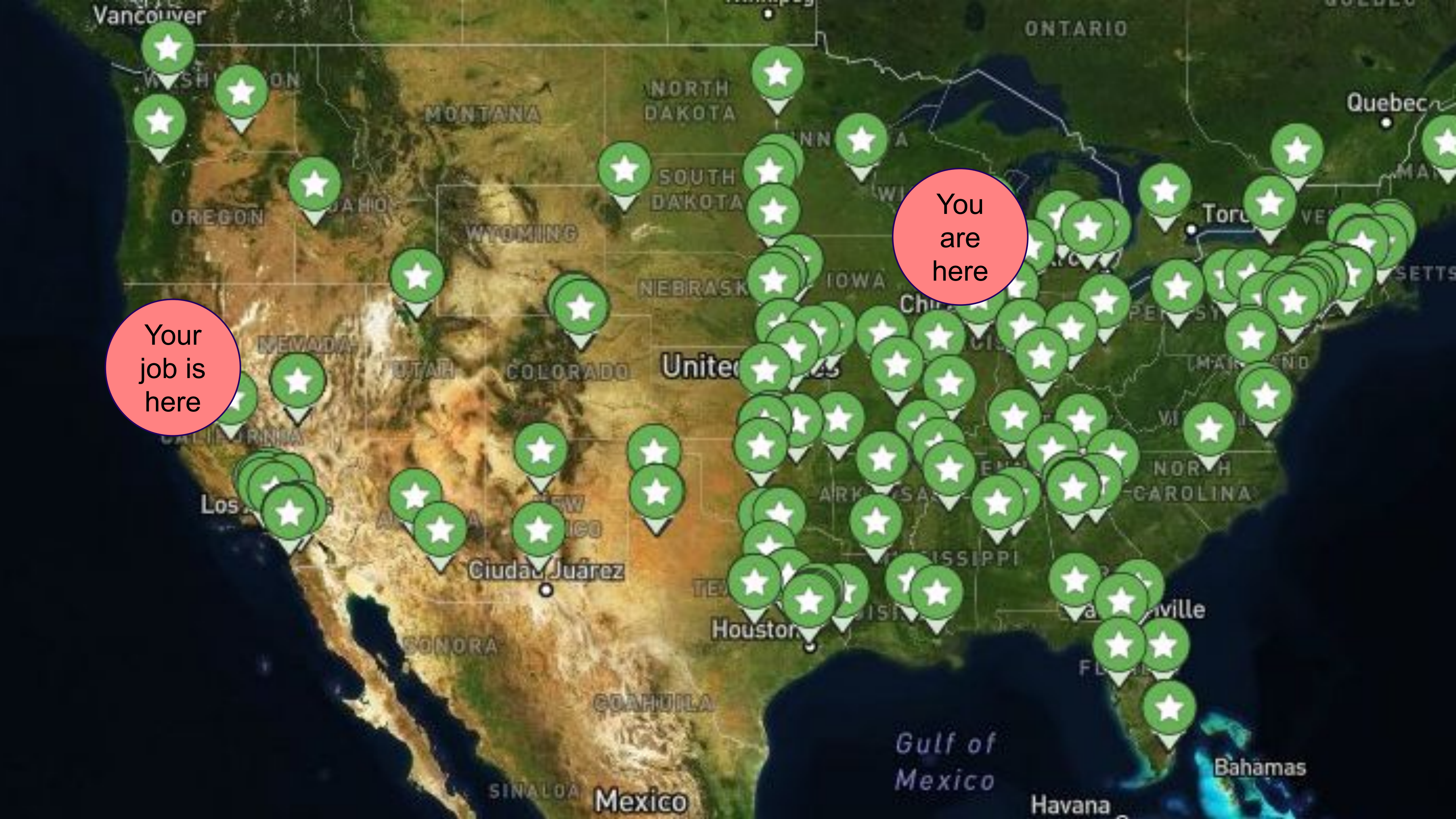
OSG

Handling Data on OSG

Wednesday, August 9

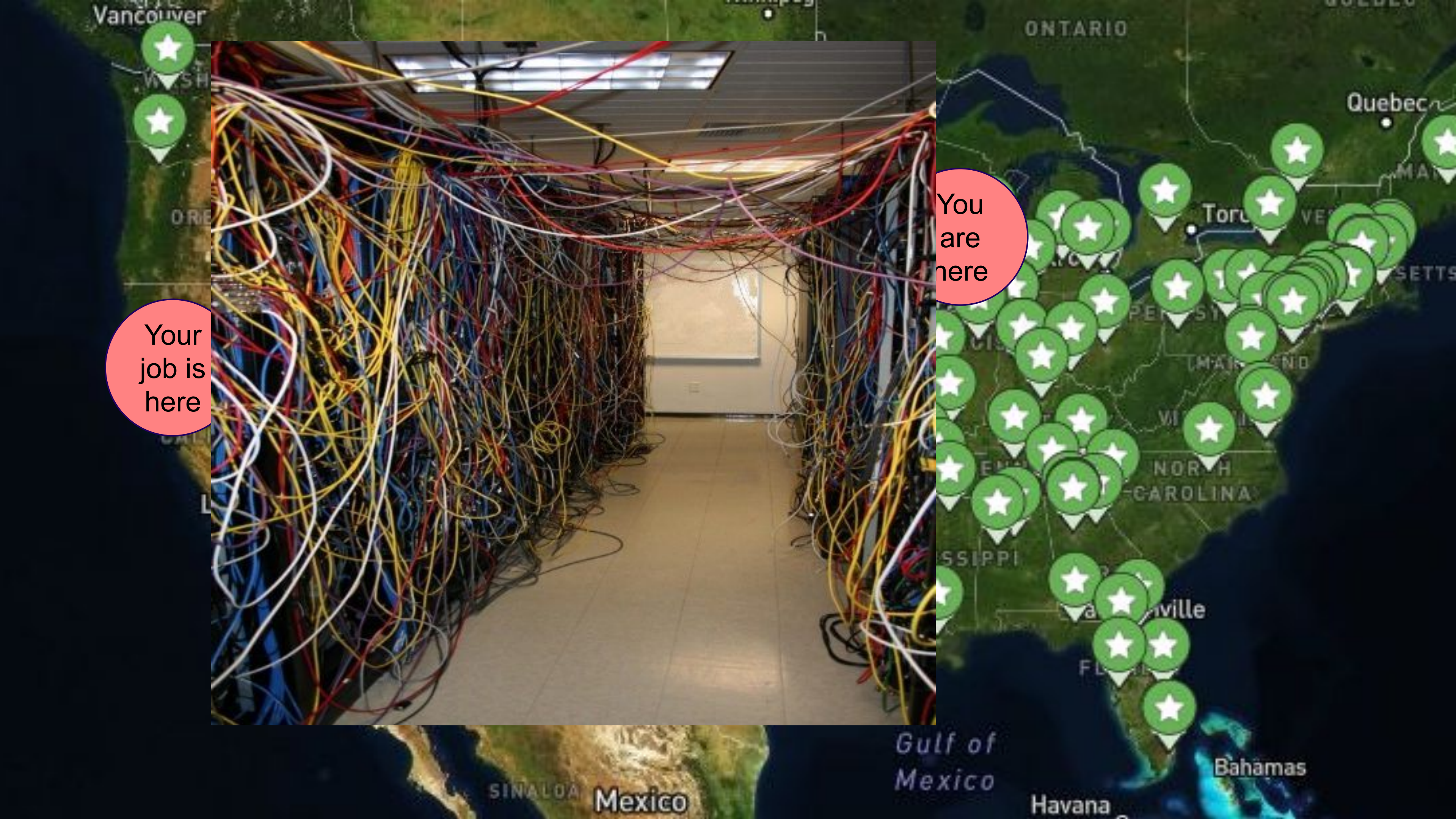
Mats Rynge

This work was supported by NSF grants MPS-1148698, OAC-1836650, and OAC-2030508



You are here

Your job is here



Your job is here

You are here

From yesterday…

```
container_image = py-cowsay.sif
```

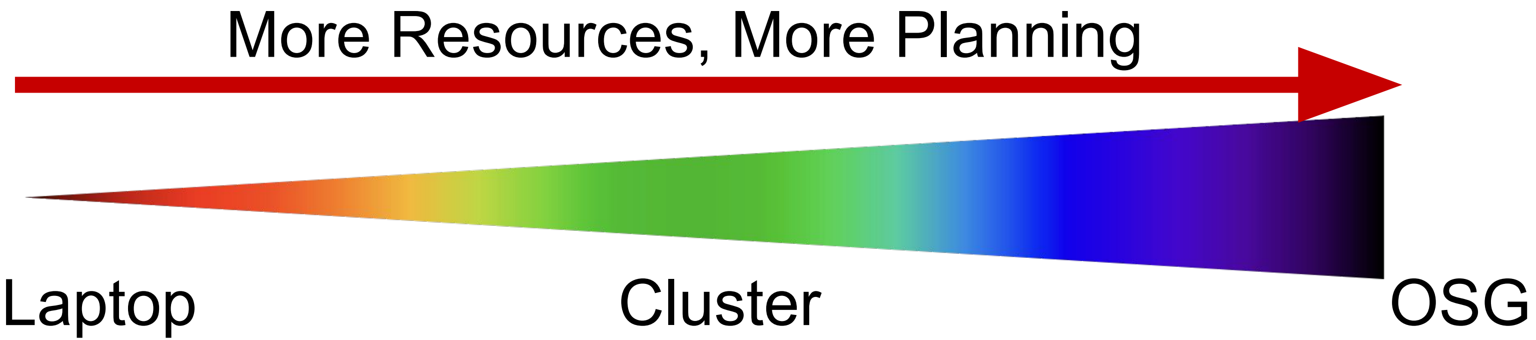
From yesterday…

```
container_image = py-cowsay.sif

queue 10000
```

Like all things

- I always think of HTC/OSG usage as a spectrum:



More Resources, More Planning

Laptop

Cluster

OSG

Handling Data on OSG

- Overview / Things to Consider
- HTCondor File Transfer
- OSDF
- Shared File Systems

What is ~~big~~ large data?

- In reality, “big data” is relative
  - What is ‘big’ for *you*? Why?

What is ~~big~~ large data?

- In reality, “big data” is relative
  - What is ‘big’ for *you*? Why?
- Volume, velocity, variety!
  - think: a million 1-KB files, versus one 1-TB file

Determining In-Job Needs

- “**Input**” includes *any* files needed for the job to run
  - `executable`
  - `transfer_input_files`
  - data ***and*** <u>software</u>

- “**Output**” includes any files produced for the job that *need to come back*
  - `output, error`

Data Management Tips

1. Determine your per-job needs
    a. minimize per-job data needs
2. Determine your batch needs
3. Leverage HTCondor and OSG data handling features!

First! Try to minimize your data

- split large input for better throughput
- eliminate unnecessary data
- file compression and consolidation
  - job input: prior to job submission
  - job output: prior to end of job
  - moving data between your laptop and the submit server

‘Large’ data: The collaborator analogy

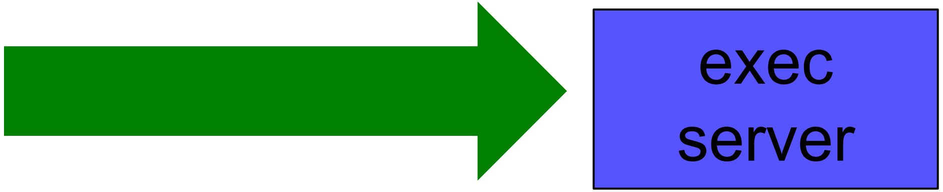
What method would you use to send data to a collaborator?

amount	method of delivery
words	email body
tiny – 100MB	email attachment (managed transfer)
100MB – GBs	download from Google Drive, Drop/Box, other web-accessible repository
TBs	ship an external drive (local copy needed)

Never underestimate the bandwidth of a station wagon full of tapes hurtling down the highway.

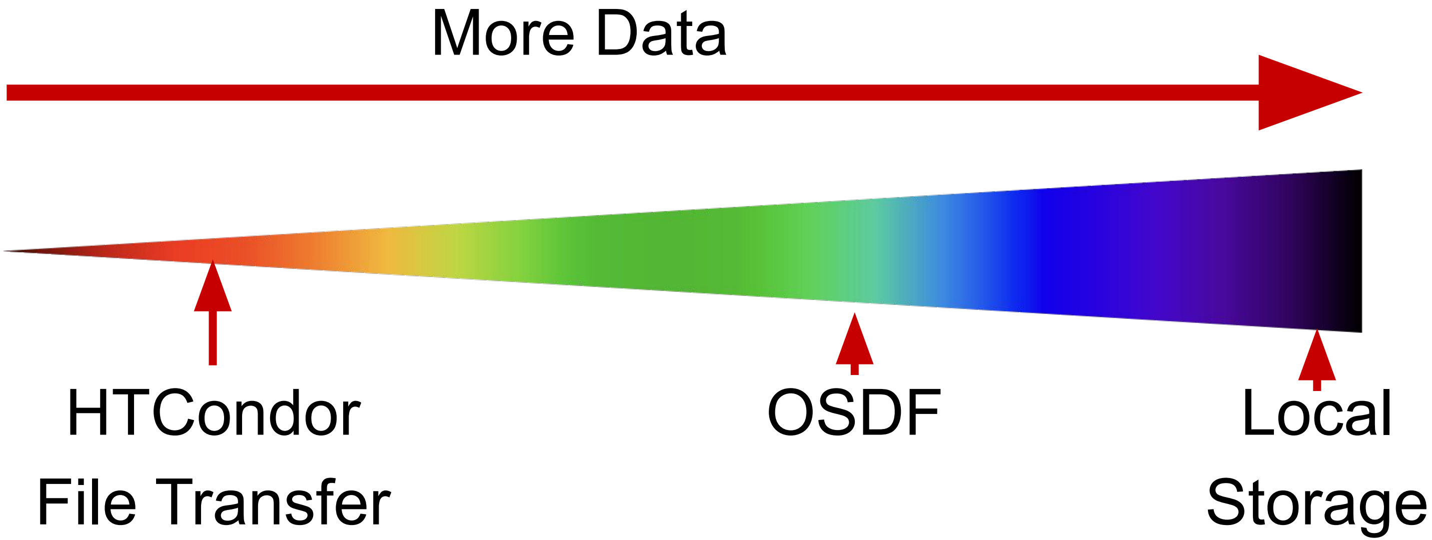
Andrew S. Tanenbaum (1981) – Professor Emeritus, Vrije Universiteit Amsterdam

Large *input* in HTC and OSG



file size	method of delivery
words	within executable or arguments?
tiny – 1GB per file	HTCondor file transfer (up to 1GB total per job)
1GB – 20GB	OSDF (regional replication)
20 GB – TBs	shared file system (local copy, local execute servers)

Transfers



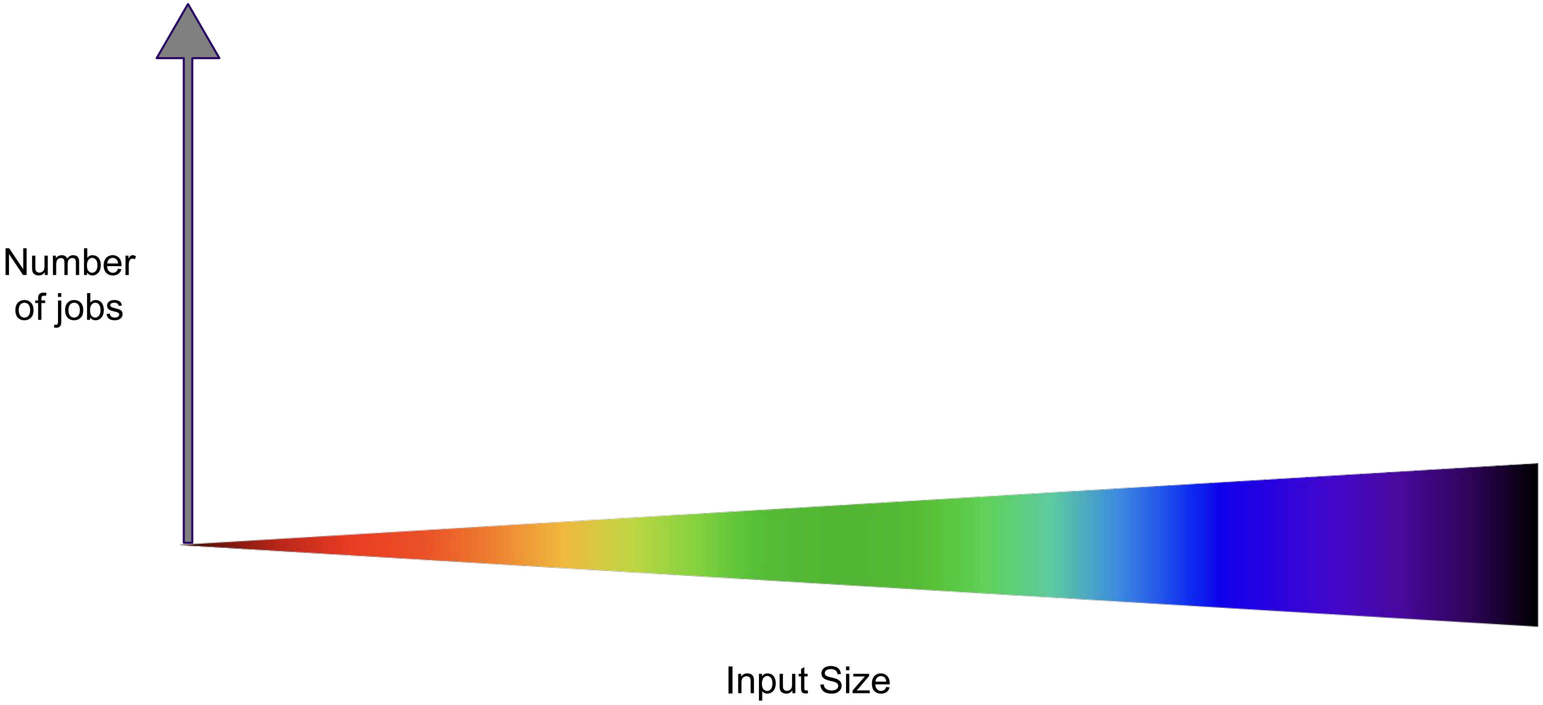
More Data

HTCondor File Transfer

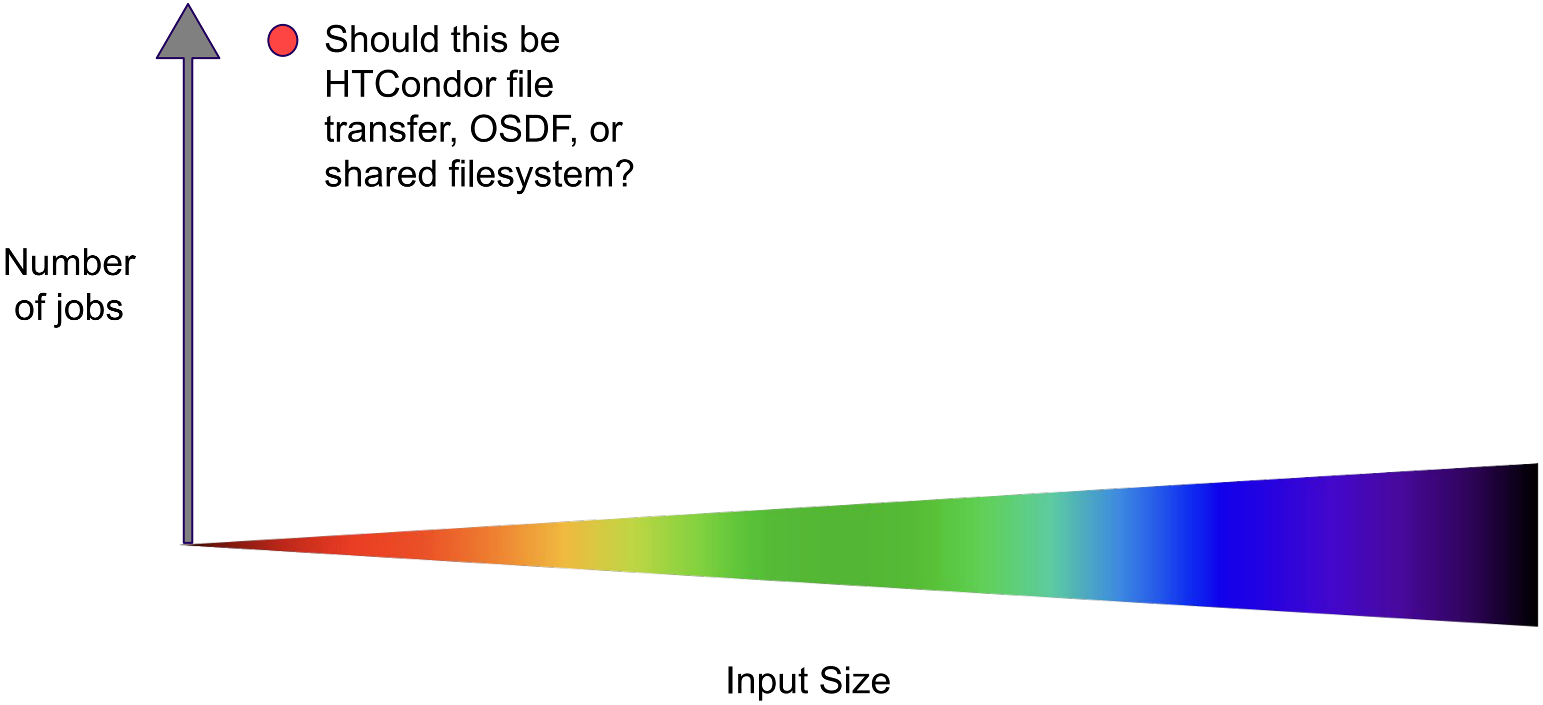
OSDF

Local Storage

Rule of thumb - many dimensions

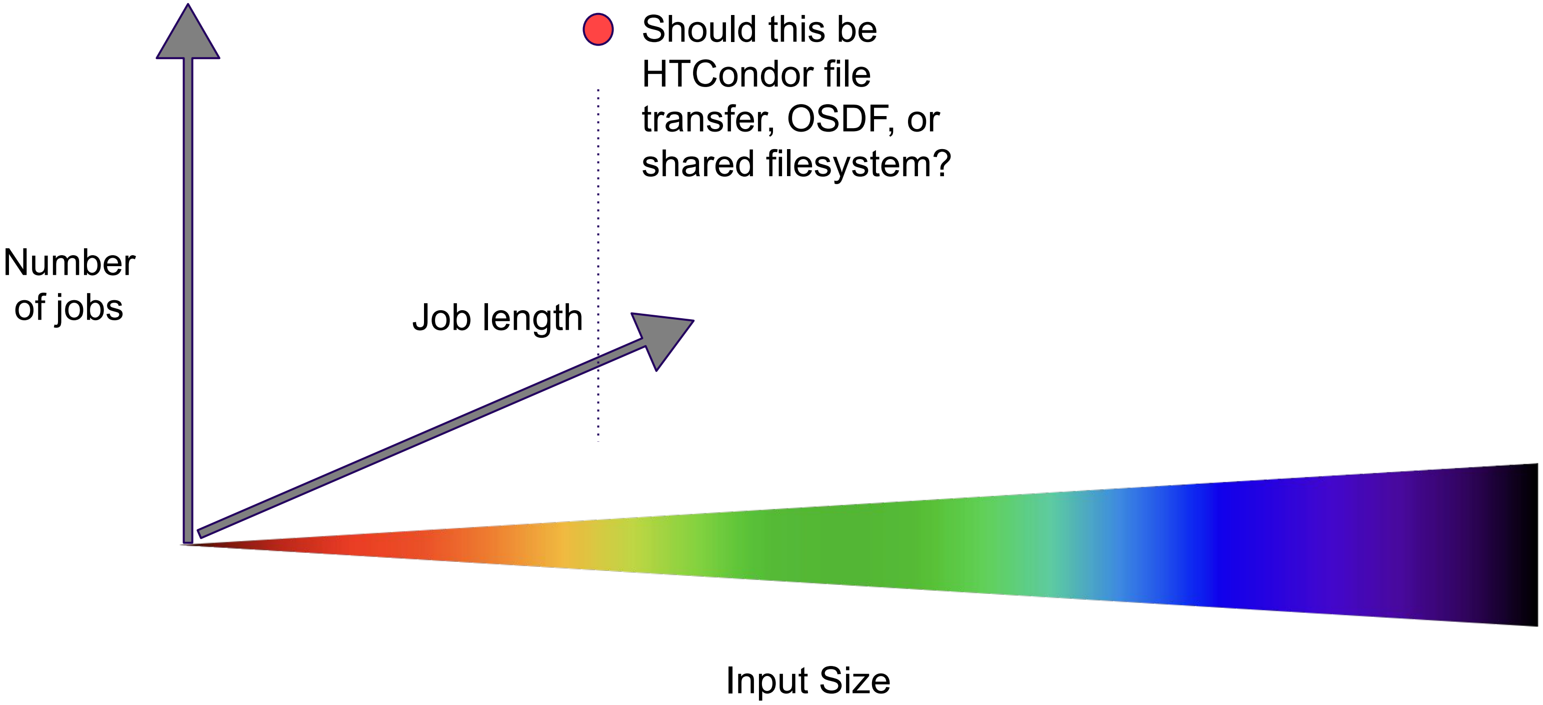


Rule of thumb - many dimensions



● Should this be HTCondor file transfer, OSDF, or shared filesystem?

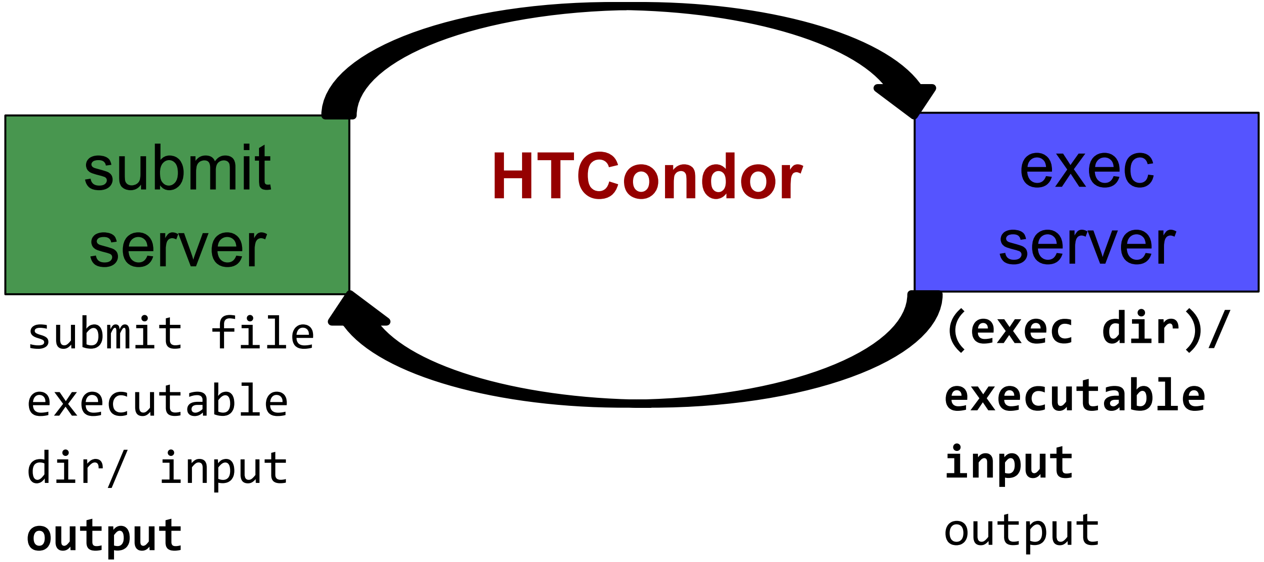
Rule of thumb - many dimensions


Should this be HTCondor file transfer, OSDF, or shared filesystem?
Number of jobs
Job length
Input Size

Handling Data on OSG

- ~~Overview / Things to Consider~~
- **HTCondor File Transfer**
- OSDF
- Shared File Systems and Other Options

Review: HTCondor Data Handling



submit server

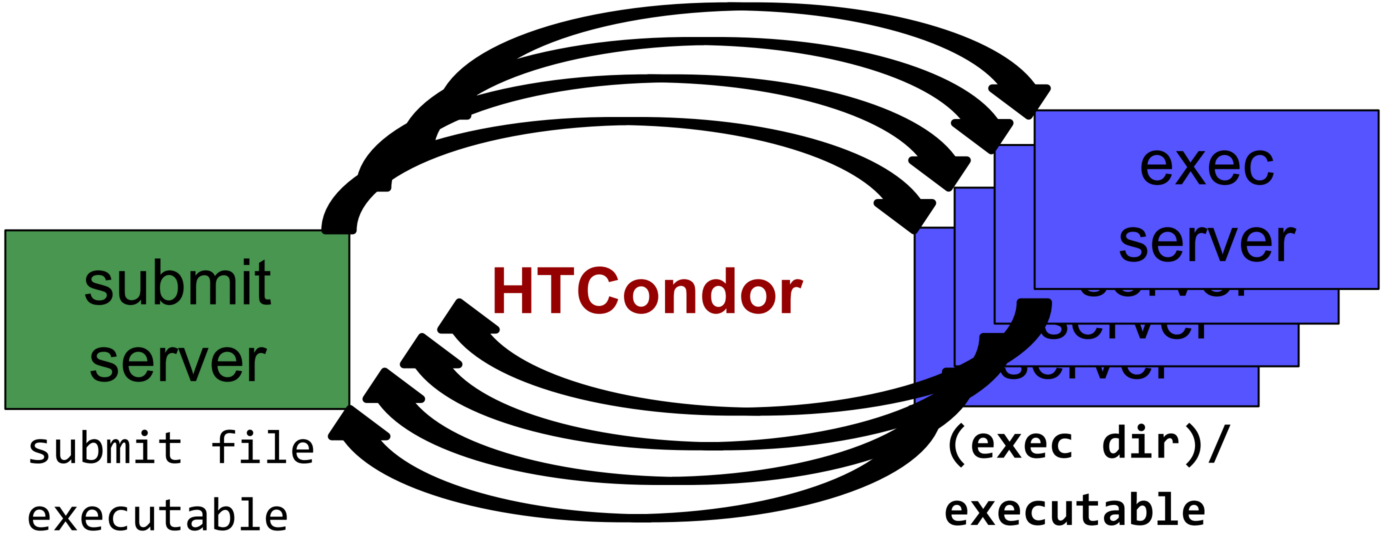
HTCondor

exec server

submit file
executable
dir/ input
output

(exec dir)/
executable
input
output

Network bottleneck: the submit server



submit server

HTCondor

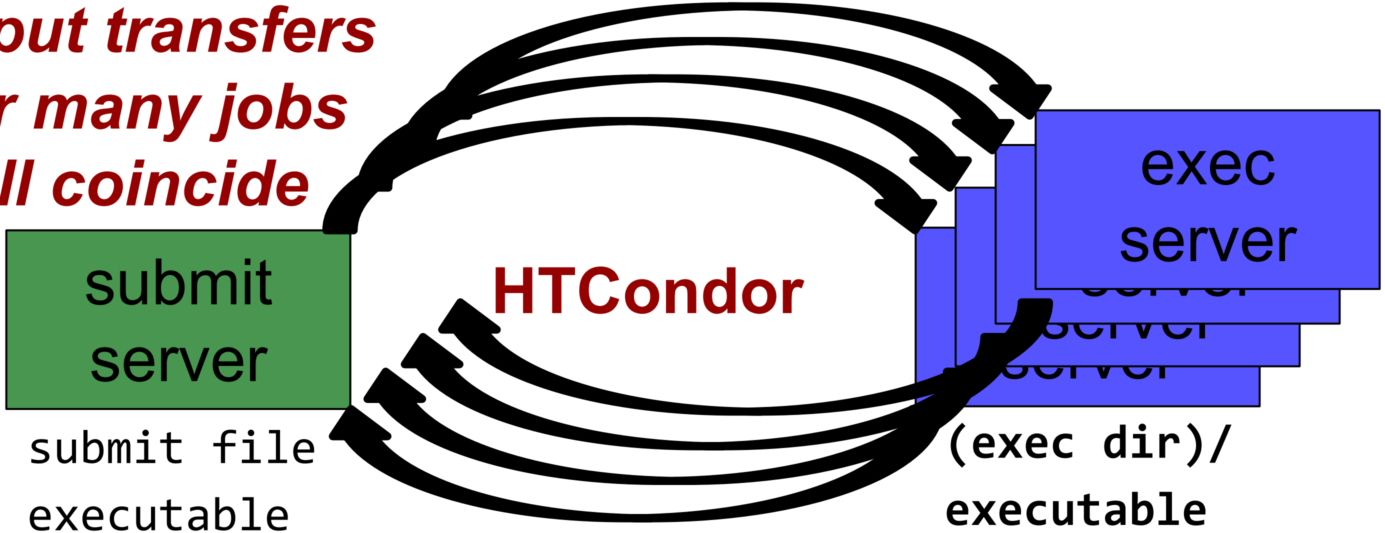
exec server

submit file
executable
dir/ input
output

(exec dir)/
executable
input
output

Network bottleneck: the submit server

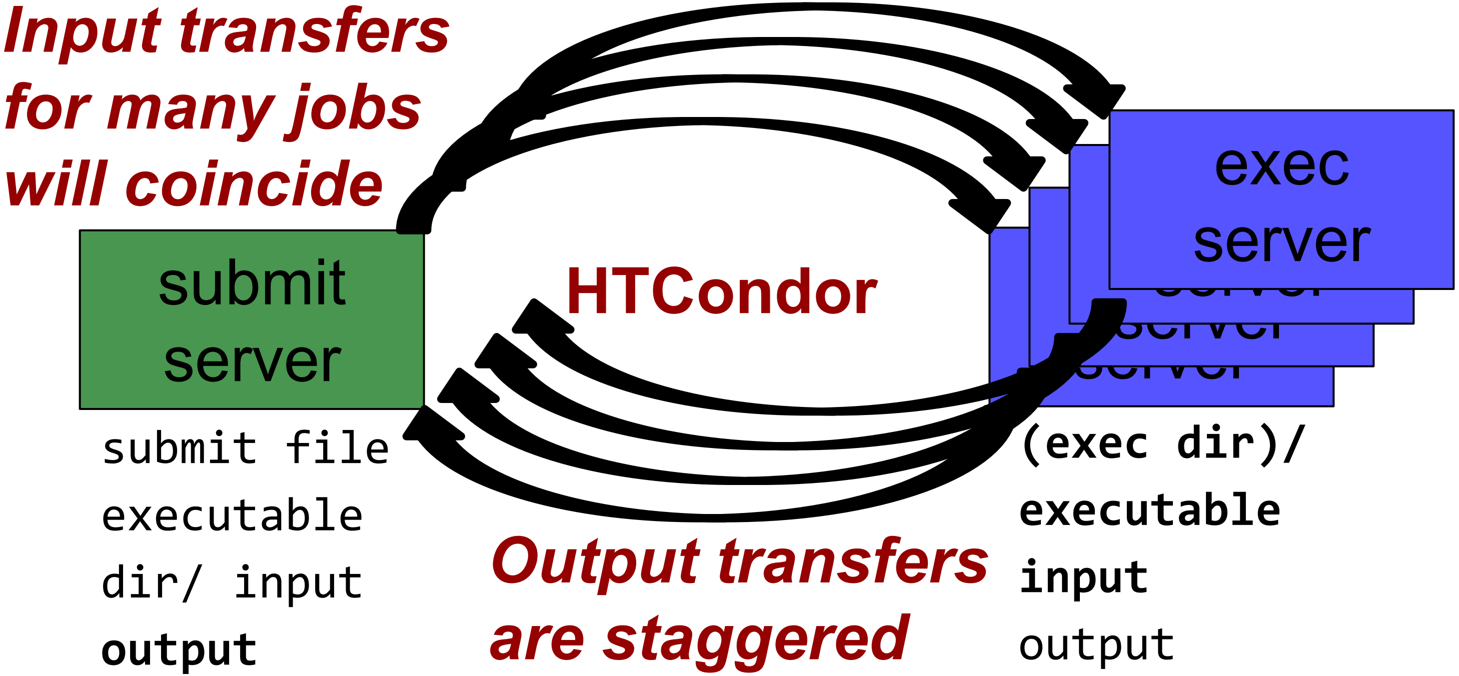
Input transfers for many jobs will coincide



submit file
executable
dir/ input
output

(exec dir)/
executable
input
output

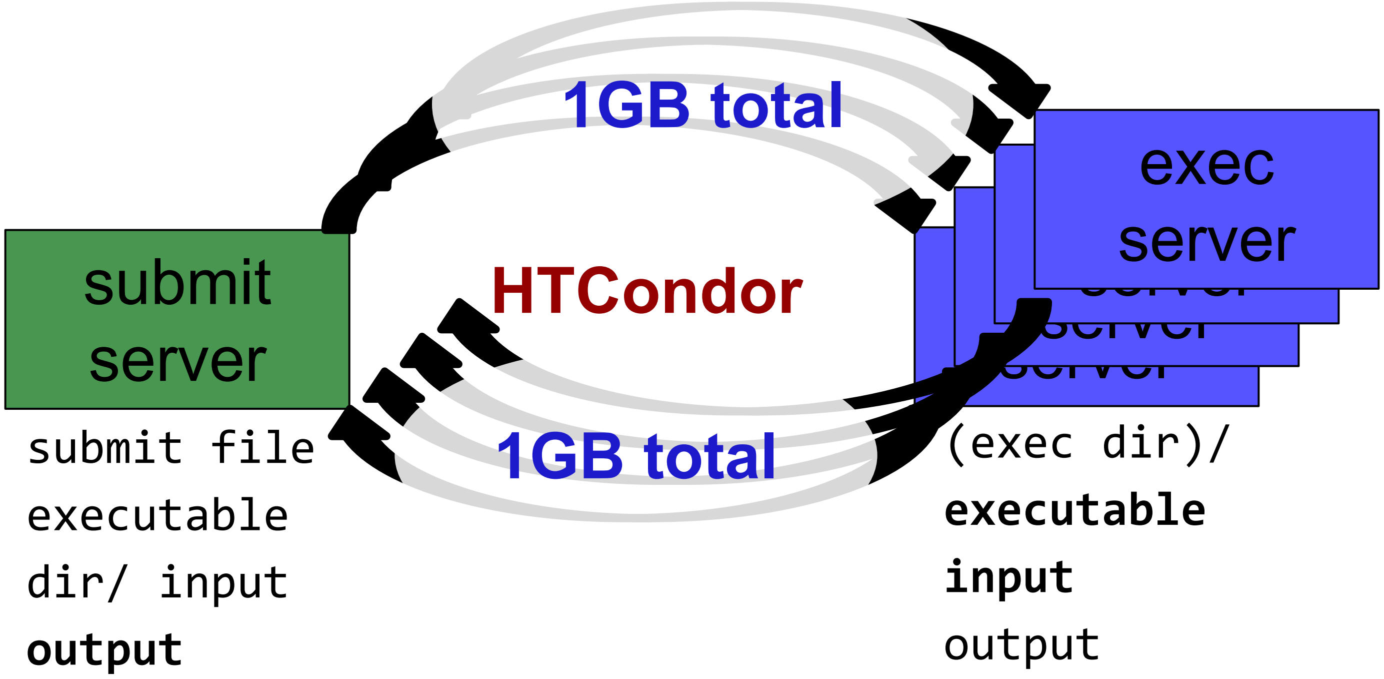
Network bottleneck: the submit server


Input transfers for many jobs will coincide
submit server
HTCondor
exec server
server
server
server
Output transfers are staggered

```
submit file
executable
dir/ input
output
```

```
(exec dir)/
executable
input
output
```

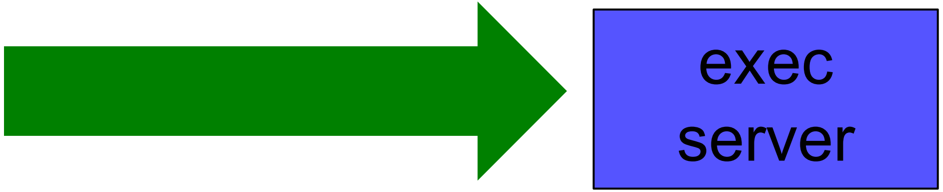
Hardware transfer limits



Handling Data on OSG

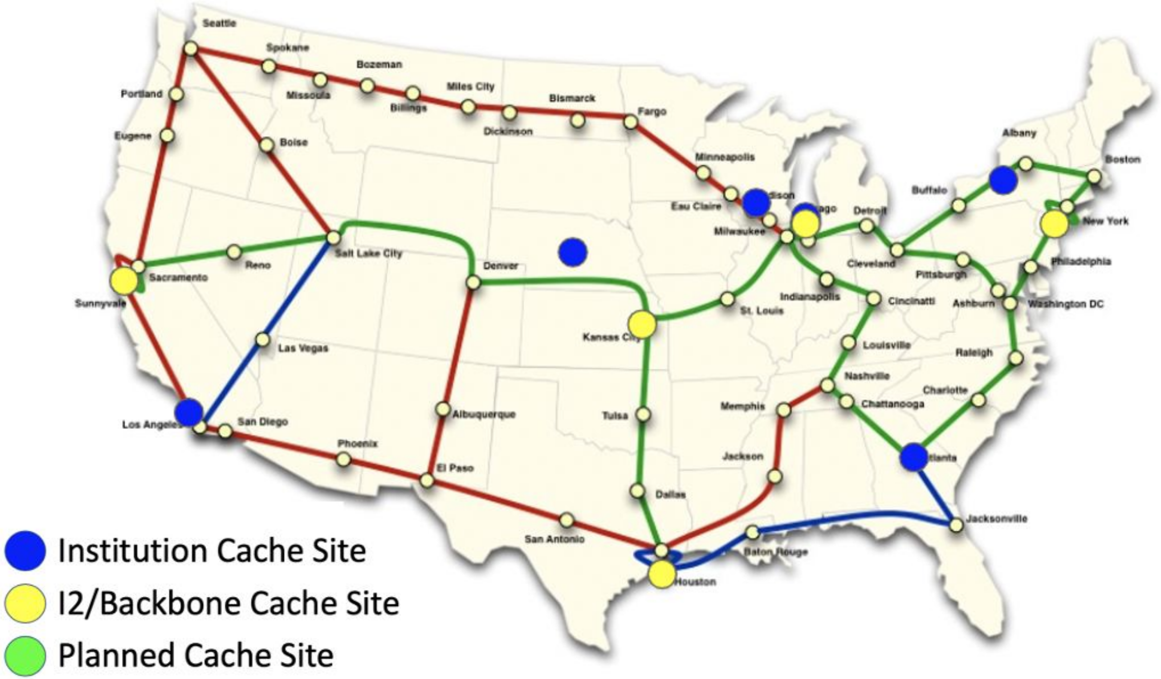
- ~~Overview / Things to Consider~~
- ~~HTCondor File Transfer~~
- **OSDF**
- Shared File Systems

Large input in HTC and OSG



file size	method of delivery
words	within executable or arguments?
tiny – 100MB per file	HTCondor file transfer (up to 1GB total per-job)
100MB – 1GB, shared	download from web server (local caching)
1GB – 20GB, unique or shared	OSDF (regional replication)
10 GB - TBs	shared file system (local copy, local execute servers)

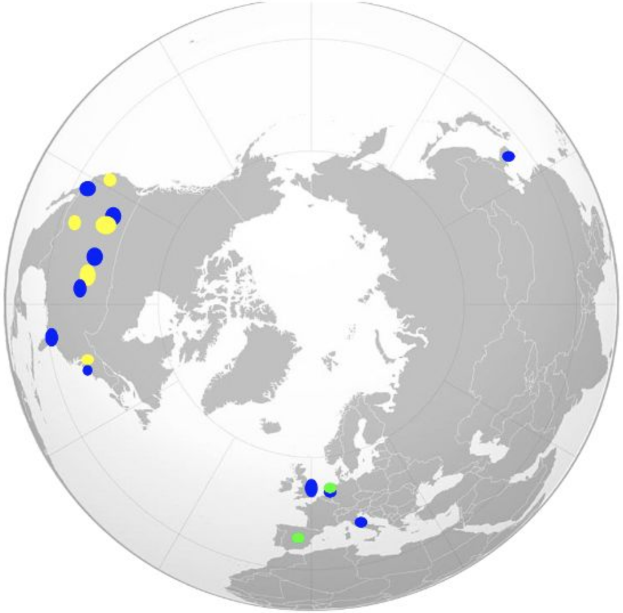
Open Science Data Federation (OSDF)



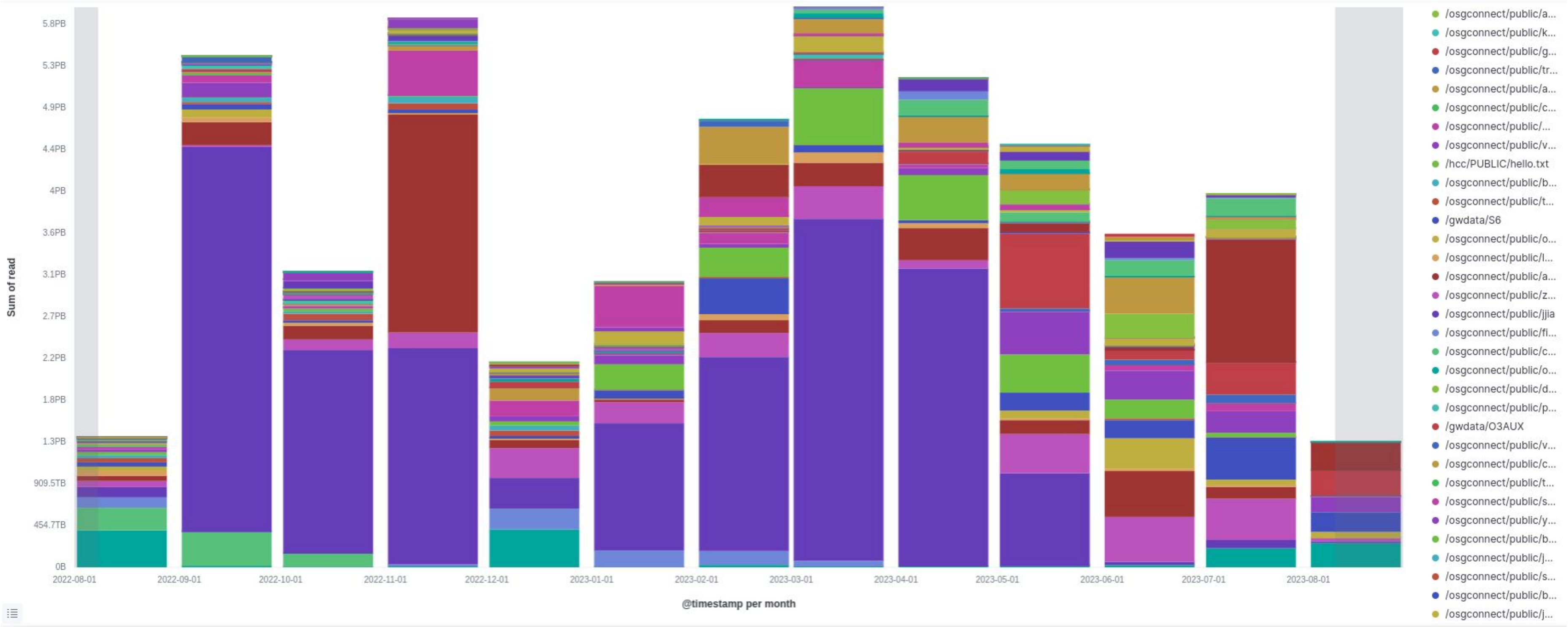
● Institution Cache Site

● I2/Backbone Cache Site

● Planned Cache Site



OSDF Usage on OSG

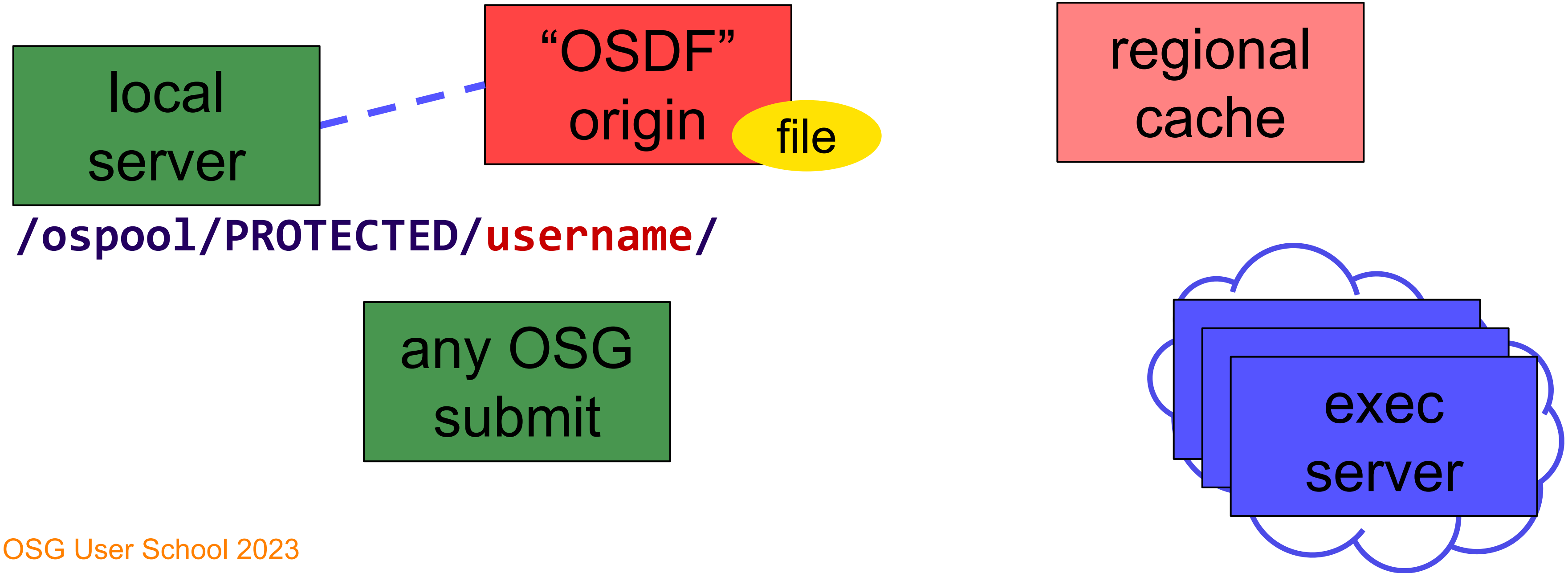


OSDF Considerations

- Available at ~95% of OSG sites
- Regional caches on *very fast* networks
  - **Recommended max file size: 20 GB**
- Can copy multiple files totaling >10GB
- Change name when update files

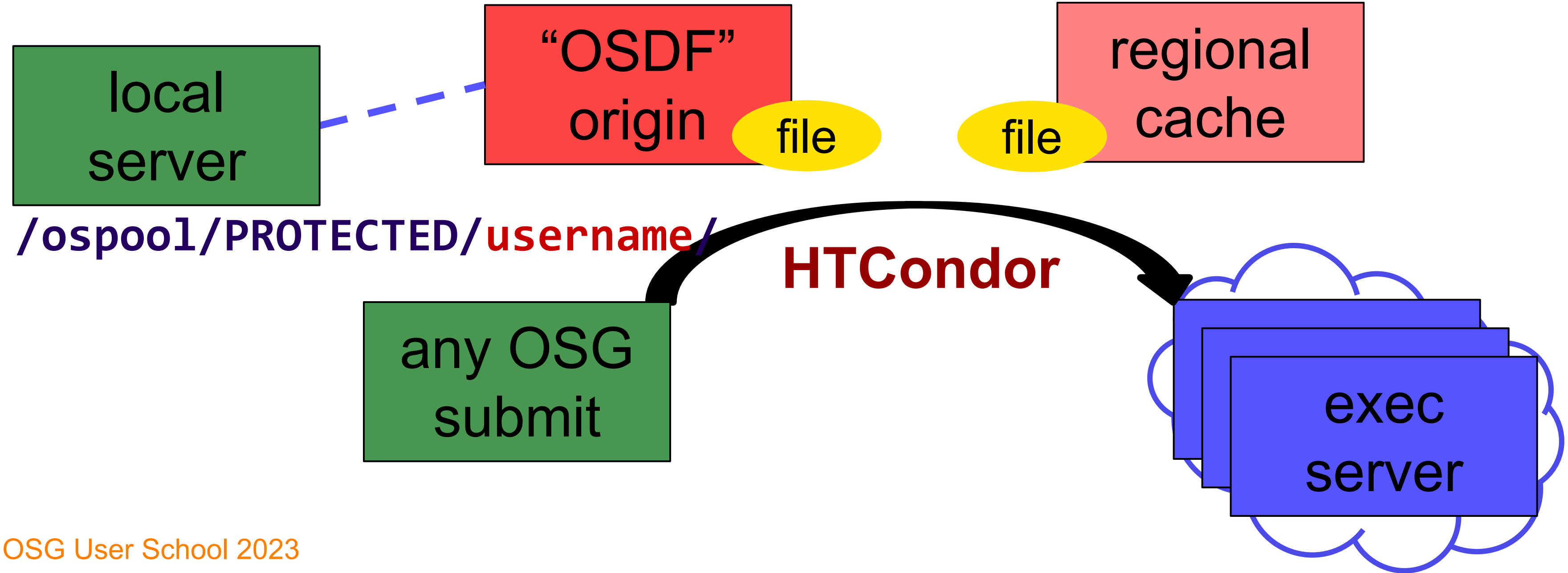
Placing Files in OSDF

- Place files in `/ospool/PROTECTED/username/`



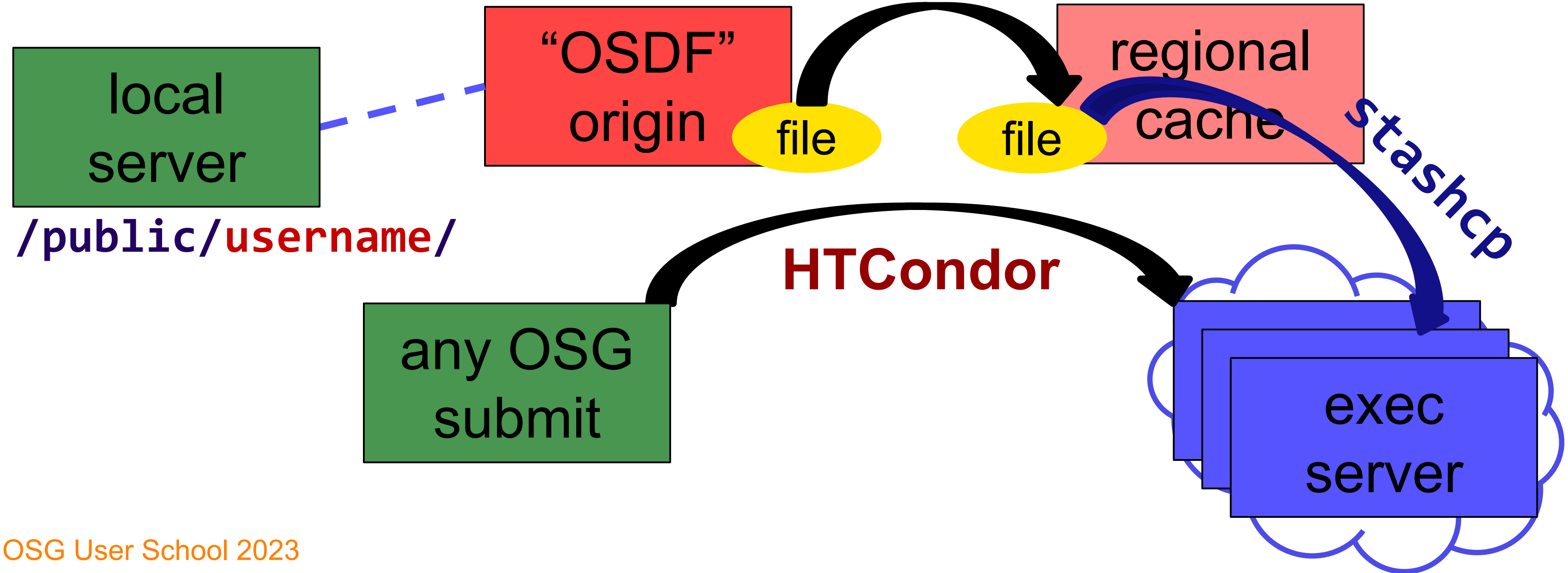
Obtaining Files in OSDF

- Use HTCondor transfer for other files

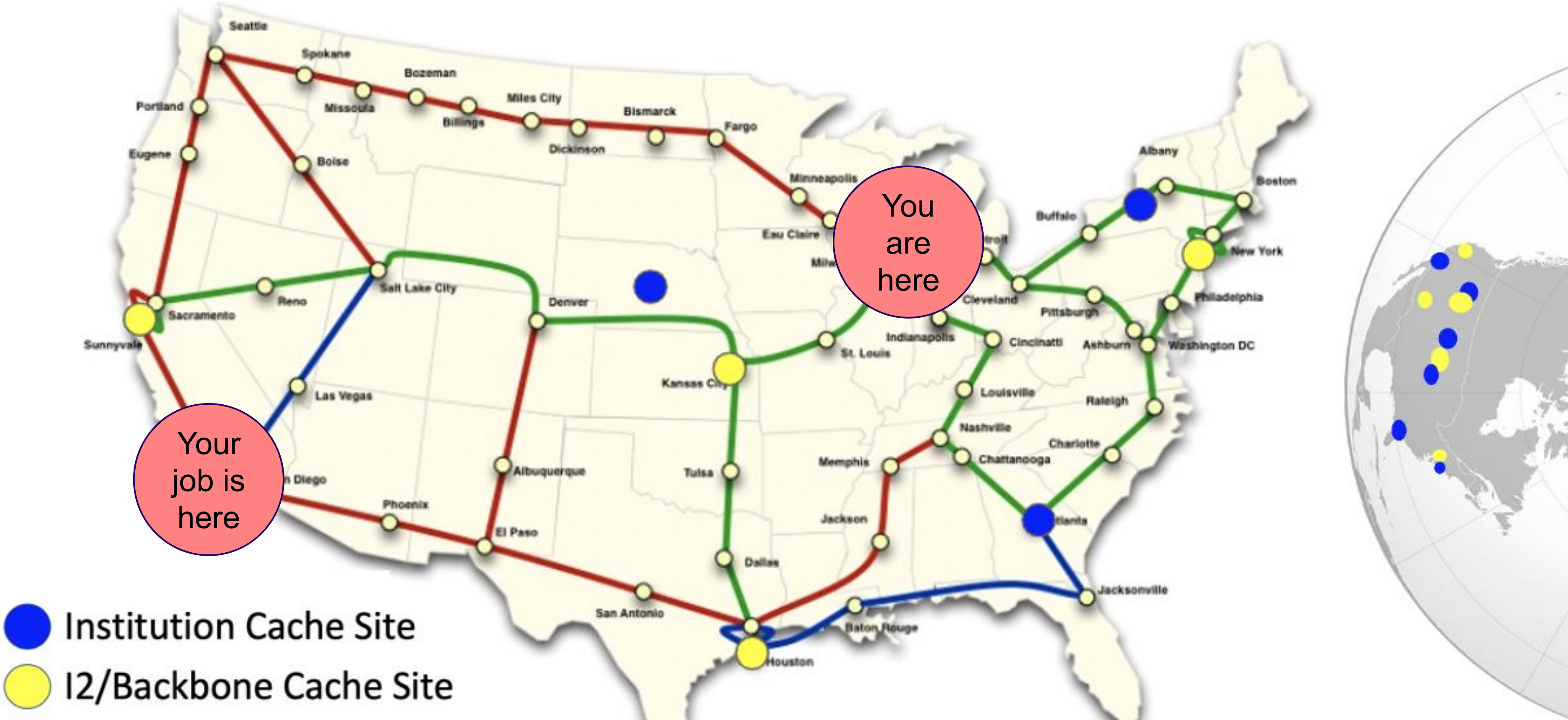


Obtaining Files in Stash

- Download using `stashcp` command



Open Science Data Federation (OSDF)



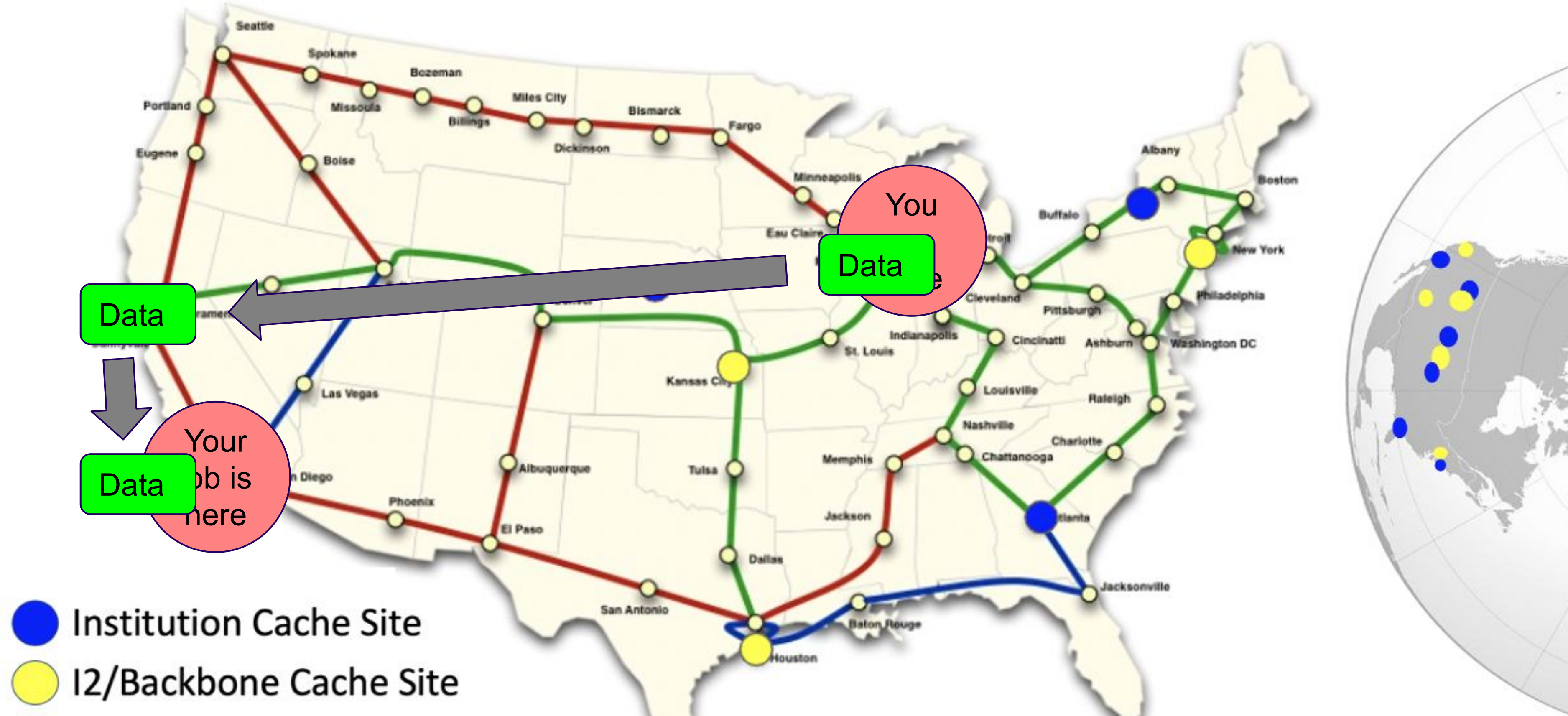
You are here

Your job is here

● Institution Cache Site

● I2/Backbone Cache Site

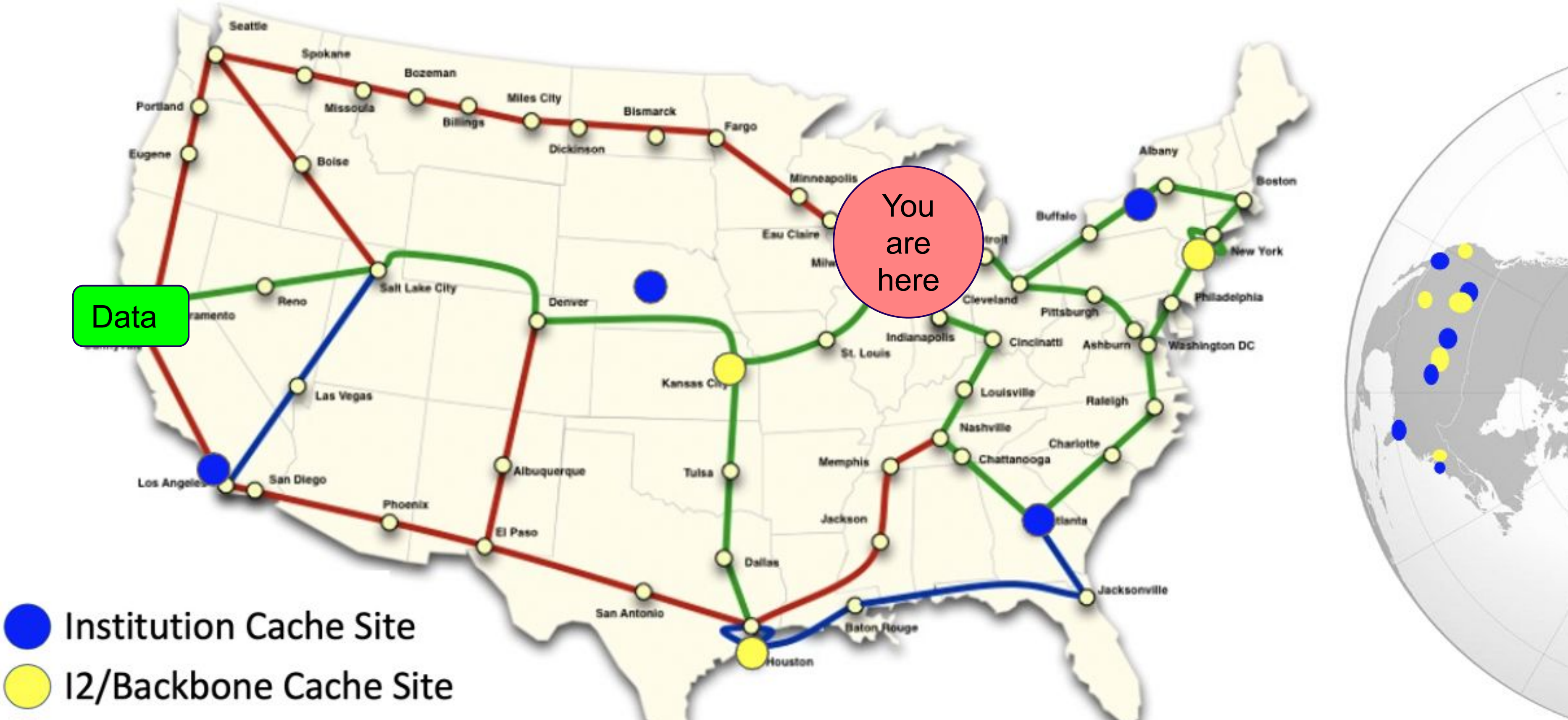
Open Science Data Federation (OSDF)



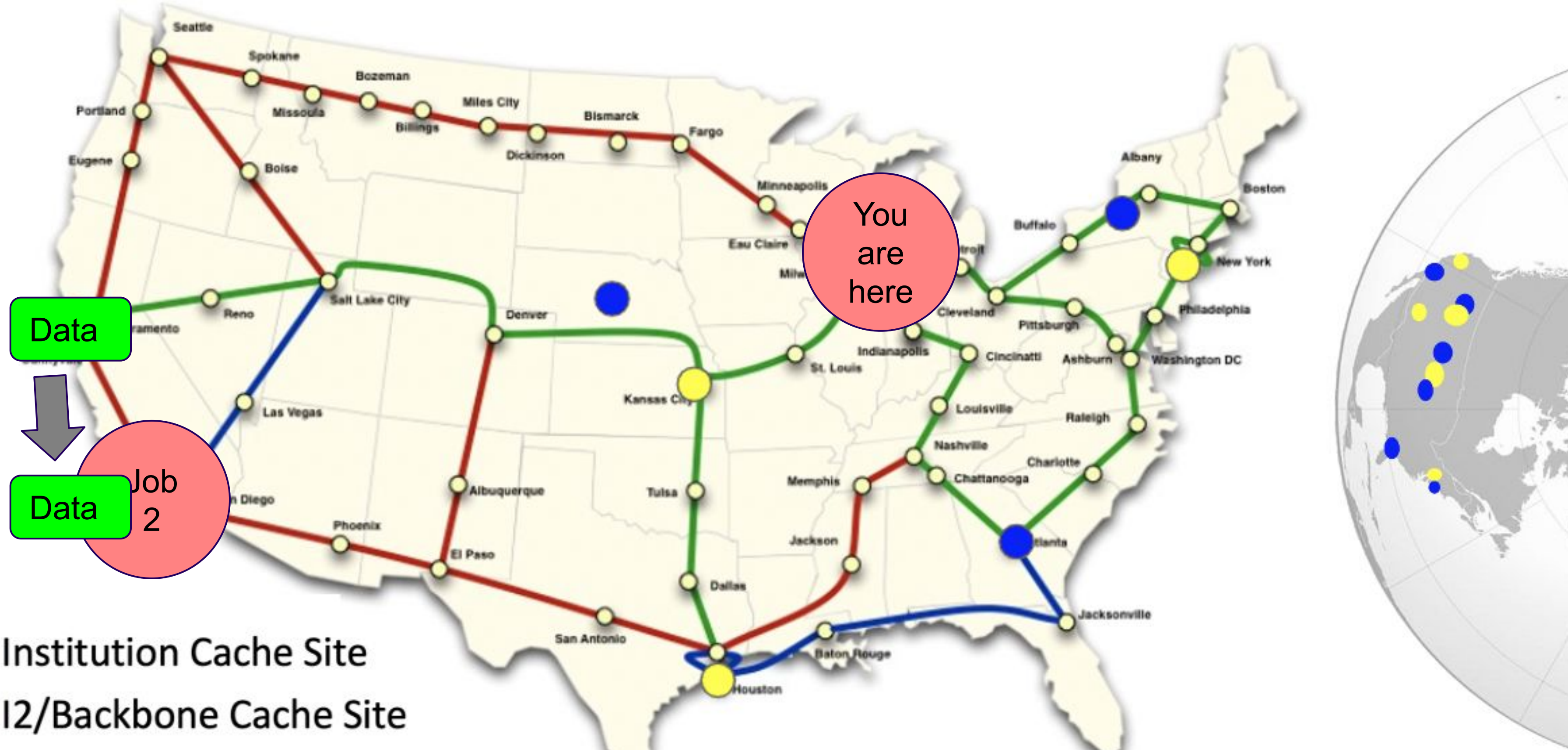
● Institution Cache Site

● I2/Backbone Cache Site

Open Science Data Federation (OSDF)



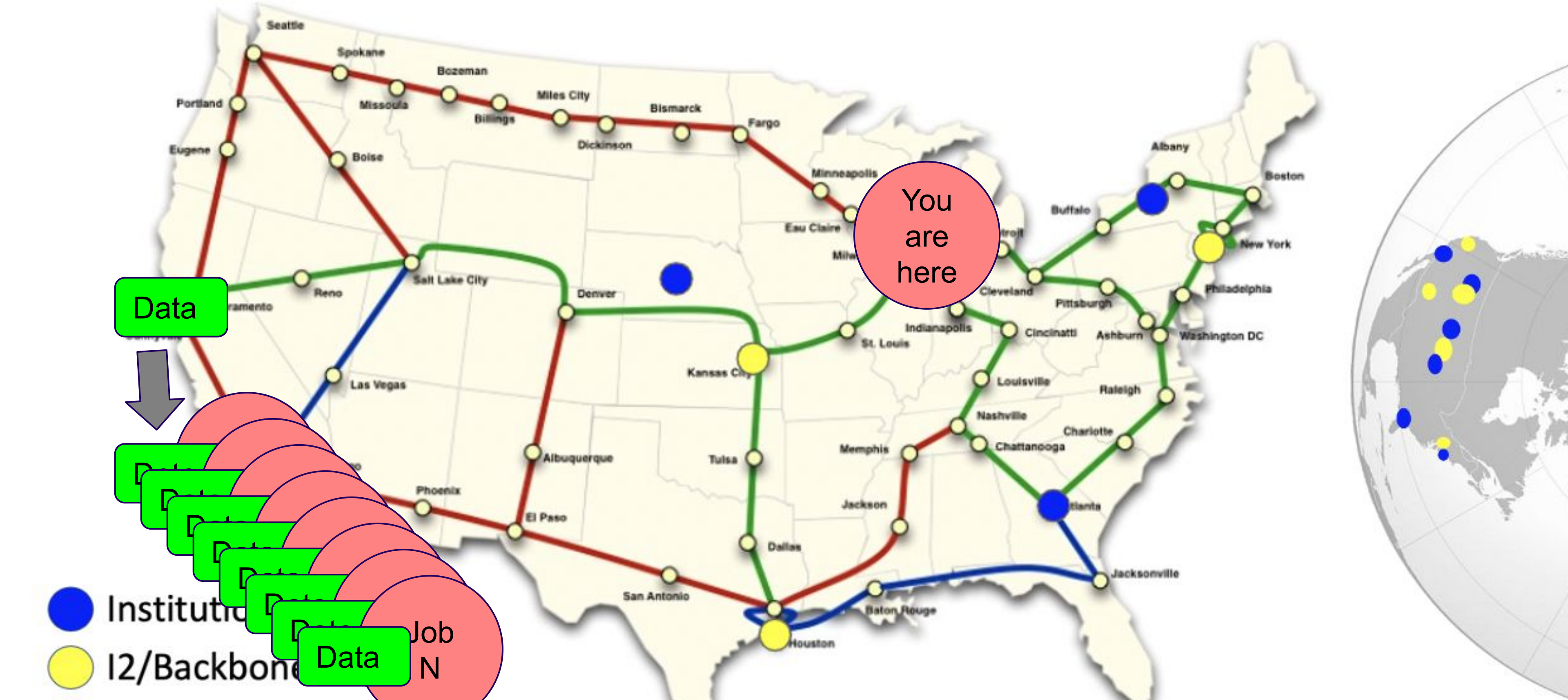
Open Science Data Federation (OSDF)



● Institution Cache Site

● I2/Backbone Cache Site

Open Science Data Federation (OSDF)



In the Submit File

```
transfer_input_files = osdf:///ospool/PROTECTED/USERNAME/…
```

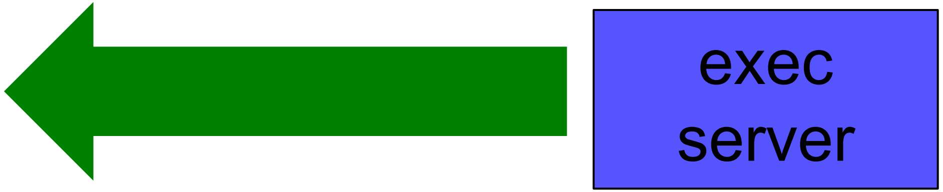
How about output?

Output for HTC and OSG

exec server

amount	method of delivery
~~words~~	~~within executable or arguments?~~
tiny – **1GB, total**	HTCondor file transfer
1GB - 20GB, unique or shared	OSDF
20GB+, total	shared file system (local copy, local execute servers)

Output for HTC and OSG



amount	method of delivery
~~words~~	~~within executable or arguments?~~
tiny – **1GB, total**	HTCondor file transfer
1GB – 20GB, unique or shared	OSDF
20GB+, total	shared file system (local copy, local execute servers)

Writing to stash

```
transfer_output_remaps = "Output.txt = osdf:///ospool/PROTECTED/<username>/Output.txt"
```

Other Considerations

- Only use these options if you MUST!!
  - Each comes with limitations on site accessibility and/or job performance, and extra data management concerns

file size	method of delivery
words	within executable or arguments?
tiny – 10MB per file	HTCondor file transfer (up to 1GB total per-job)
10MB – 1GB, shared	download from web server (local caching)
1GB - 10GB, unique or shared	OSDF (regional replication)
10 GB - TBs	shared file system (local copy, local execute servers)

Cleaning Up Old Data

Make sure to delete data when you no longer need it in the origin!!!

Servers do NOT have unlimited space!

- Some may regularly clean old data for you. Check with local support.

Quick Reference

Option	Input or Output?	File size limits	Placing files	In-job file movement	Accessibility?
HTCondor file transfer	Both	100 MB/file (in), 1 GB/file (out); 1 GB/tot (either)	via HTCondor submit node	via HTCondor submit file	anywhere HTCondor jobs can run
OSDF	Both	20 GB/file	via HTCondor submit server	transfer_*_file / stashcp command	OSG-wide (most sites), by anyone
Shared filesystem	Input, likely output	TBs (may vary)	via mount location (may vary)	use directly, or copy into/out of execute dir	local cluster, only by YOU (usually)

Acknowledgments

- This work was supported by NSF grants OAC-1836650, and OAC-2030508

Additional Slides

Shared Filesystem Details

(Local) Shared Filesystems

- data stored on file servers, but network-mounted to local submit and execute servers
- use local user accounts for file permissions
  - Jobs run as YOU!
  - readable (input) and writable (output, most of the time)
- *MOST* perform better with fewer large files (versus many small files of typical HTC)

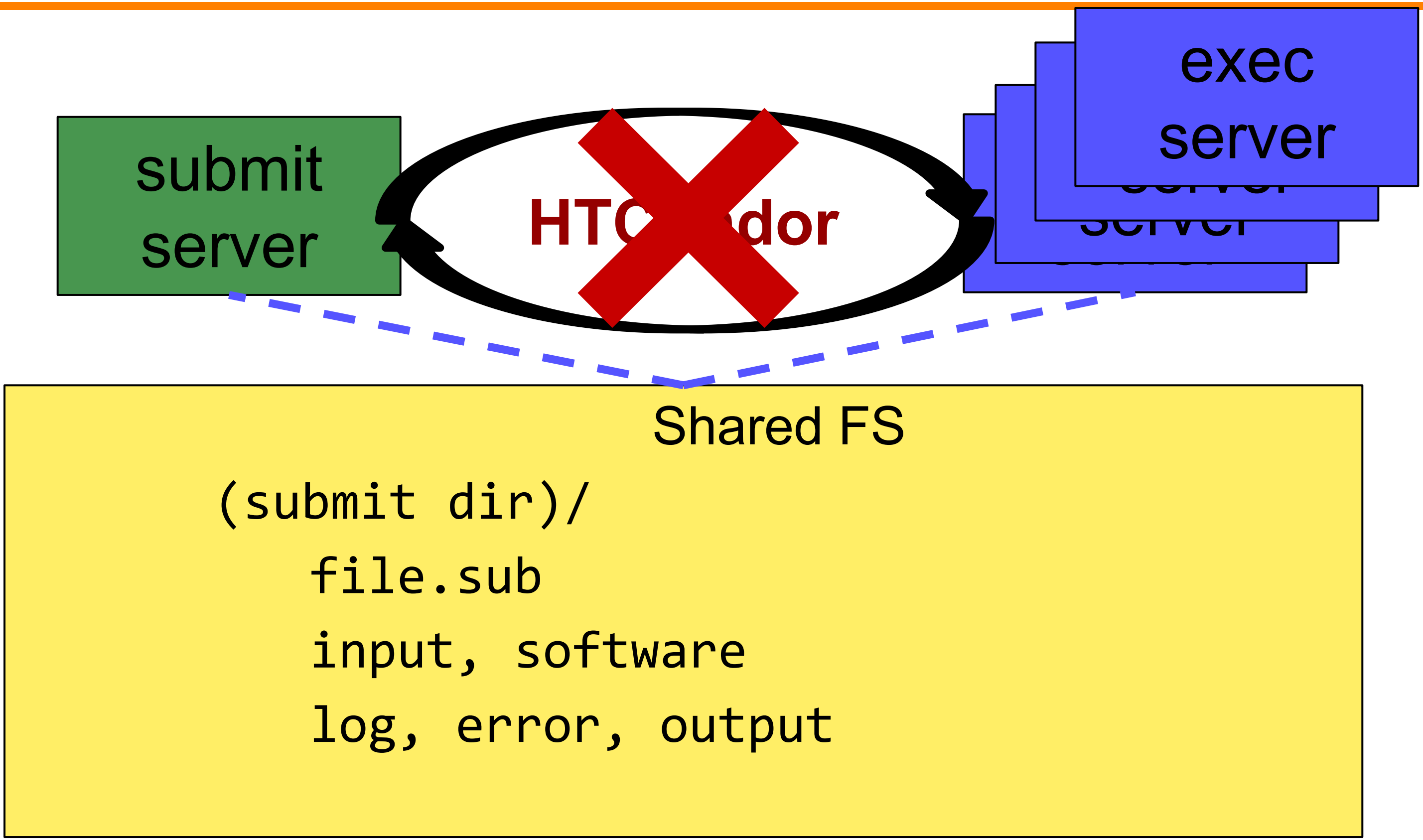
Shared FS Technologies

- *via network mount*
  - NFS
  - AFS
  - Lustre
  - **`/staging`** (may use NFS mount)
  - Isilon (may use NSF mount)
- *distributed file systems (data on many exec servers)*
  - HDFS (Hadoop)
  - CEPH

Shared FS Configurations

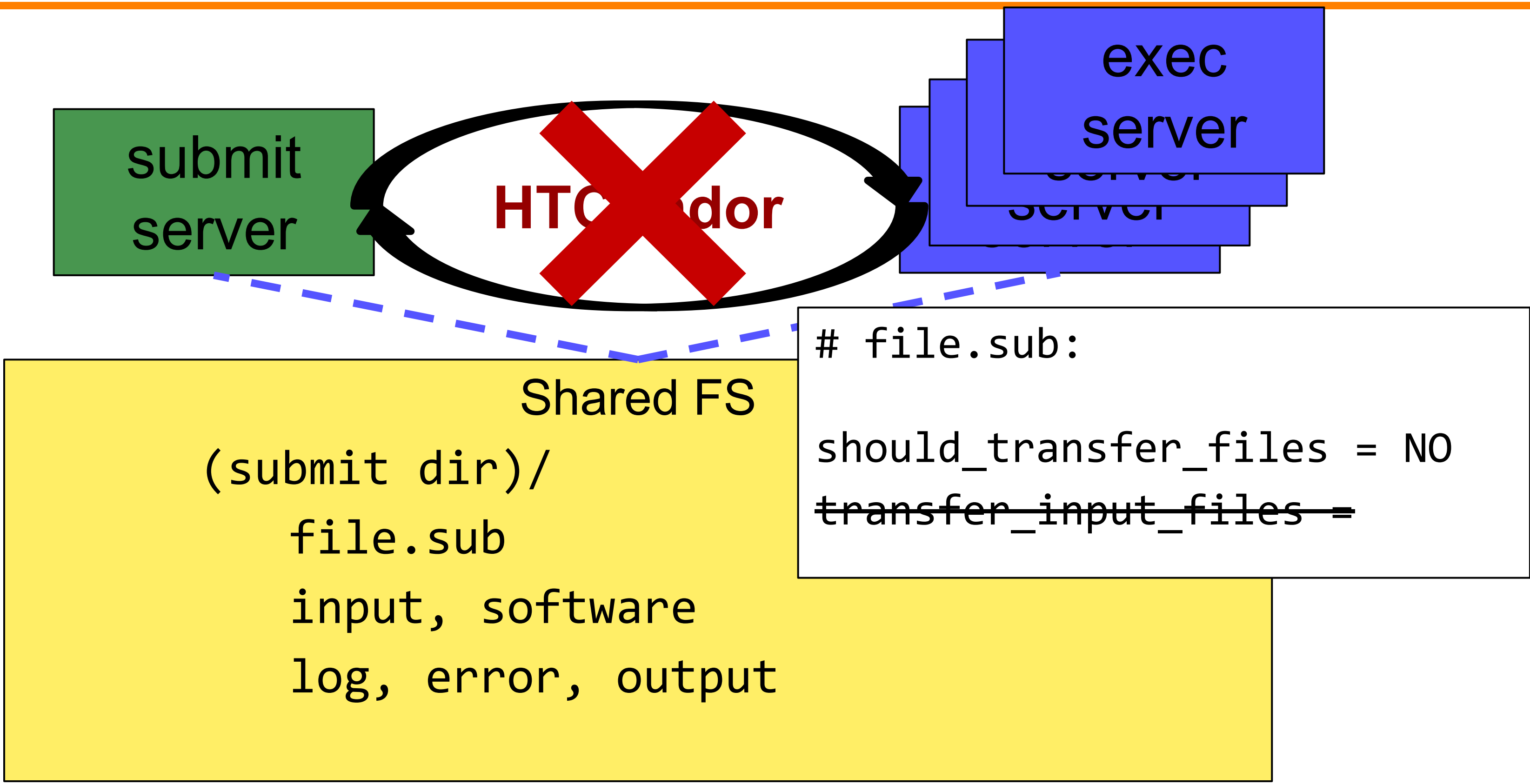
1. Submit directories *WITHIN* the shared filesystem
   - most campus clusters
   - limits HTC capabilities!!
2. Shared filesystem separate from local submission directories
   - supplement local HTC systems
   - treated more as a repository for VERY large data (>GBs)
3. Read-only (input-only) shared filesystem
   - Treated as a repository for VERY large input, only

Submit dir within shared FS



```
(submit dir)/
    file.sub
    input, software
    log, error, output
```

Submit dir within shared FS



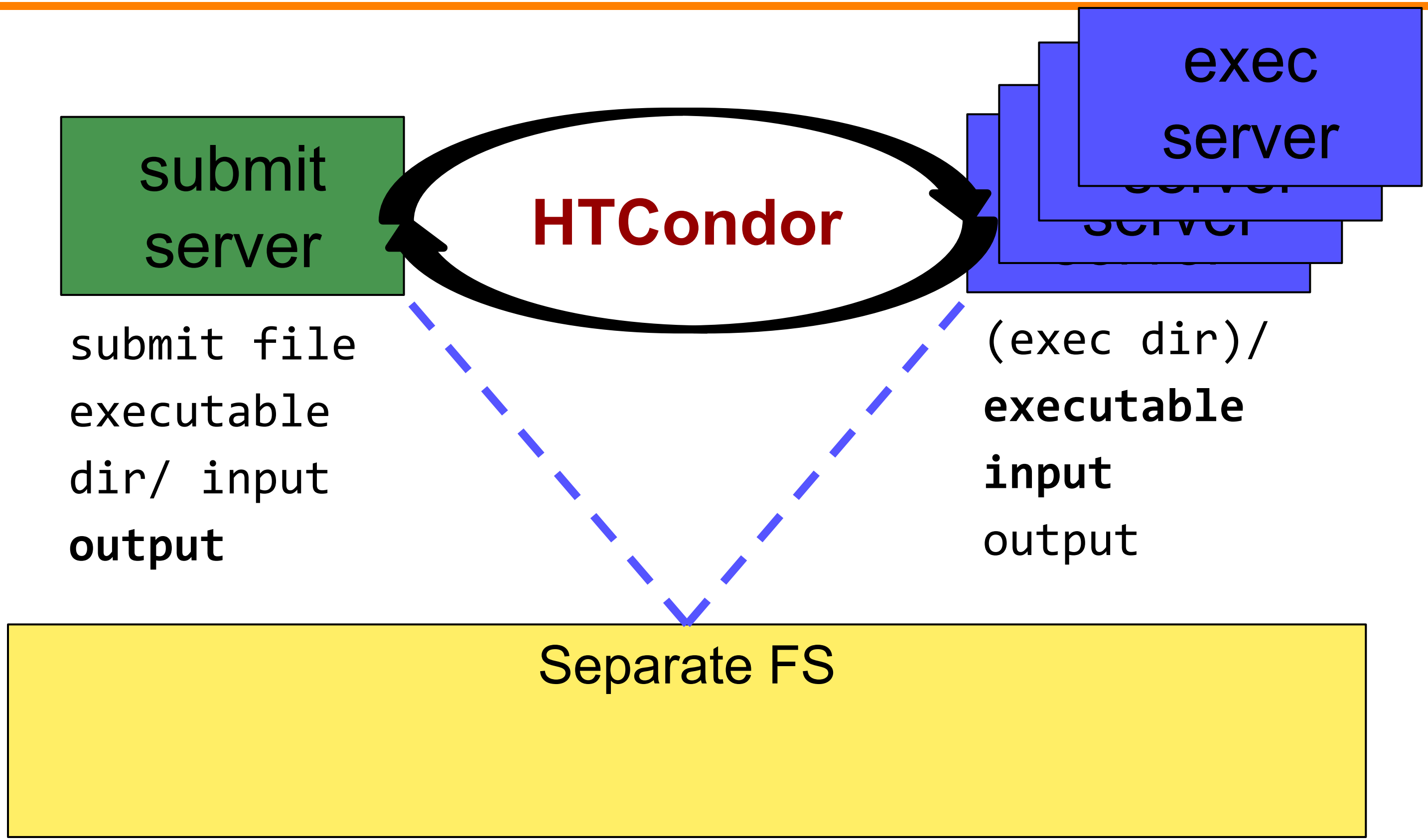
Shared FS

```
(submit dir)/
    file.sub
    input, software
    log, error, output
```

```
# file.sub:

should_transfer_files = NO
```
~~transfer_input_files =~~

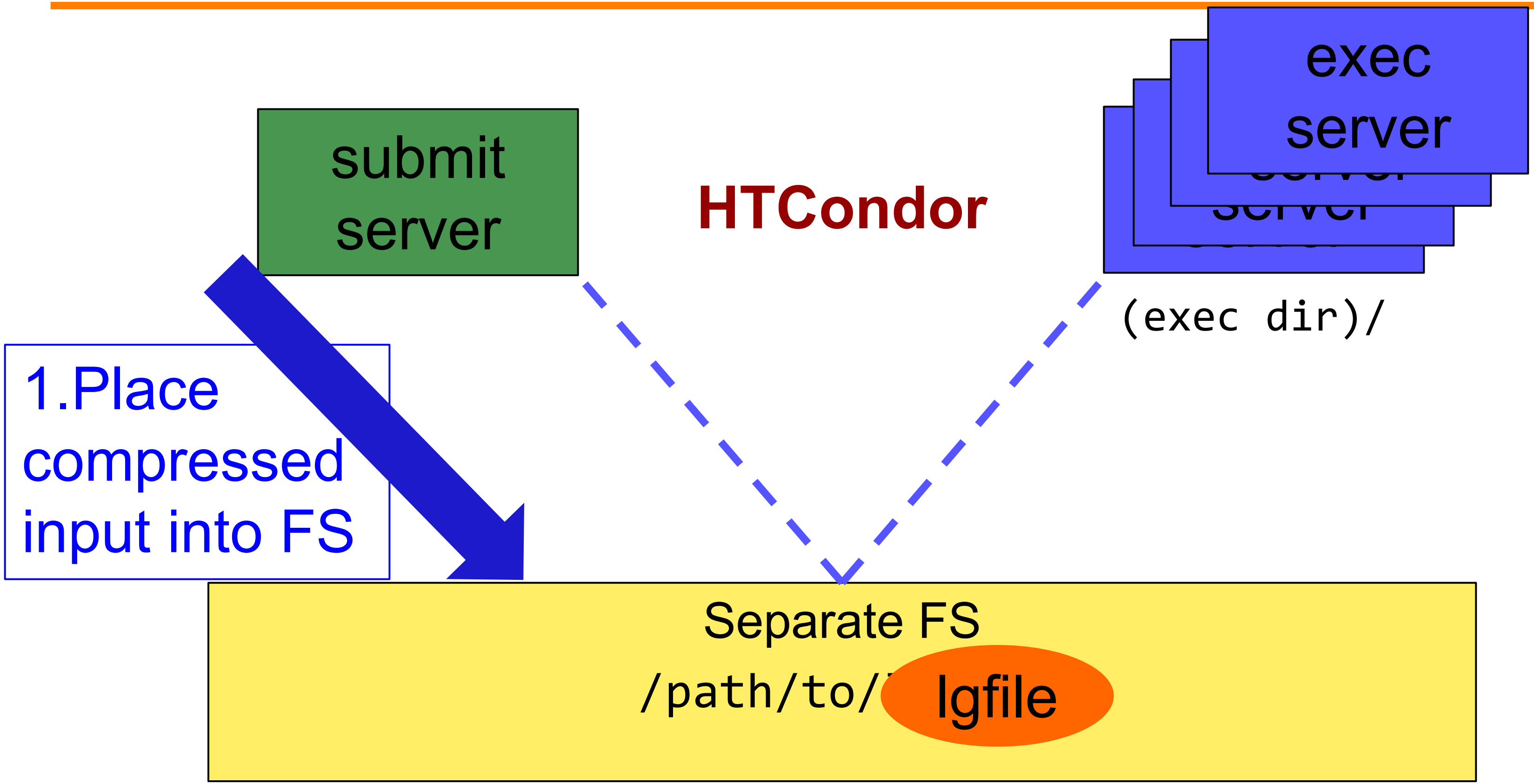
Separate shared FS



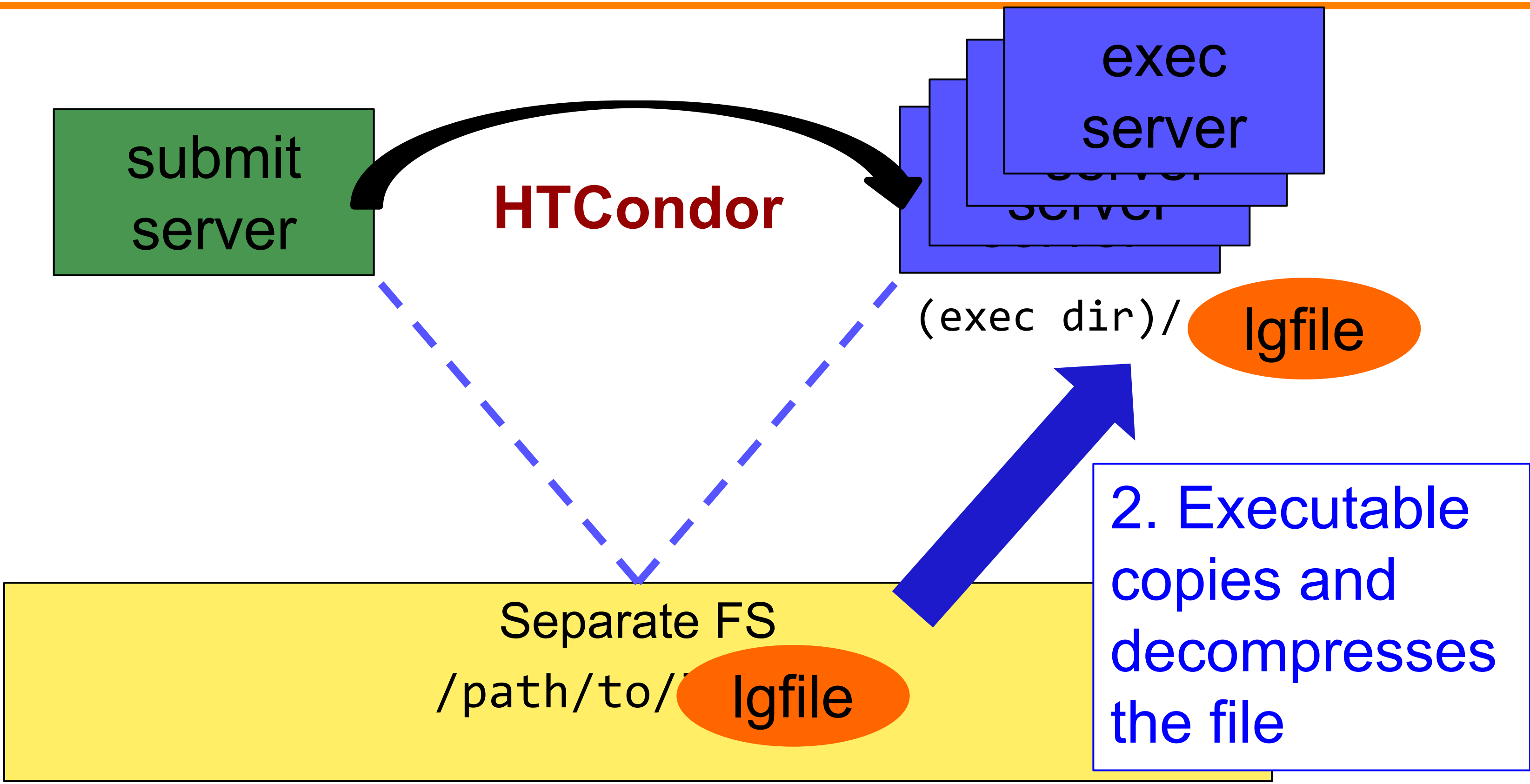
```
submit file
executable
dir/ input
output
```

```
(exec dir)/
executable
input
output
```

Separate shared FS - Input

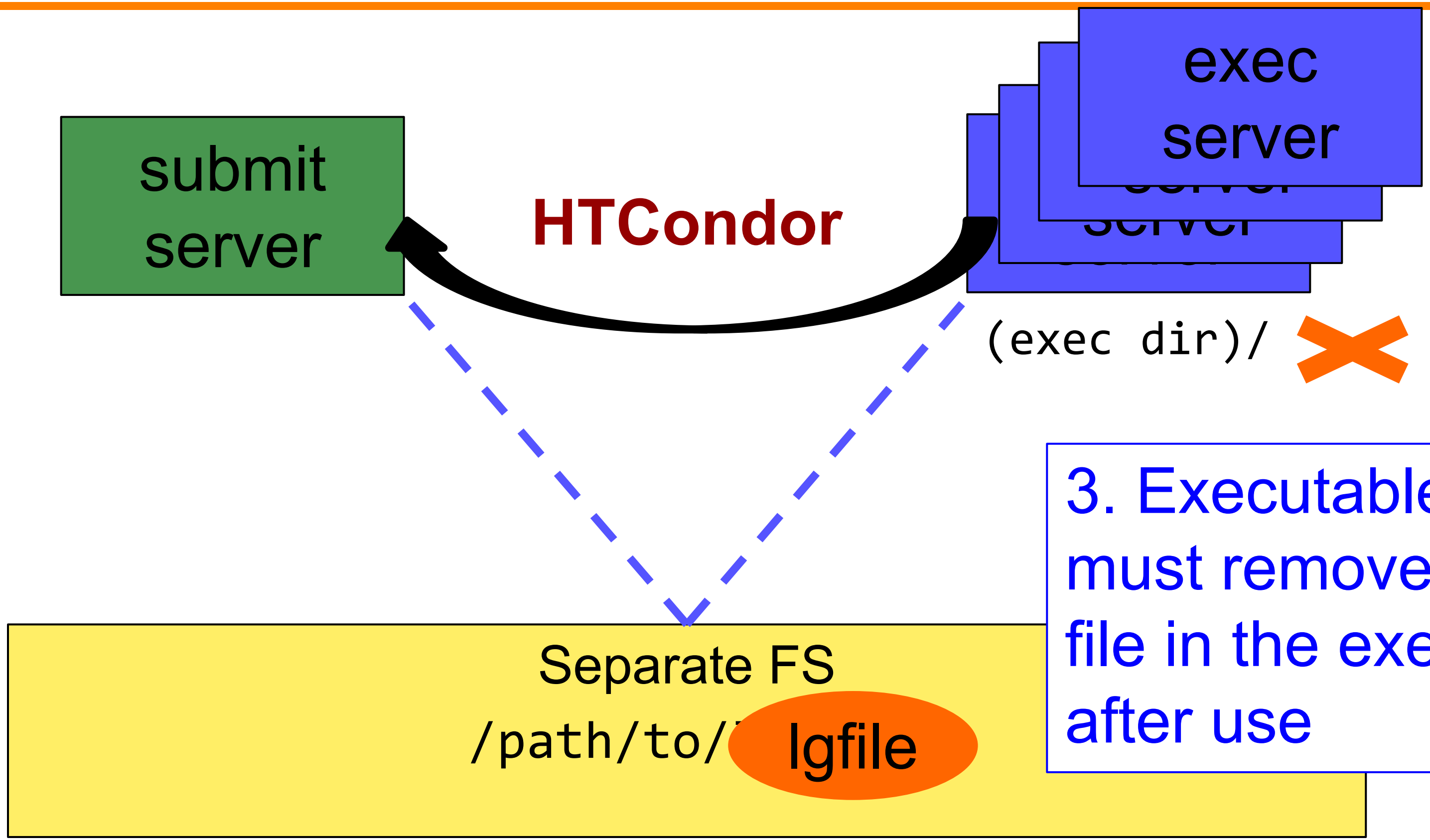


Separate shared FS - Input



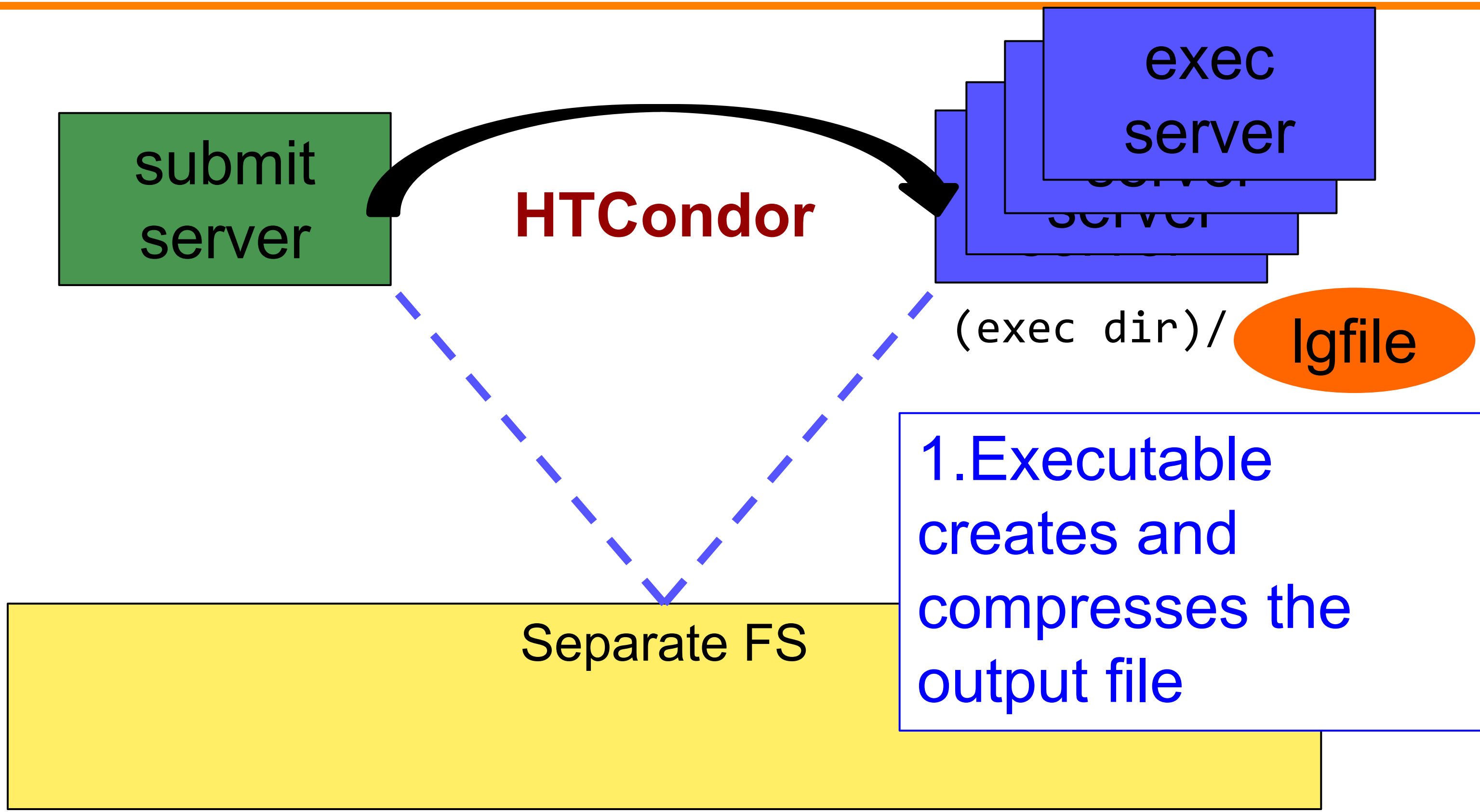
2. Executable copies and decompresses the file

Separate shared FS - Input



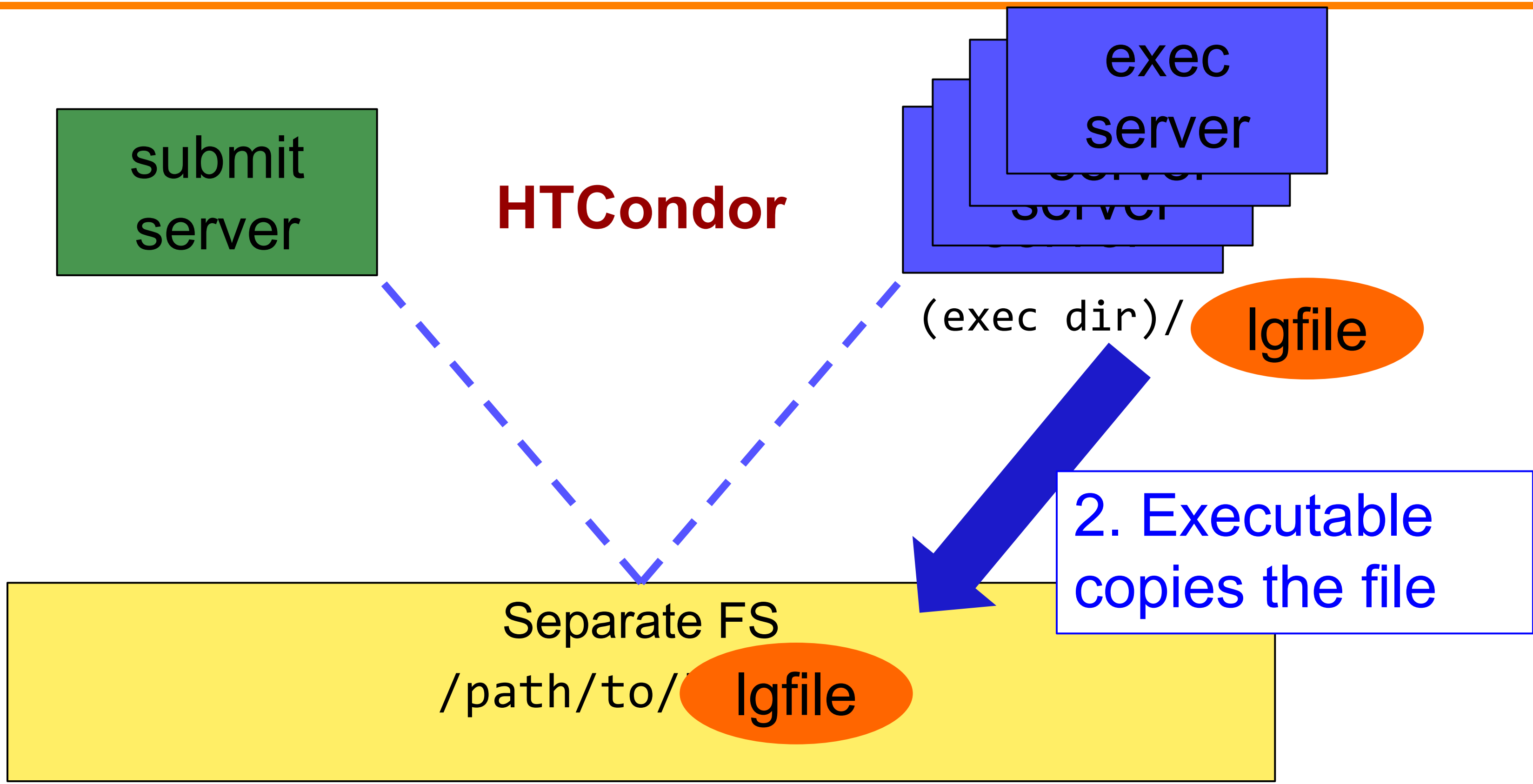
3. Executable must remove the file in the exec dir after use

Separate shared FS - Output



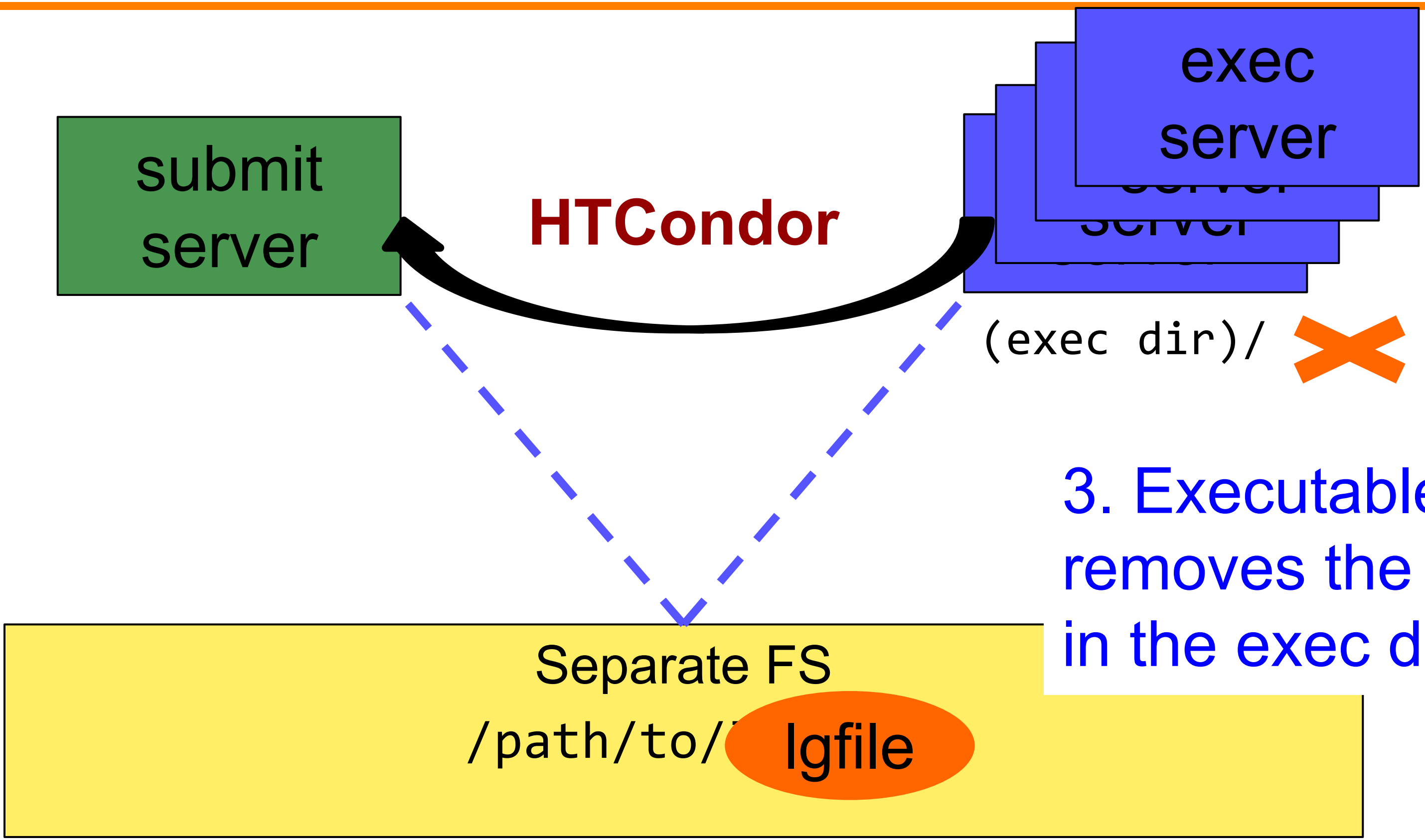
1. Executable creates and compresses the output file

Separate shared FS - Output



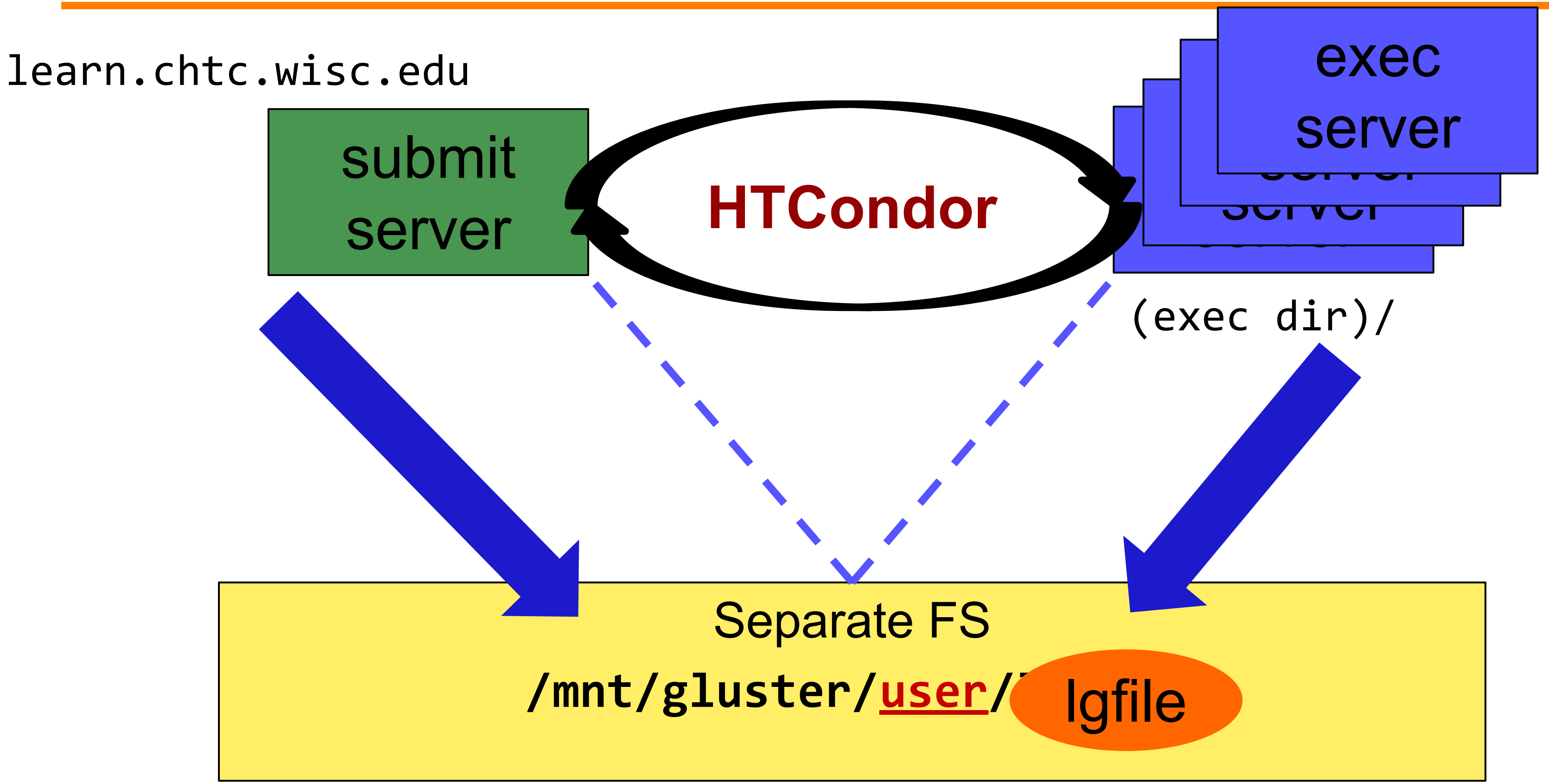
2. Executable copies the file

Separate shared FS - Output



3. Executable removes the file in the exec dir

At UW-Madison (Ex. 3.1-3.2)



`learn.chtc.wisc.edu`

submit server

HTCondor

exec server

`(exec dir)/`

Separate FS

`/mnt/gluster/user/`

lgfile