glideinWMS
The Larger Picture

by Igor Sfiligoi, Jeff Dost (UCSD)

Why this talk?

This talk introduces
glideinWMS and key concepts to help understand
the system in a high level context

The basics

- glideinWMS has been designed to address the needs of **High Throughput Computing (HTC)**
  - Better known as **batch processing**
- In a nutshell, we are trying to facilitate the **effective use**
  of a large number of CPUs
  by a large number of users

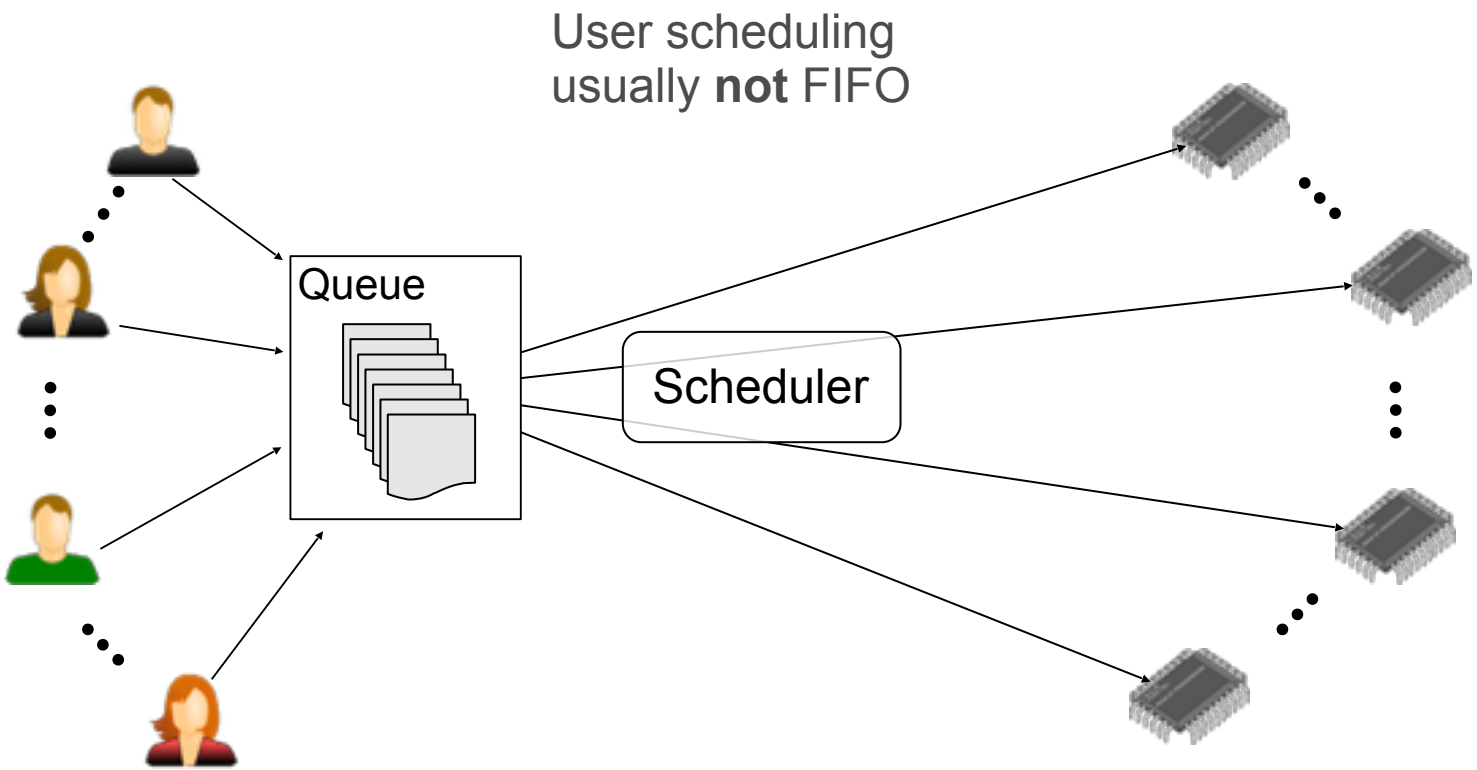
High Throughput Computing

- The basic premise of HTC is that there is **always more demand than available CPUs**
- We should make good use of those CPUs
  - **Keep them busy**, ideally, **24x7x365**
- **Sustained utilization** is thus **more important than peak performance**
  - Measure of success is
    FLOPY = Floating Points per Year
    not
    FLOPS = Floating Points per Second

HTC from the user point of view

- As a side effect, users must be HTC-aware
- There are some negative aspects
  - No interactive access, only process queuing
    - Usually referred to as **user jobs**
  - **Waiting in line to get access to CPUs**
- But the payoff is potentially huge
  - A single user can use 1000s CPUs at a time
  - **Perform computations that would take several years on a single machine in only a few days**

HTC in simplified picture



HTC products

- There are many HTC products available
  - Also known as “batch systems”
- A non exhaustive list:
  - HTCondor
  - PBS, with variants like Torque/Maui
  - LSF
  - SGE, also known as Oracle Grid Engine

Why another system?

- All of the mentioned HTC systems **assume full control** of compute resources (i.e. CPUs)
- glideinWMS developed to support **non-dedicated use** of compute resources
  - when CPUs are given to the system only for **limited duration at a time**

Non-dedicated resources

- In the past decade, two paradigms emerged
  - Grid computing
  - Cloud computing
- Both allow a user community to use compute resources they don't own
  - Often called **resource elasticity**
- Managing large number of Grid and Cloud resources by hand is impractical
- **glideinWMS automates the creation of an HTC system on top of Grid and Cloud resources**

Cloud vs Grid
(a short summary)

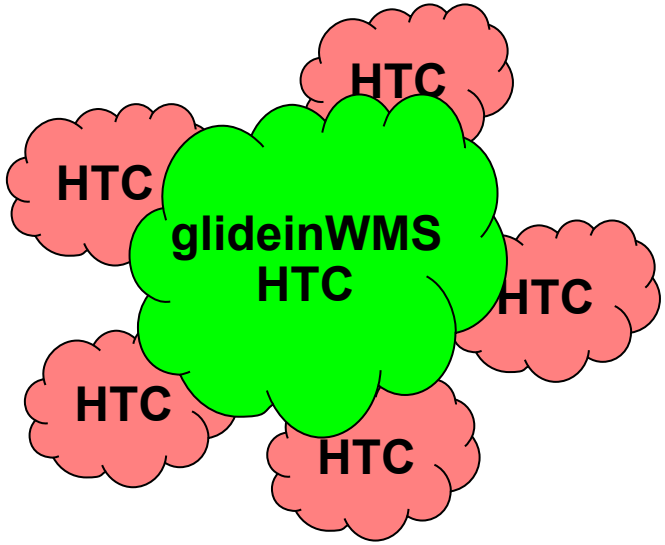
- (Commercial) Clouds are about leasing resources on a pay-as-you-go basis
- Provide Virtual Machine
- Instances expected to start almost immediately
  - (in on-demand case)

- Grid computing is a federation of HTC clusters
  - AKA **Distributed HTC**
- Provide bare metal CPU
- Job queuing is a native paradigm

glideinWMS and the Grid

(Cloud resources are used in a similar way)

- glideinWMS creates an **overlay system** on top of various **HTC clusters**
  - From the user community point of view, looks like a **single HTC system**
  - **dynamic** - size can grow and shrink based on demand



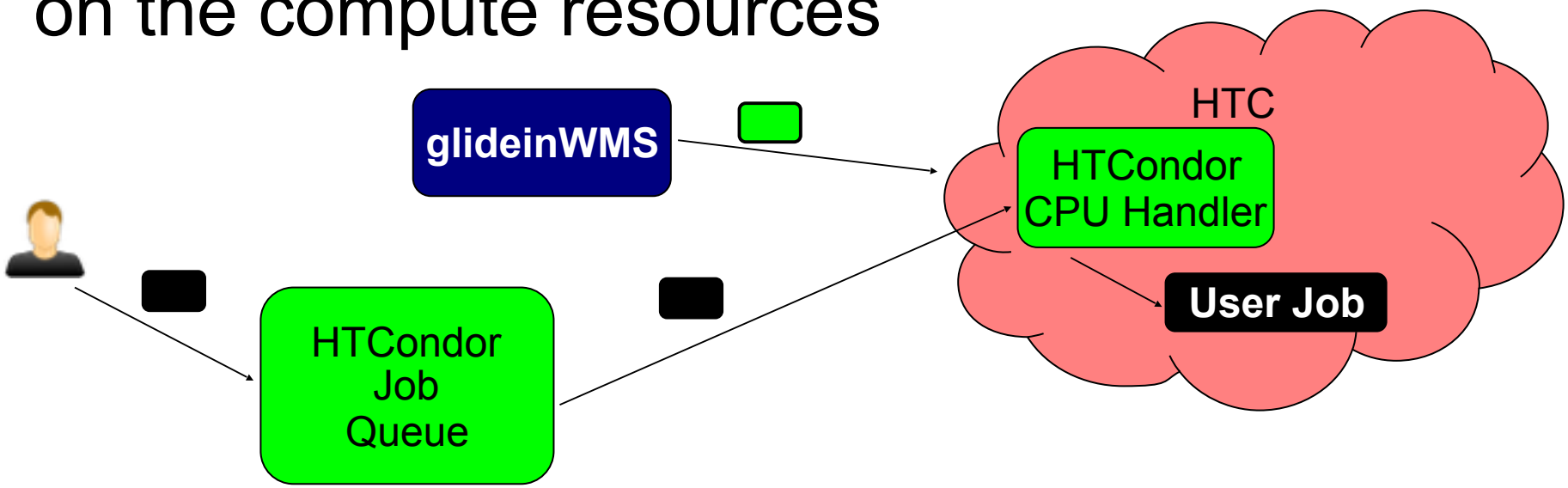
- glideinWMS **automates** the creation and maintenance of the overlay

Implementation and support

- glideinWMS is heavily based on HTCondor
- Most of the software support thus comes from the HTCondor development team
  - At University of Wisconsin – Madison
    http://research.cs.wisc.edu/htcondor/
- The glideinWMS-specific layer supported by a team based in Fermilab
  http://glideinwms.fnal.gov

glideinWMS and HTCondor

- HTCondor handles the overlay batch system
  - Most HTCondor features thus available
- glideinWMS role is limited to scheduling, configuring and starting the Condor process on the compute resources



Summary

- glideinWMS is an HTC product that enables effective use of a large number of CPUs by a large number of users
- glideinWMS creates an HTC system out of non-dedicated compute resources
  - e.g. Grid and Cloud resources
- glideinWMS is heavily based on HTCondor
  - thus benefits from HTCondor team support

Pointers

- glideinWMS development team is reachable at glideinwms-support@fnal.gov
- The official project Web page is http://glideinwms.fnal.gov
- OSG glidein factory at UCSD

  http://www.t2.ucsd.edu/twiki2/bin/view/UCSDTier2/OSGgfactory
  http://gfactory-1.t2.ucsd.edu/factory/monitor/

Acknowledgments

- This document was sponsored by grants from the US NSF and US DOE,
  and by the UC system