# Introduction to HTC

From the CMS VO point of view

by Igor Sfiligoi (UCSD)

# The problem(s)

- Users have many jobs that must be run
  - Each user has multiple tasks at once
- There are many CPUs to run them on
  - But they are not all the same
- **How do we schedule them to get the results in the shortest amount of time?**
  - Assuming one result per task
- **How do we treat all users in a fair way?**
  - Independently of how many jobs they submit

# The HTC Paradigm

- The solution is
  **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**

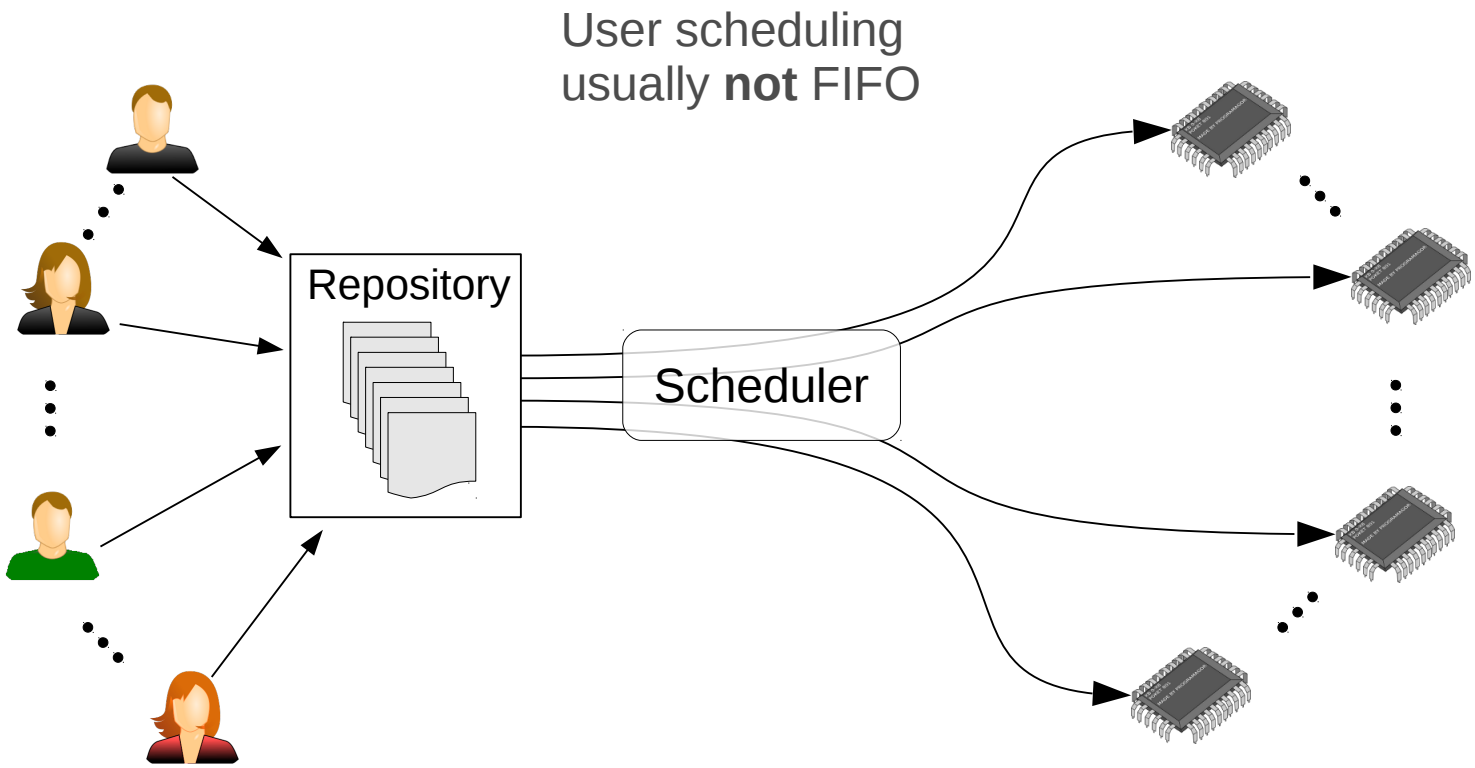
# High Throughput Computing

- 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
  - **Performing in a few hours computations that would take over a year on a single machine**

# HTC in simplified picture

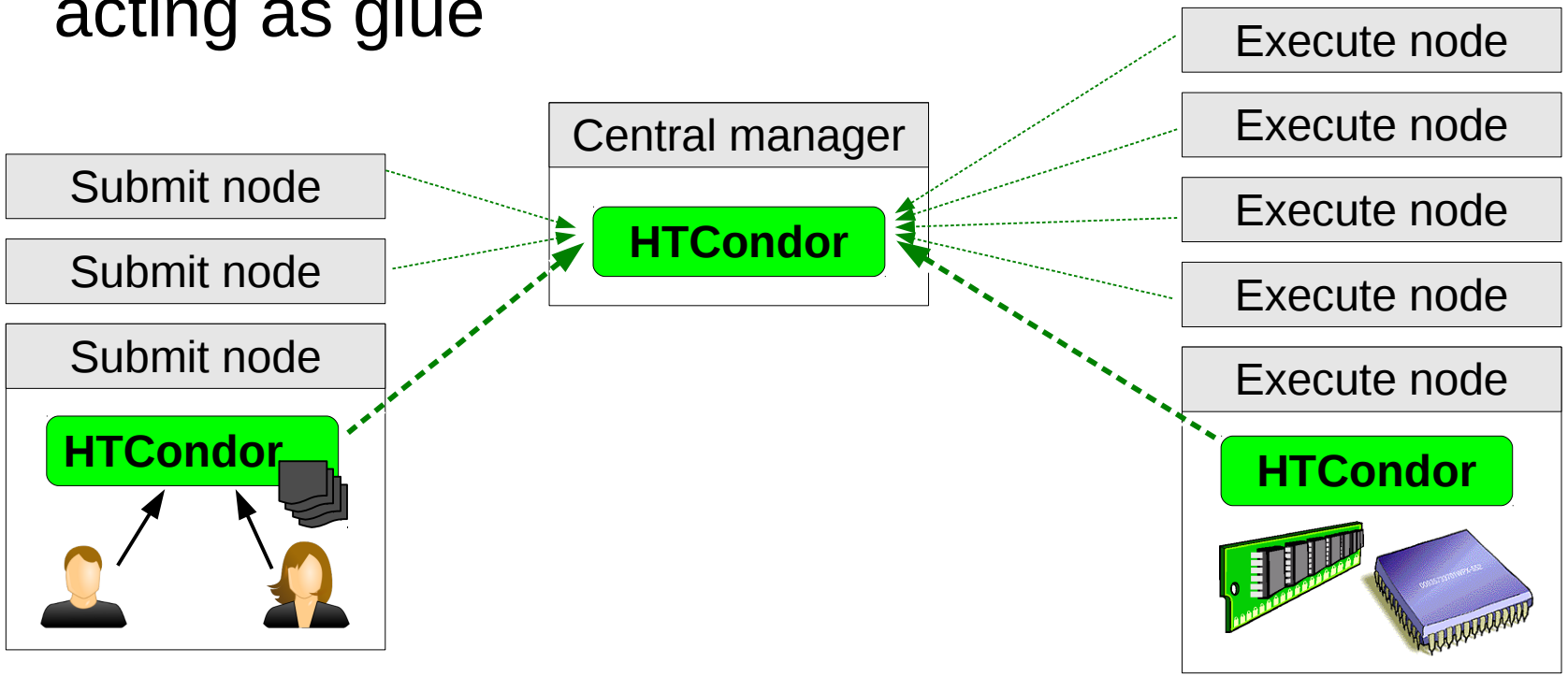


# HTC products

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

# CMS HTC based on HTCondor

- CMS HTC based on HTCondor
- There can be many submit nodes
  - With a Central Manager acting as glue



# HTCondor concepts

- The actual work quanta are
  - **Jobs** – on the submit node, typically many/node
  - **Slots** – on the execute node, typically only a few/node
- While internally implemented differently, **jobs and slots are conceptually very similar**
  - Both describe a logical entity
  - Both have attributes describing it
  - Both have requirements

"**ClassAd**" in HTCondor speak

- HTCondor's main activity is to effectively **match jobs to slots**

# Matchmaking

- Jobs that are not running
  (i.e. are **“idle”** in HTCondor speak)
  will be matched against
  Slots that don't yet run anything
  (i.e. are **“Unclaimed”** in HTCondor speak)
- Requirements expressions can (and usually do) **reference attributes in the other ClassAd**
- Both **sides must evaluate to True** for a match
  - Although CMS encourages all logic
    to reside in the Slot requirements

# Matching order

- Most of the time, there are way more idle jobs than Unclaimed slots
  - So order is important
- Two policies
  1) Jobs from highest priority user first
  2) Priority-FIFO policy for jobs of the same user
- User priority based on usage
  - The more resources you use, the lower the priority (with priority recovery over time)
  - But some users may be marked as “more important” (priority multipliers and group quotas)

# File transfer

- HTCondor can also take care of file transfer between the submit node and the worker node
  - Both input and output files
- CMS jobs typically only use it for small files
  - Most of the data is read-from and written-to remotely directly from the jobs themselves, without HTCondor intervention (or knowledge)

# CMS submission model

- CMS users almost never submit directly to HTCondor
- CMS provides portal tools for job submission
  - CRAB for Physics Analysis users
  - WMAgent for generation of reconstruction and Monte Carlo workloads

# CMS HTC in numbers

- 30k-100k Slots
- 100k – 500k Jobs
- Each job typically can run on no more than 1/10$^{th}$ of Slots

- About 1400 users, about 100 active at any time

# Acknowledgments


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