Introduction to DHTC

Brian Lin

OSG Software Team

University of Wisconsin - Madison

blin@cs.wisc.edu
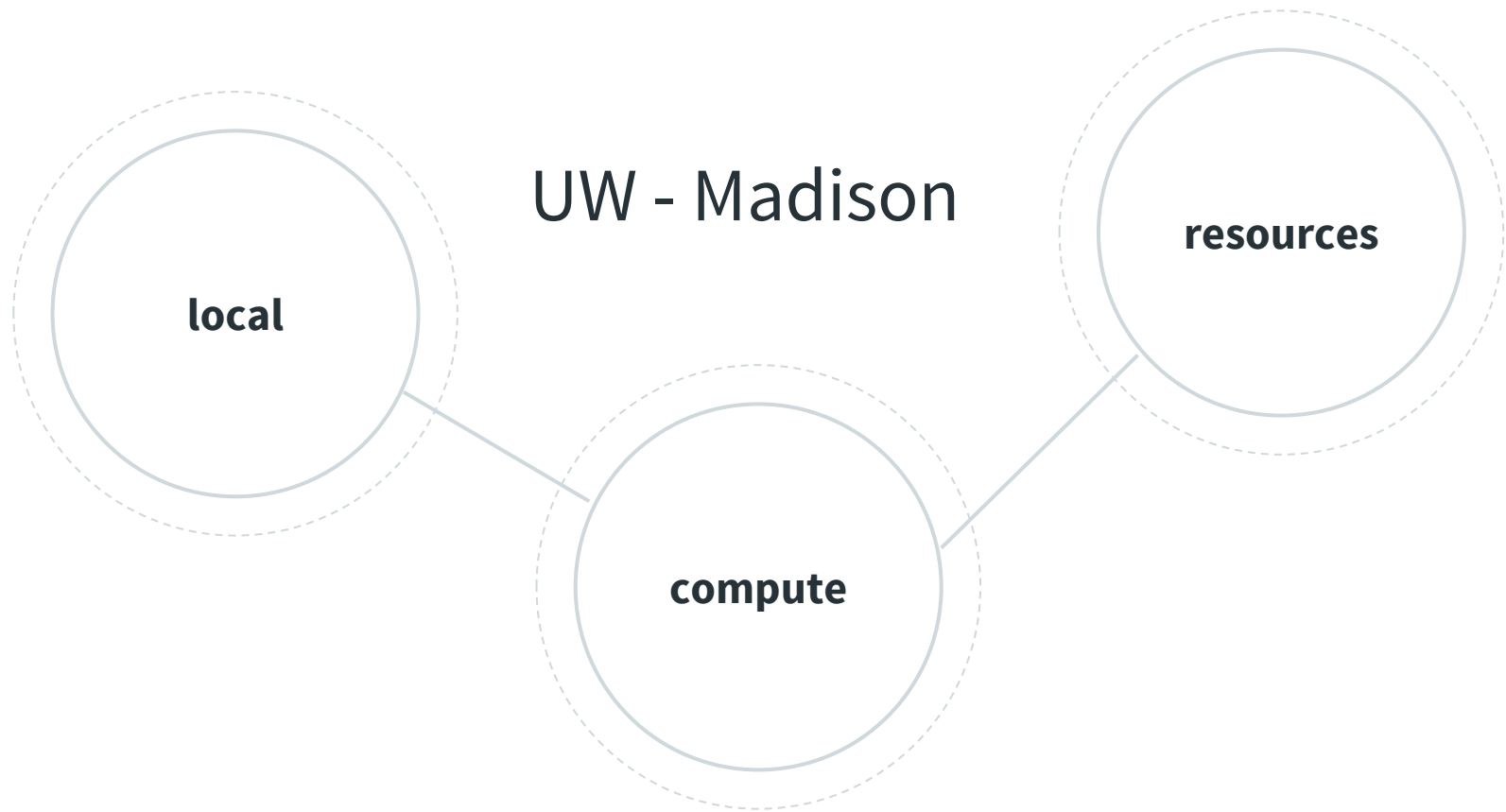
UW - Madison

local

resources

compute

Local High Throughput Computing
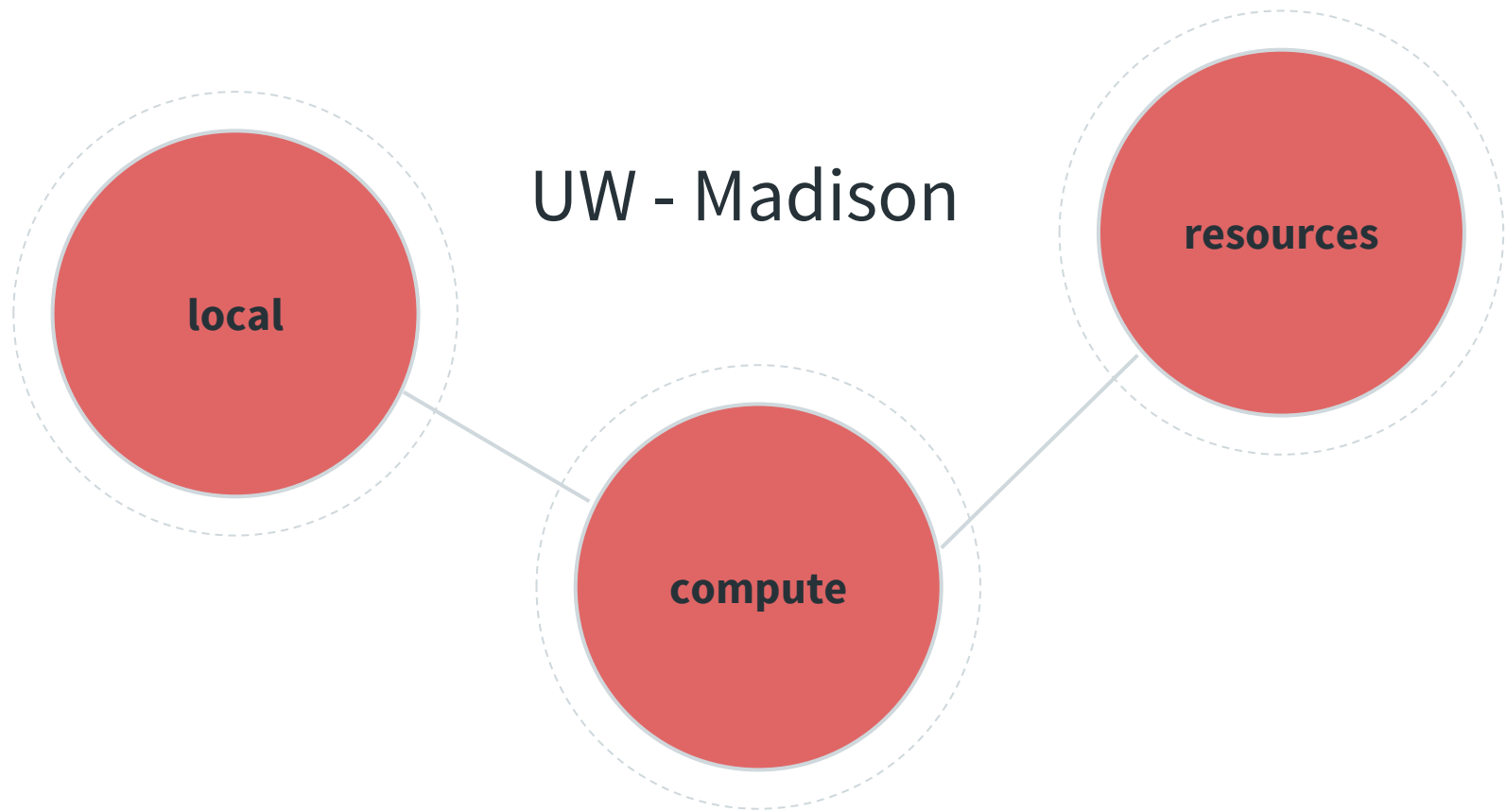
UW - Madison

local

compute

resources

How do you get more computing resources?

#1: Buy Hardware

Who doesn't love to play with new toys?

#1: Buy Hardware

◎ Costs $$$
- Initial cost
- Maintenance
- Management
- Power and cooling

◎ Takes time

◎ Rack/floor space

◎ Obsolescence

◎ Plan for peak loads, pay for all loads

#2: Use the Cloud

Everyone's favorite buzzword!

#2: Use the Cloud - Paying Per Cycle

◎ e.g. Amazon Web Services, Rackspace
◎ Fast spin-up
◎ Costs $$$
◎ Still needs expertise + management
◎ Does it fit with your university's policies?

## #2: Use the Cloud - 'Managed' Clouds

◎ e.g. Cycle Computing, Globus Genomics

◎ Pay someone to manage your cloud resources — still costs $$$

◎ Researchers have used this to great success. See HTCondor Week 2015 talks:

  ○ http://research.cs.wisc.edu/htcondor/HTCondorWeek2015/presentations/CottonB_CycleComputing.pptx

  ○ http://research.cs.wisc.edu/htcondor/HTCondorWeek2015/presentations/Madduri-CondorWeek-2015.pptx

#3: Share Resources

Sharing is caring, it can be fun!
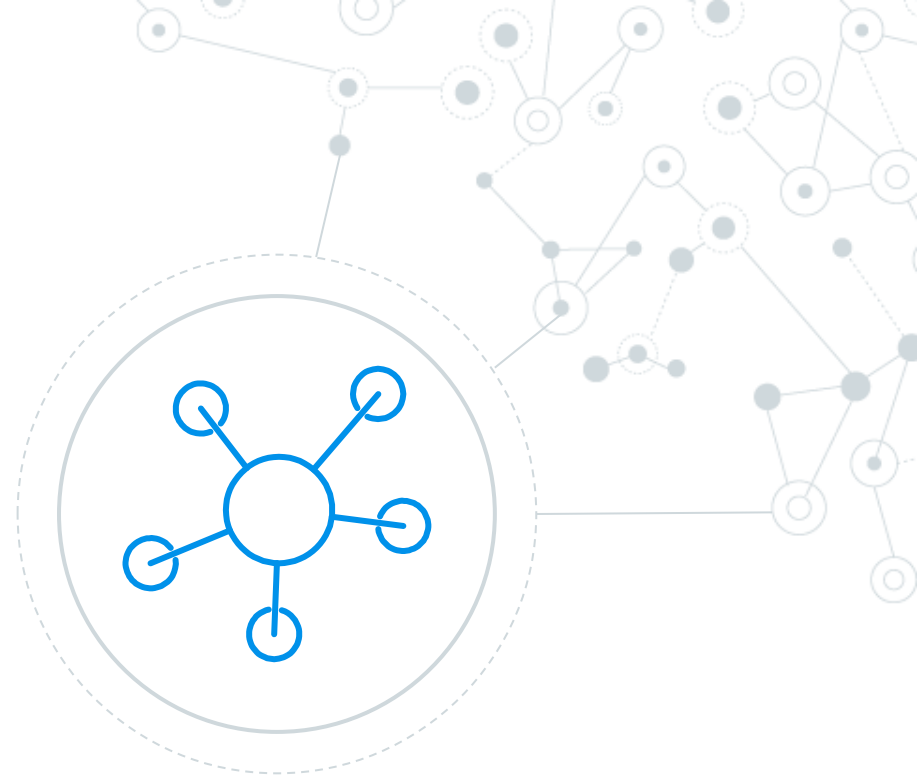
Distributed High Throughput Computing
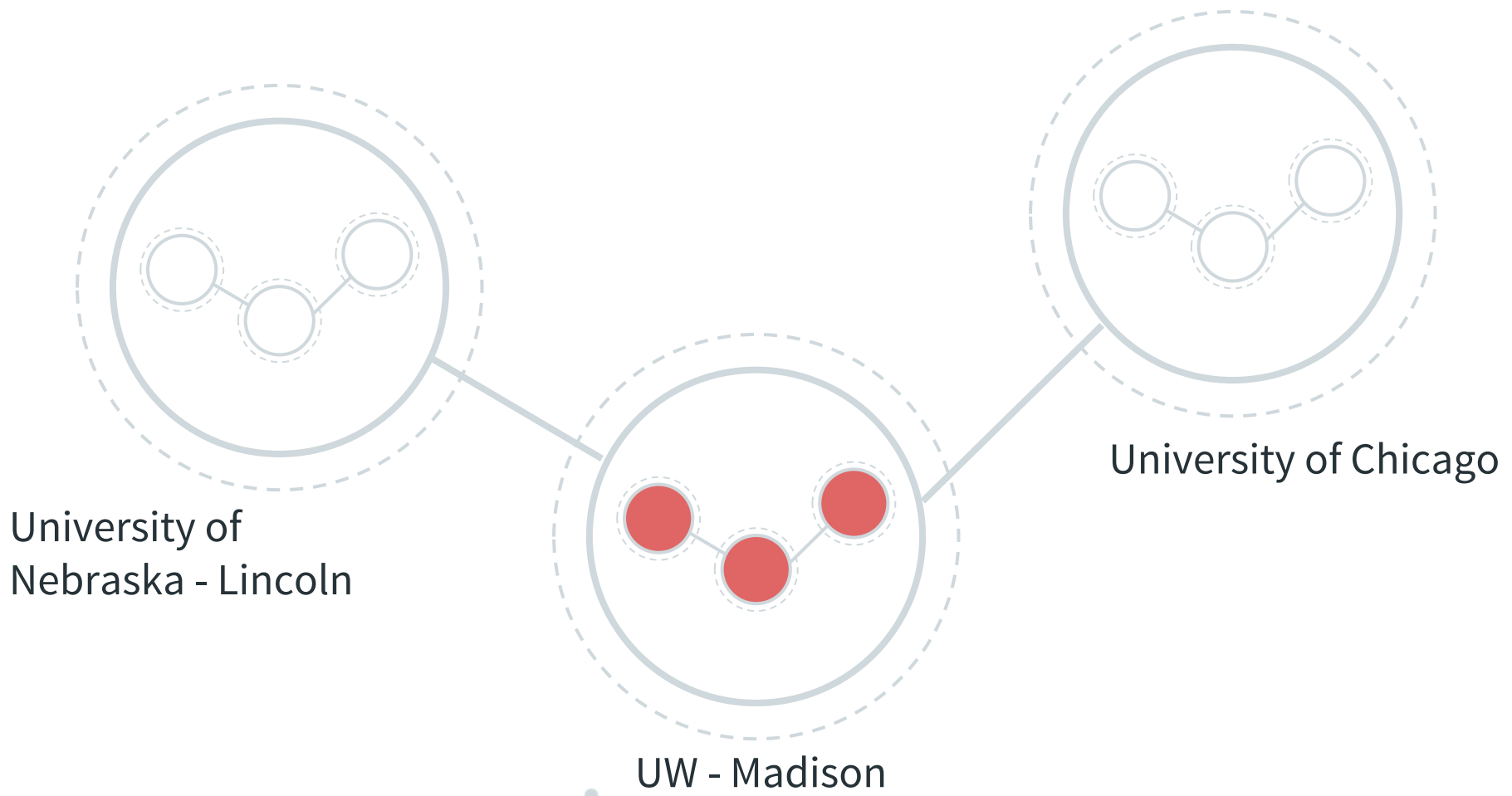
University of
Nebraska - Lincoln

UW - Madison

University of Chicago

i.

Manual Job Partitioning

Let's start sharing!

Manual Job Partitioning

◎ Obtain sharing agreements

◎ Query each site for idle resources

◎ Partition and submit jobs based on availability

Manual Job Partitioning — Shortcomings

◎ More sharing agreements = more account management

◎ Fewer sharing agreements = fewer available resources

◎ Query + partition is tedious and inaccurate

ii.

Automatic Job Partitioning

Let the computers do the work

◎ Same shortcomings as manual job partitioning

◎ Don't want to/can't share our resources

◎ Not all sites use HTCondor — other job schedulers e.g., SLURM, PBS, etc.

◎ Pools are independent

What are our requirements?

This may be harder than we originally thought...

Sharing Resource: Requirements

- ◎ Minimal account management
- ◎ No manual job partitioning
- ◎ Single pool
- ◎ Don't have to learn additional job schedulers
- ◎ Don't have to share our own resources

iii.

Overlay Systems

Let the OSG do the heavy lifting

Overlay Systems — What Are They?

◎ OSG as a resource broker — leasing
◎ Leased resources appear in a new pool
◎ Matchmaking occurs, jobs run as normal except…
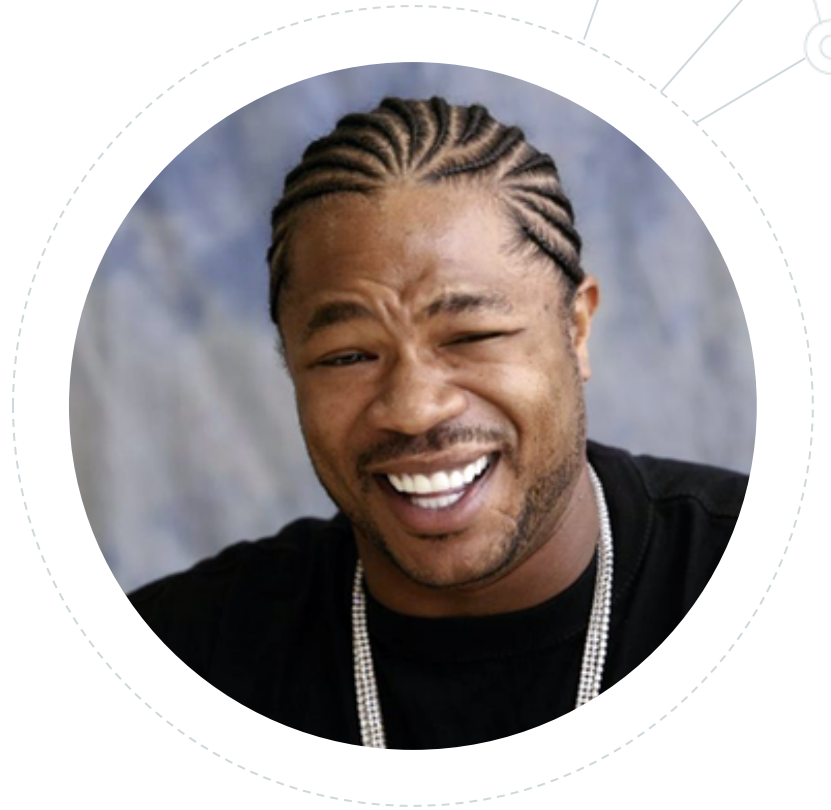 ○ The lease expires after a certain amount of time
  ○ Leases can be revoked

Overlay Systems — How Do They Work?

◎ Pilot jobs (or pilots) are just jobs
◎ Pilots are sent to sites with idle resources
◎ Pilot payload = HTCondor Startd!
◎ Startd reports to your OSG pool

Yo Dawg, I Heard You Like Jobs…

When your job runs in the OSG, it runs within a pilot job.

◎ Monitors supply (idle slots)
- ○ Site location
- ○ Site job scheduler

◎ Monitors demand (idle jobs)
- ○ Job submission location

◎ Submits pilots based on supply, demand, and site policy

Overlay Systems — Collection of Pools

- ◎ Your OSG pool is just one of many
- ◎ Separate pools for each [virtual] organization (VO)
- ◎ You will be part of the OSG VO

Overlay Systems — Leasing the Cloud

◎ What if there aren't enough idle resources?

◎ Combine overlay system with cloud technology

◎ Solutions in the works but not production ready

◎ Expect some of your jobs to run in the cloud in the next few years

Credits

Special thanks to all the people who made and released these awesome resources for free:

◎ Presentation template by SlidesCarnival

◎ Photographs by Unsplash & Death to the Stock Photo (license)

Thanks!

Any questions?