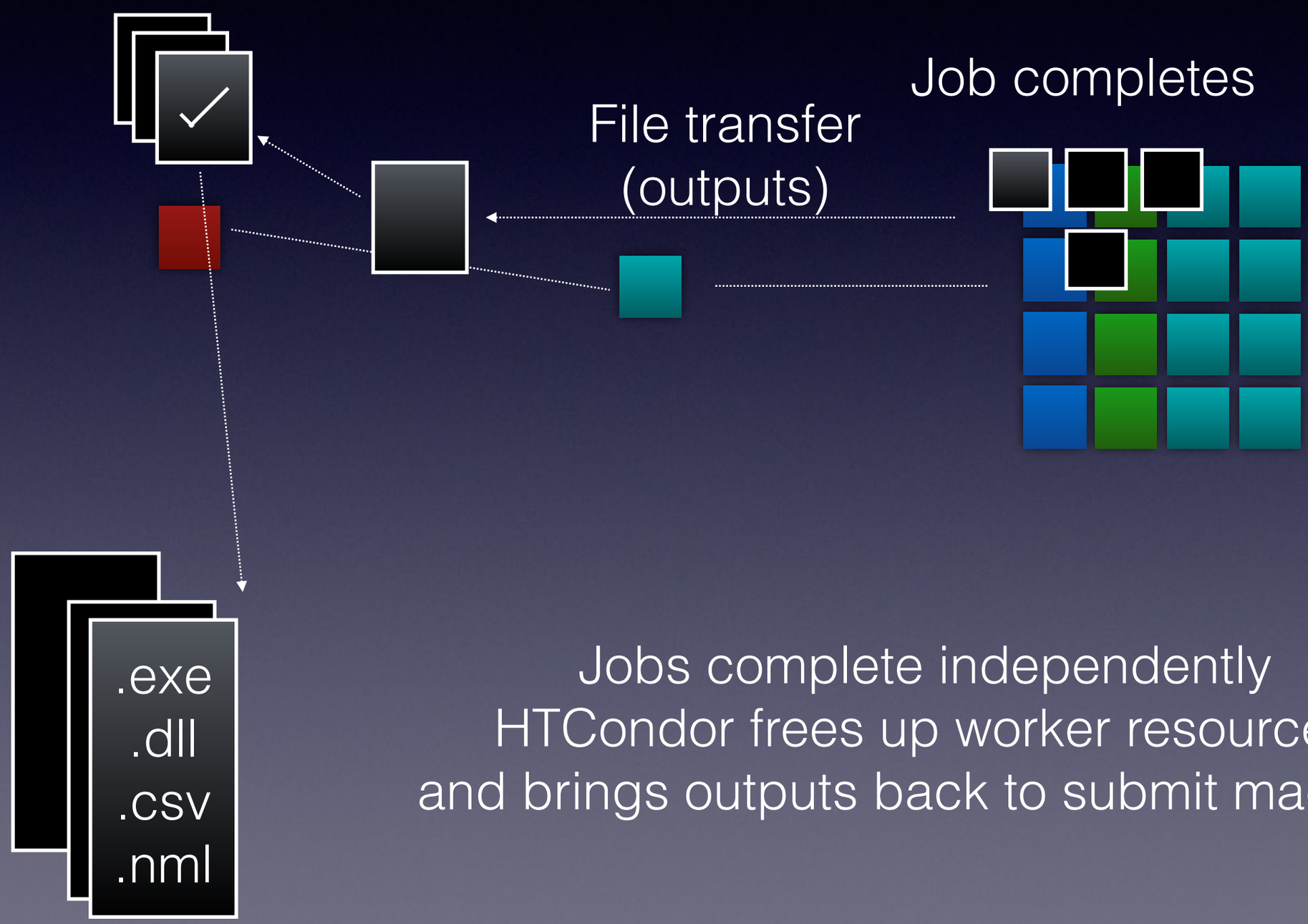
HTCondor at work





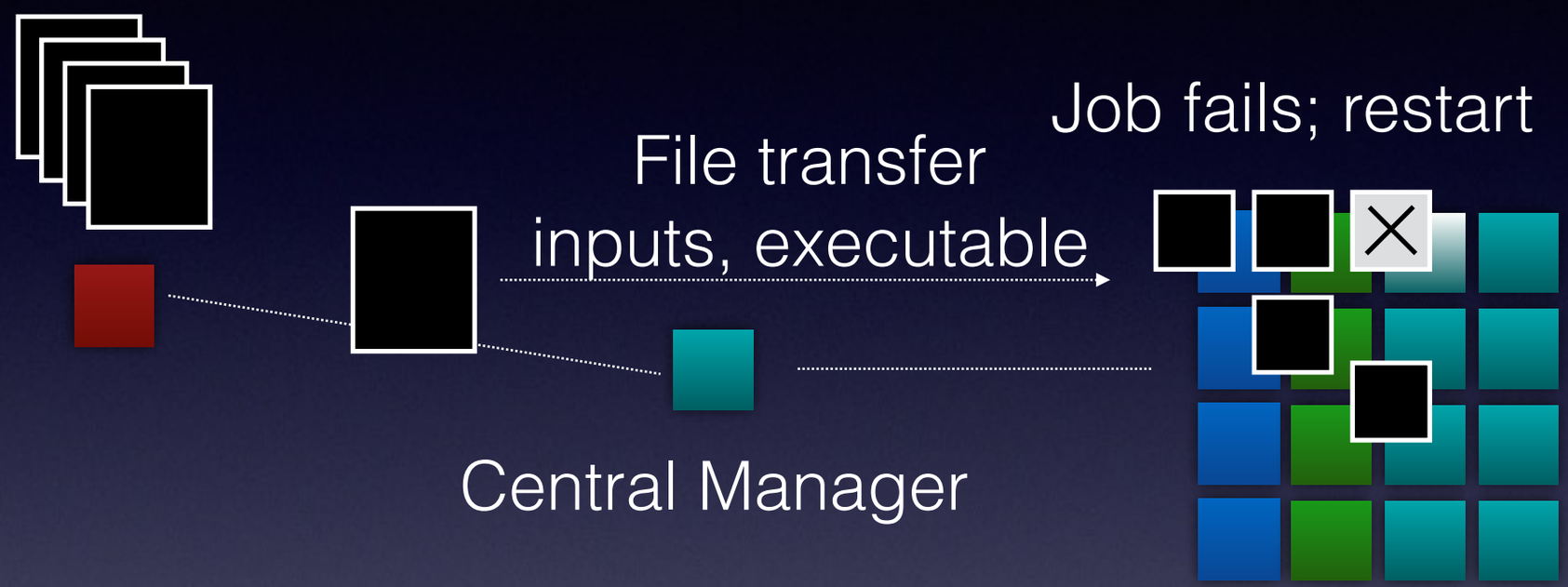
Jobs have requirements
Resources have capabilities and workload
Matchmaker - online dating for jobs+resources
Allows you to specify constraints
(e.g. "only run on my lab machines")

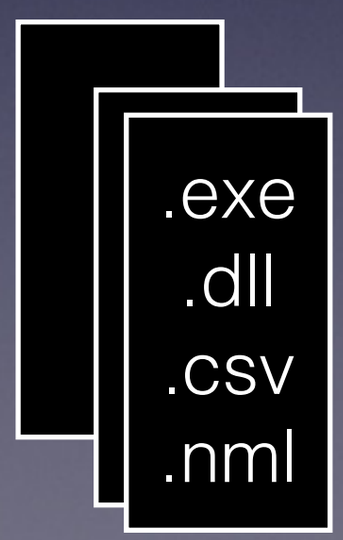
HTCondor at work



Jobs complete independently
HTCondor frees up worker resources
and brings outputs back to submit machine

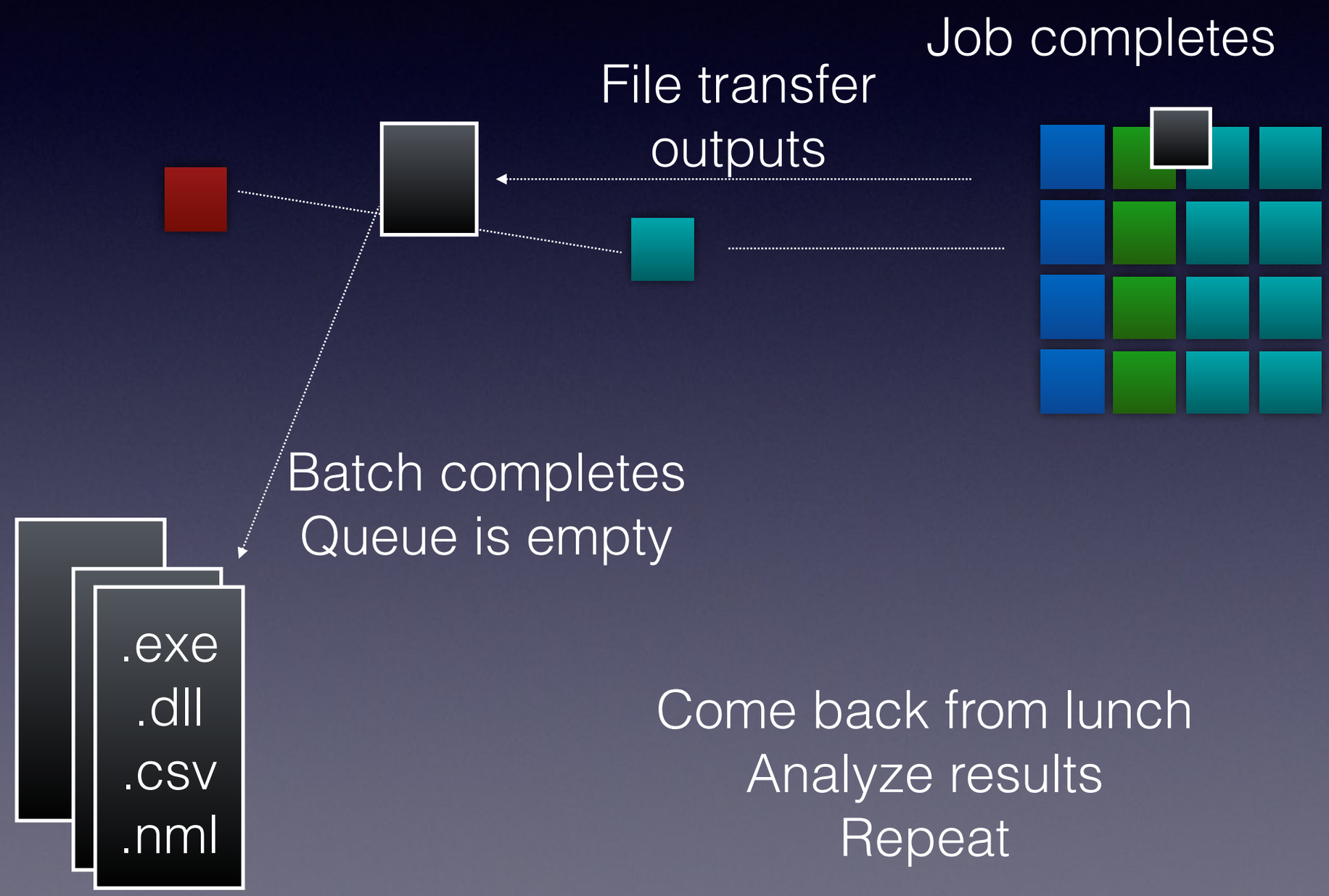
HTCondor at work





HTCondor deals with failures so you don't have to
When failure is detected, job is automatically restarted

HTCondor at work

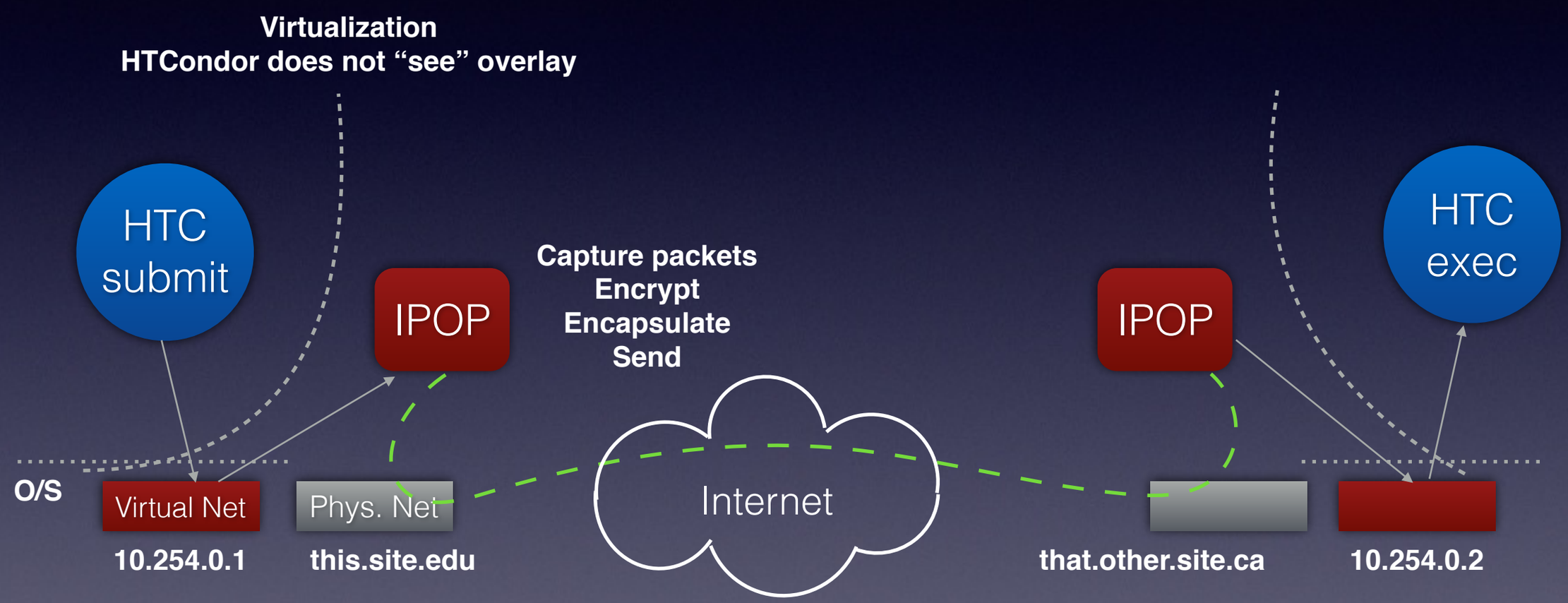


Come back from lunch
Analyze results
Repeat

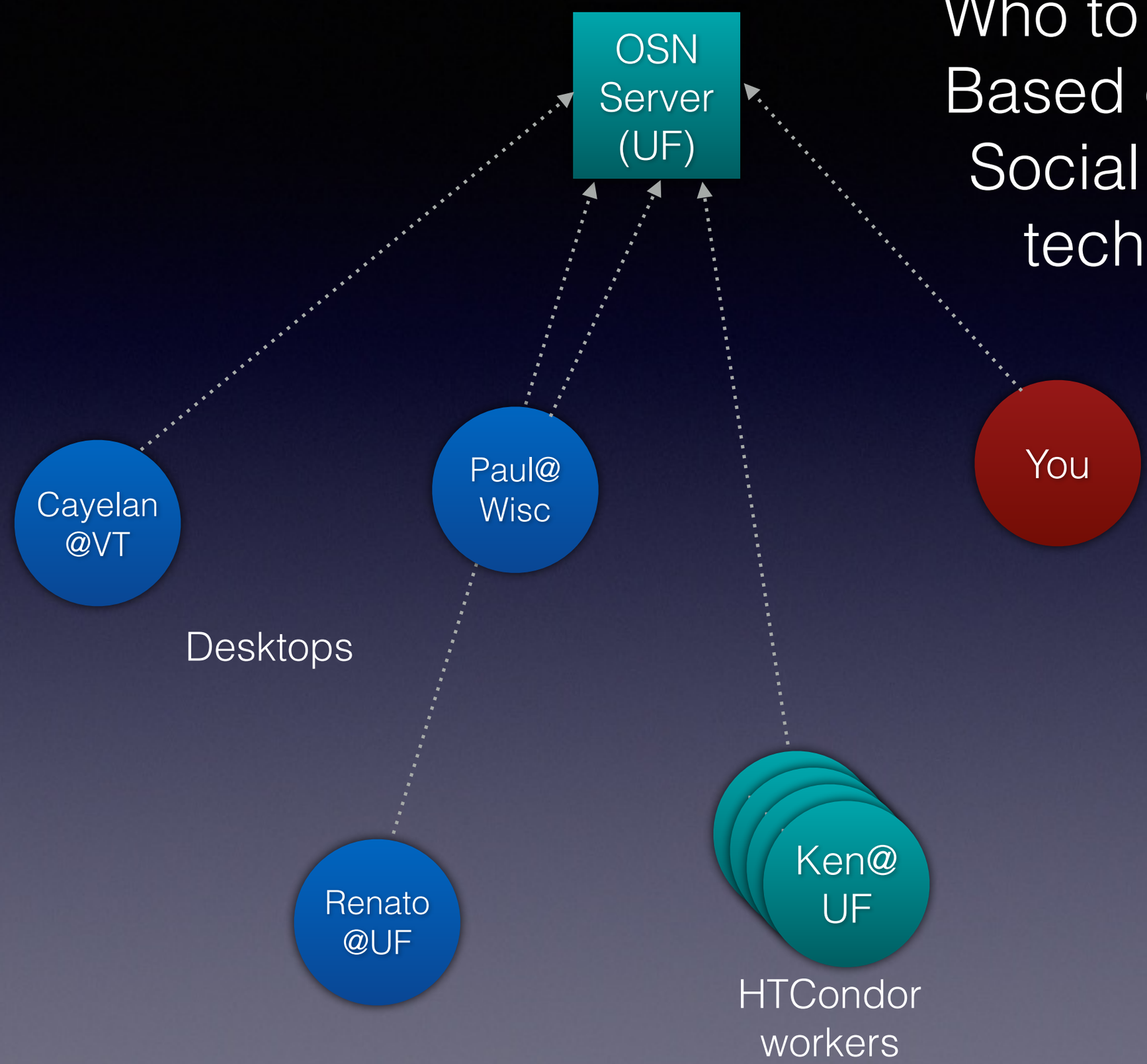
IPOP: Overlay Network

- Resources that HTCondor needs are distributed across multiple GLEON/PRAGMA institutions
  - Challenges: firewalls, Internet addresses
- Overlay network brings resources together
  - Under the same umbrella
  - Virtual private network (GroupVPN) across collaborators
- Enables HTCondor to work as if in a single institution

IPOP: Overlay software



Who to connect? Based on Online Social Network techniques



Workflow Revisited

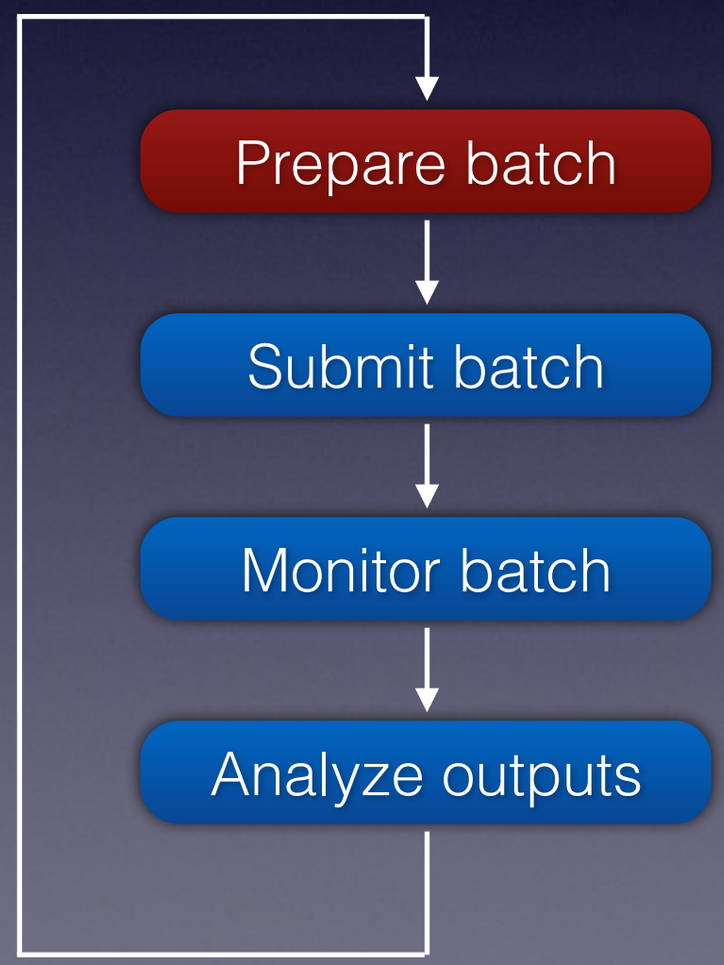
- Download and install HTCondor software
- Download and install IPOP overlay software
- Request to join the “social network”
- Prepare job batches
- Compute away!
- https://github.com/GRAPLE/documentation/wiki

Demo - outline

- Look and feel of the user interface
- HTCondor commands, behavior, terminology
- Practical example:
  - GLM/Mendota simulations
  - Efforts towards simplifying user interaction
- Give us your input so we can improve!

Workflow

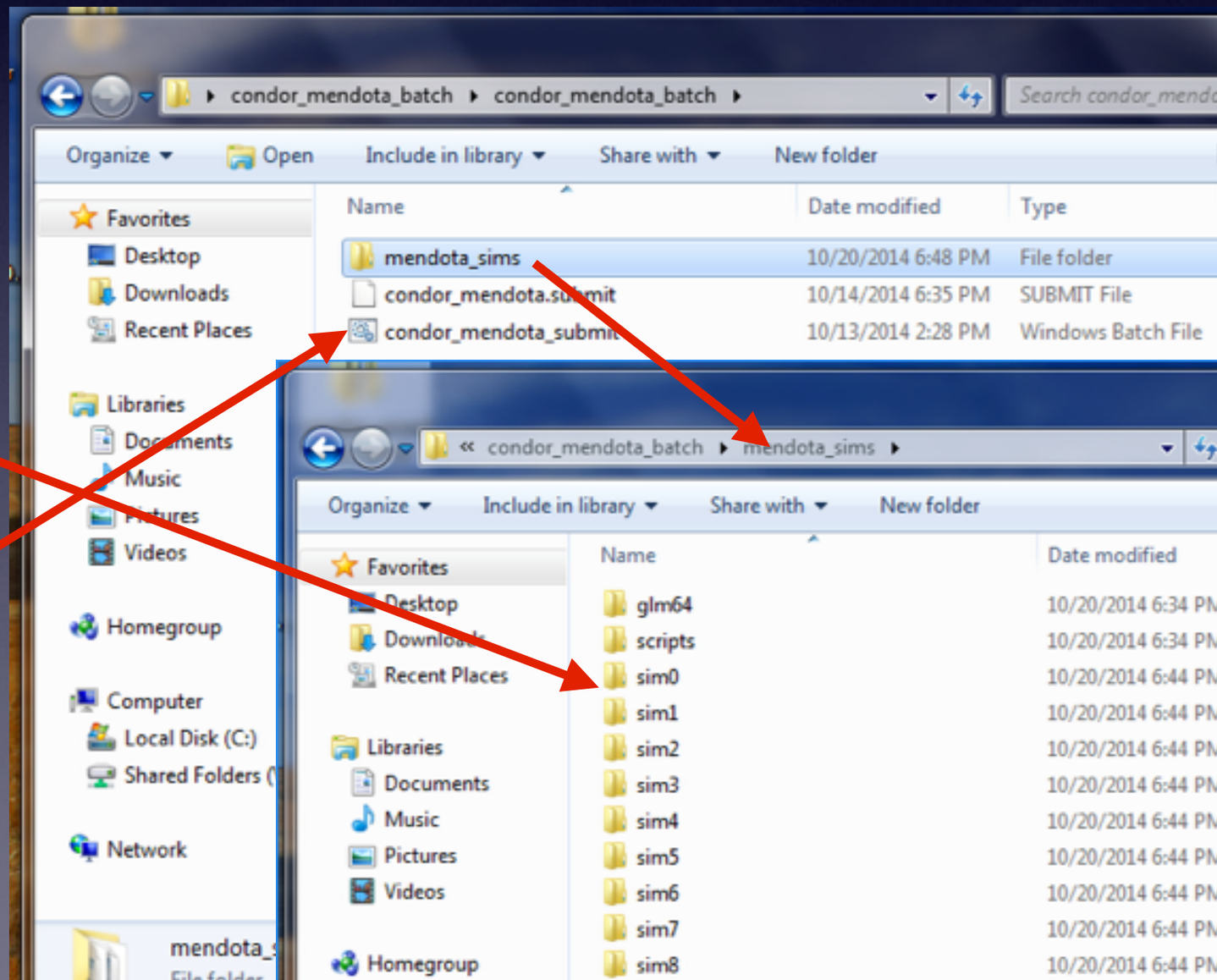
- Will focus on the “lifecycle” of a batch of jobs
  - Assuming IPOP/HTCondor software installed



Inputs, executable, libraries
Folders

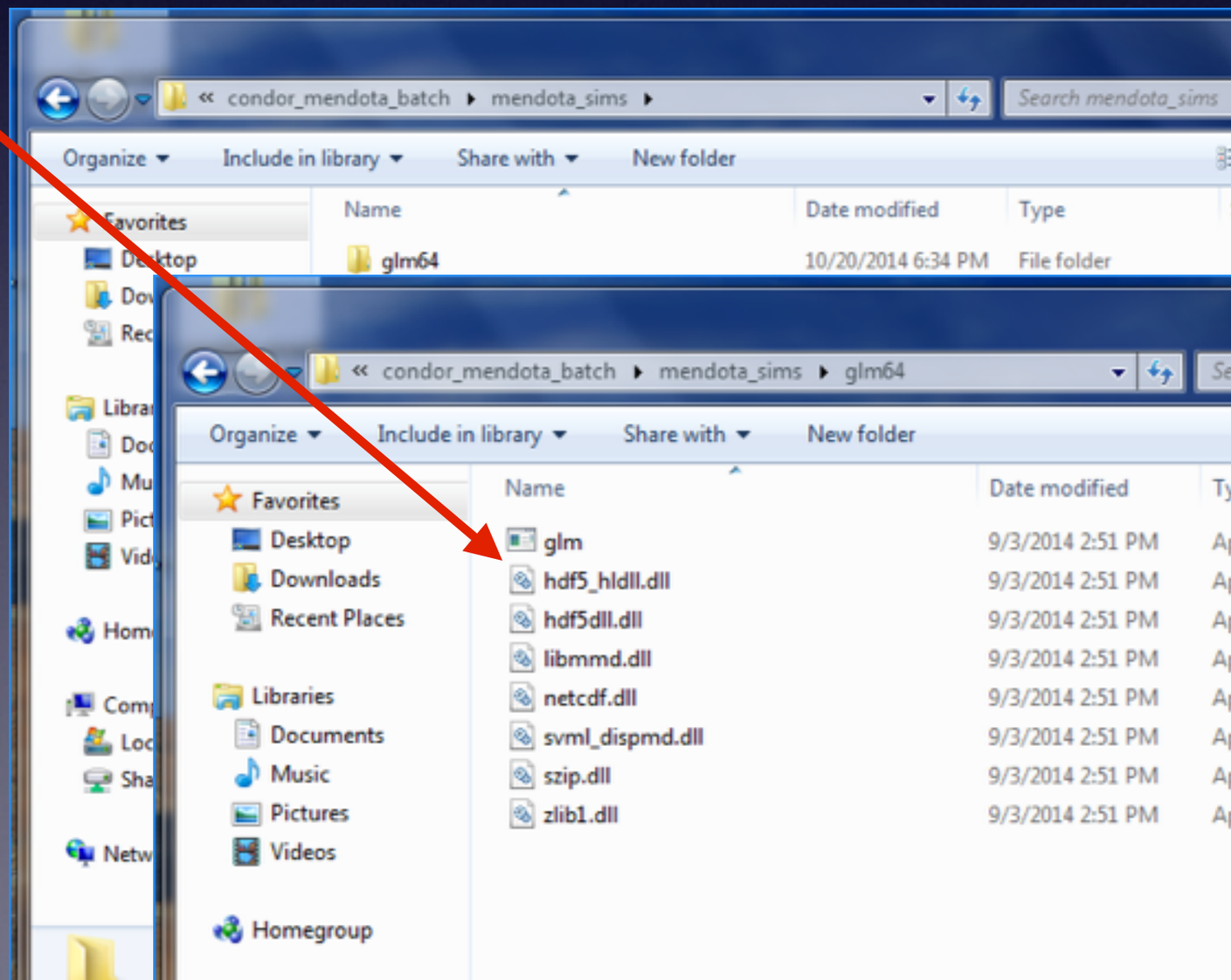
Preparing batch

- Folders in your computer
  - Executables (.exe, .dll)
  - Input files (.nml, .csv)
  - Each folder is a job
- Condor submit file
  - Specifies batch



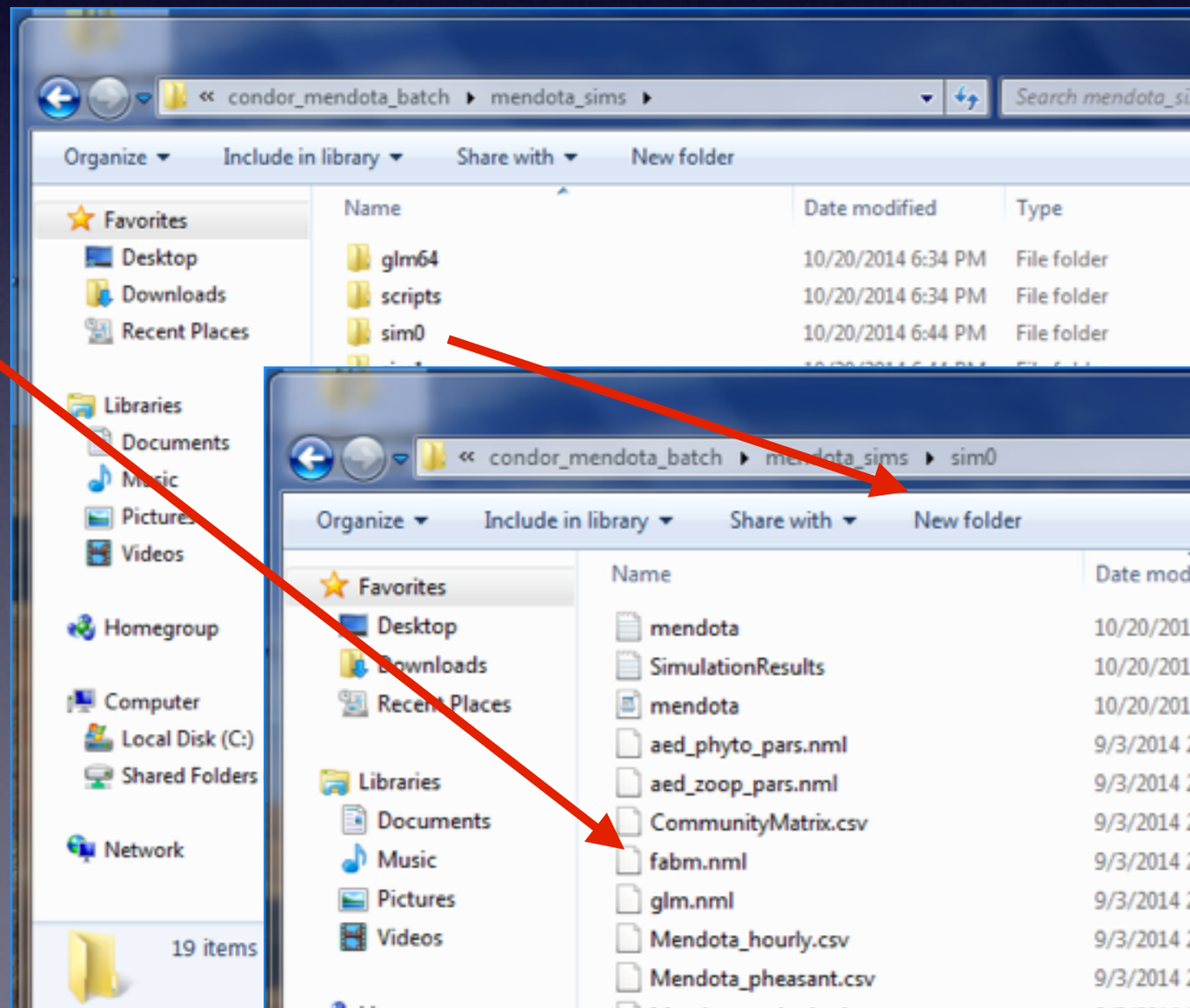
Preparing batch

- Folders in your computer
  - Executables (.exe, .dll)
  - Input files (.nml, .csv)
  - Each folder is a job
- Condor submit file
  - Specifies batch



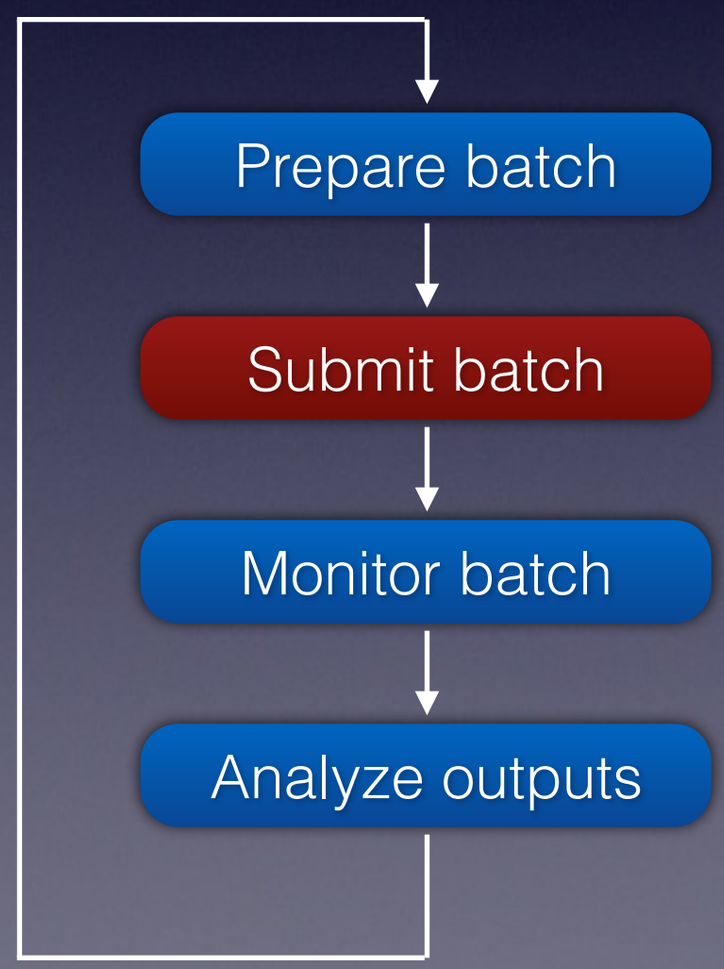
Preparing batch

- Folders in your computer
  - Executables (.exe, .dll)
  - Input files (.nml, .csv)
  - Each folder is a job
- Condor submit file
  - Specifies batch



Workflow

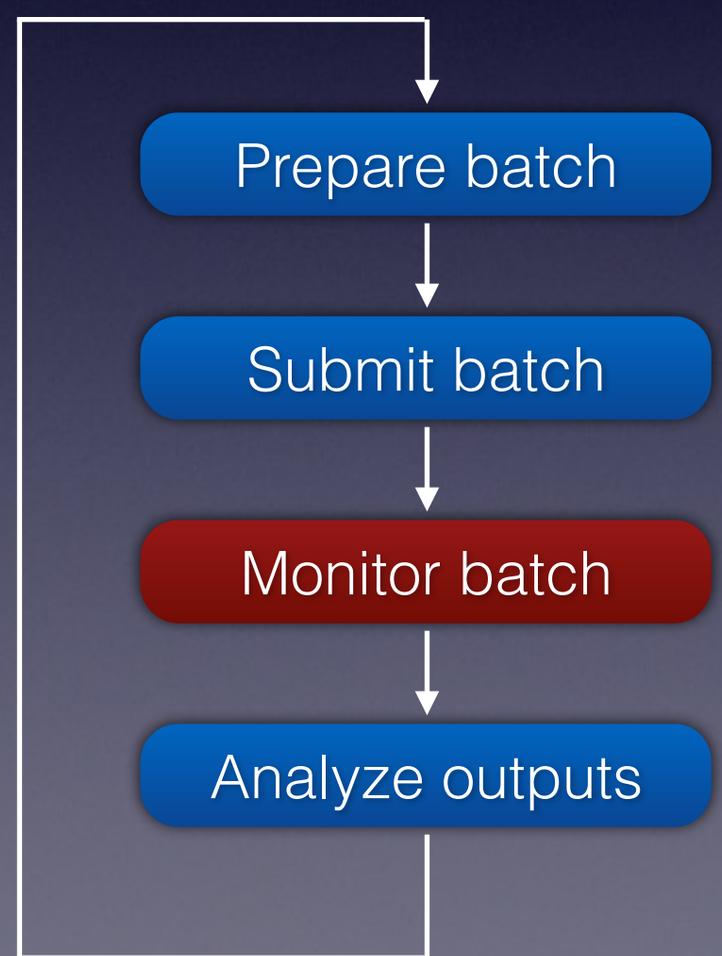
- Will focus on the “lifecycle” of a batch of jobs
  - Assuming IPOP/HTCondor software installed



HTCondor submit file
condor_submit

Workflow

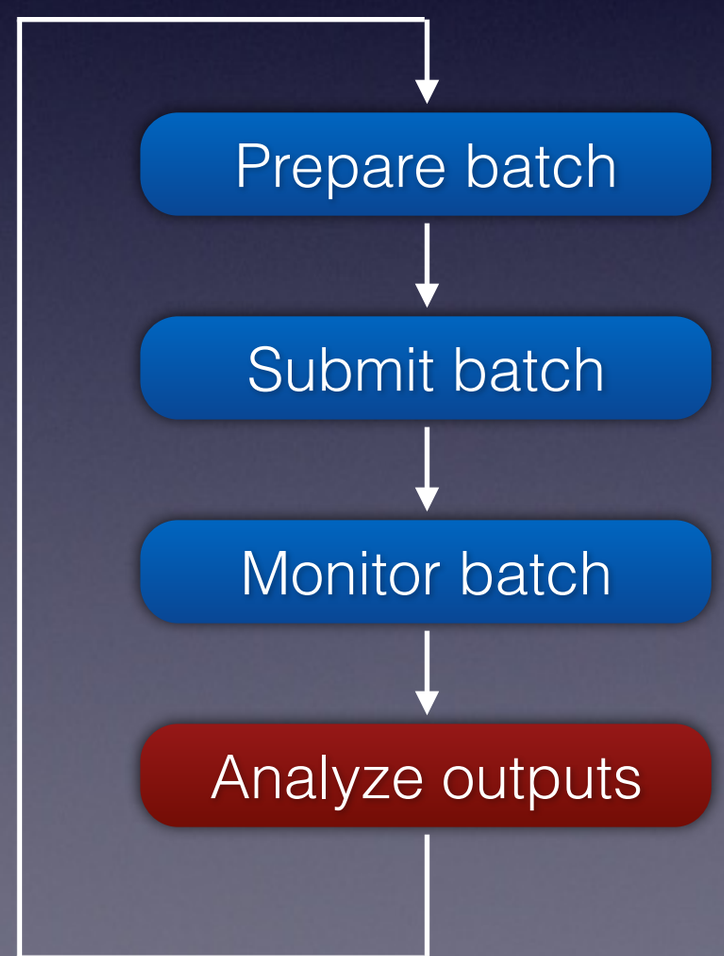
- Will focus on the “lifecycle” of a batch of jobs
  - Assuming IPOP/HTCondor software installed



condor_status
condor_q

Workflow

- Will focus on the “lifecycle” of a batch of jobs
  - Assuming IPOP/HTCondor software installed



R, Excel, …

Acknowledgments

- This material is based upon work supported in part by the National Science Foundation under Grants No. 1339737 (SI2-SSE) and 1234983 (SAVI-PRAGMA)