

The background of the slide is a photograph of a server room. The room is filled with rows of server racks on both sides, creating a perspective that leads towards a bright light at the far end of the aisle. The lighting is a deep blue, and there are some green indicator lights visible on the left side of the racks. In the distance, two people, a man and a woman, are standing and talking. The man is holding a tablet or a small screen.

XDMoD Overview & Single Job Viewer Demo March 15 2016

Webex + conference line:
1-800-501-8979
1951250

XD Metric Service Team

- **UB:**

- Tom Furlani, Matt Jones, Steve Gallo, Bob DeLeon, Ryan Rathsum, Jeff Palmer, Tom Yearke, Joe White, Jeanette Sperhac, Abani Patra, Nikolay Simakov, Cynthia Cornelius, Martins Innus, Ben Plessinger

- **Indiana**

- Gregor von Laszewski, Fugang Wang

- **TACC**

- Bill Barth, Todd Evans, Jim Browne, Weijia Xu

- **NCAR**

- Shiquan Su

- **NSF Funding**

- TAS: OCI 1025159, SUPReMM: OCI1203560
- XMS: ACI-1445806



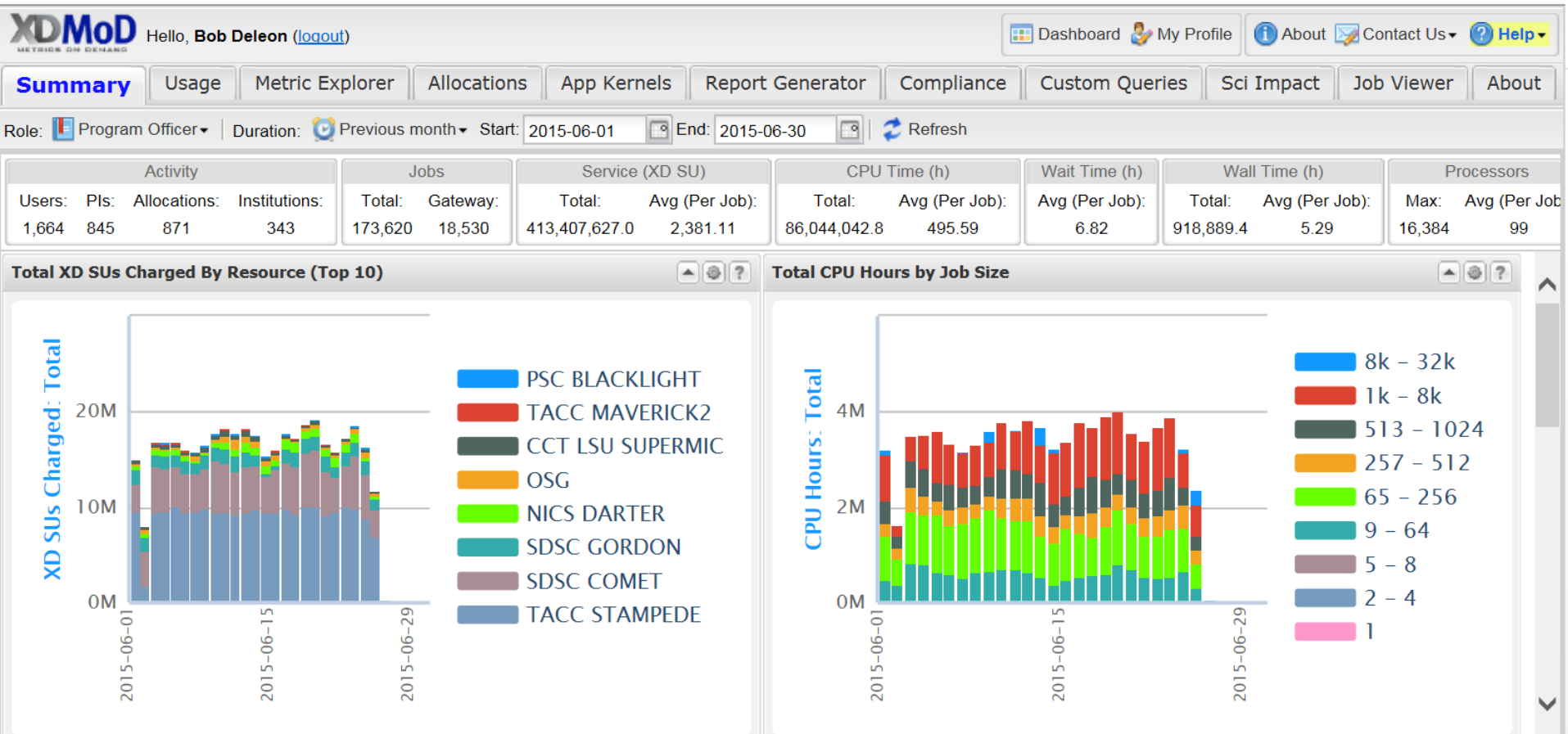
XMS: XD Metrics Service Goals

- Support comprehensive resource management for NSF's XD program, including XSEDE
 - XDMoD
- Support comprehensive resource management for HPC systems
 - Open XDMoD
- Provide audit services of XSEDE
- XDMoD/Open XDMoD designed to meet the following objectives:
 1. provide the user community with a tool to optimize their use of HPC resources,
 2. provide operational staff with ability to monitor, diagnose, and tune system performance as well as measure performance of all user jobs,
 3. provide application developers with ability to obtain detailed analysis of application performance,
 4. provide stakeholders with a tool to facilitate HPC planning and analysis,
 5. provide metrics to help measure scientific impact.



Motivation

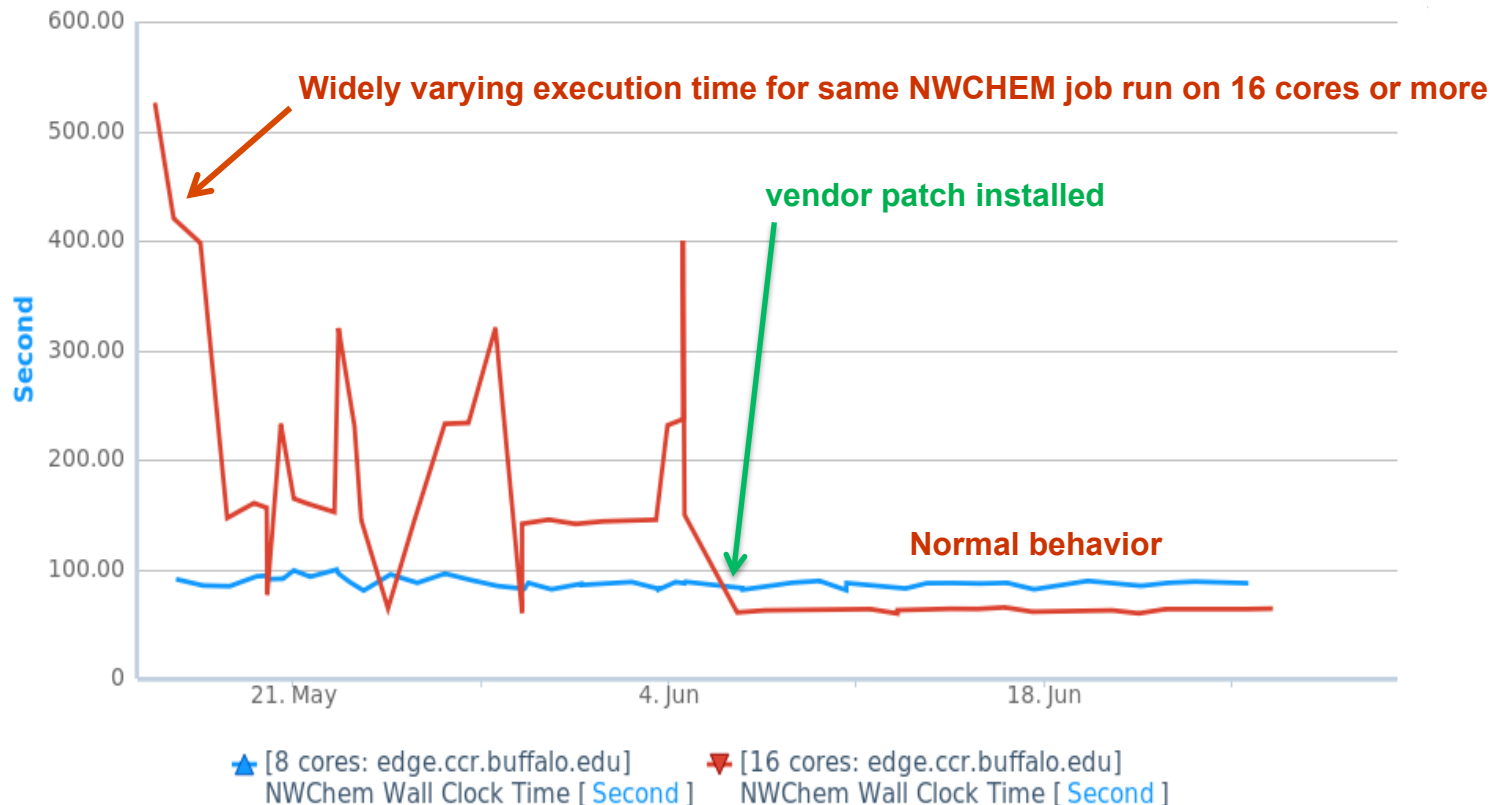
- Provide detailed operational and usage data



Motivation

- Improve User Experience
 - User shouldn't be the "canary in the coal mine" identifying problems

Example: Application Kernels Discovers Bug in Panasas File System for NWChem Code



2012-05-15 to 2012-07-01 Src: XDMoD App Kernels. Powered by XDMoD/Highcharts

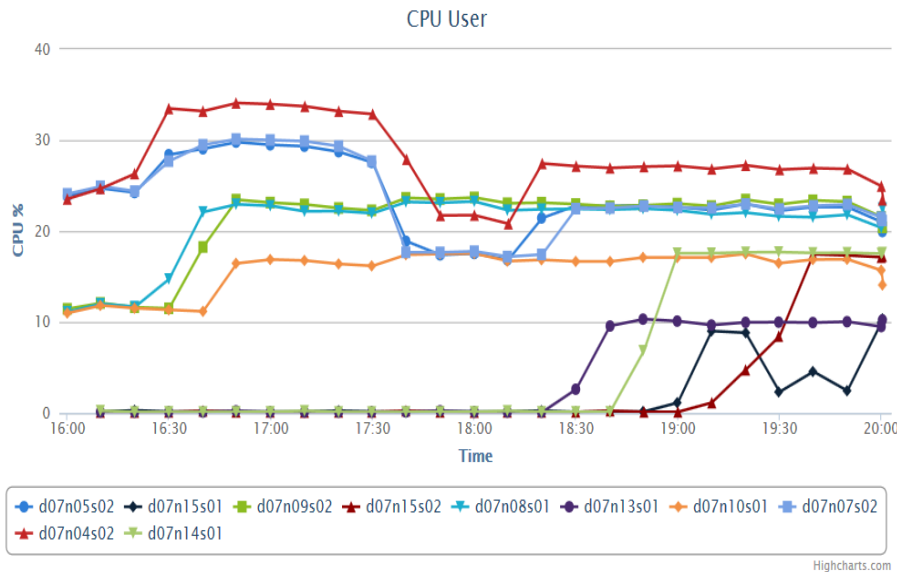


Motivation

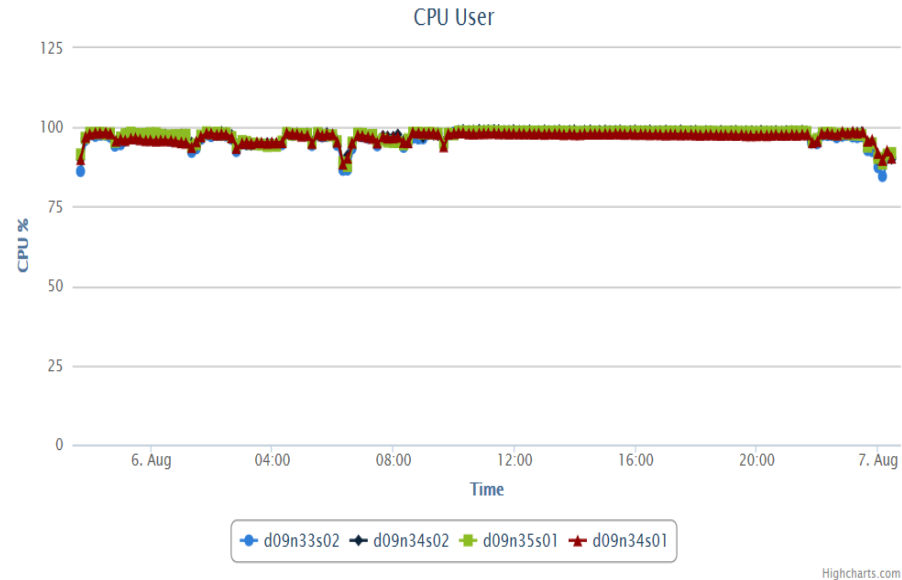
- Improve User Throughput

- Software tools to automatically identify poorly performing jobs
- Job 2552292 ran very inefficiently
- After HPC specialist user support, a similar job was vastly improved

Before CPU efficiency below 35%

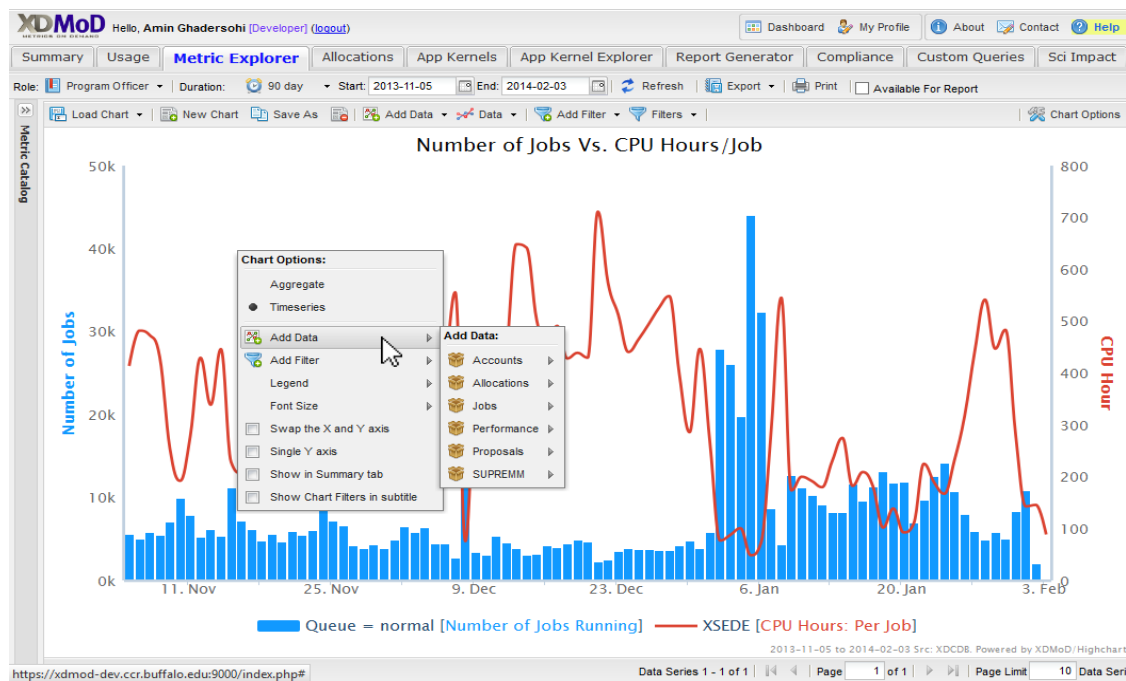


After CPU efficiency near 100%



XDMoD Portal

- Display Metrics – GUI Interface
 - Utilization, performance, publications
- Role Based: View tailored to role of user
 - Public, End user, PI, Center Director, Program Officer
- Custom Report Builder
 - Multiple File Export Capability – Excel, PDF, XML, RSS, etc
- Job Viewer Tab
 - Access detailed job level performance data



Measuring Job Level Performance

- Collaboration with Texas Advanced Computing Center
- Integration of XDMoD with Monitoring Frameworks
 - **TACC_Stats/Lariat**, Performance CoPilot, Ganglia, etc
 - Supply XDMoD with job performance data – applications run, memory, local I/O, network, file-system, and CPU usage
- **In production in XDMoD (XSEDE)**
- Available in Open XDMoD in Beta Release (HPC Centers)
- Identify poorly performing jobs (users) and applications
 - Automated process
 - Thousands of jobs run per day – not possible to manually search for poorly performing codes
 - Jobs can be flagged for:
 - Idle nodes, Node failure, High Cycles per Instruction (CPI)
 - HPC consultants can use tools to identify/diagnose problems
 - **Job viewer tab in XDMoD portal**
- User Report Card



XD METRICS SERVICE



Job Level Performance Metrics on XSEDE Resources

•XSEDE Sites

- **TACC Lonestar, Stampede, Wranger, Maverick**
 - Data flowing into XDMoD
- **SDSC Gordon & Comet**
 - Data flowing into XDMoD
- **LSU SuperMic**
 - Data flowing into XDMoD, updating to latest version
- **NICS Darter**
 - In the process of establishing data flow into XDMoD (uses Cray RUR)
- **PSC**
 - **Planned for Bridges**

•Non XSEDE Sites (Open XDMoD)

- **Purdue**
 - Installed and running on Conte cluster
- **NCAR**
 - Plan to install on system to succeed Yellowstone
 - Ganglia for host monitoring
- **Texas A&M**
 - Installing SUPReMM
- Many other installs of Open XDMoD with SUPReMM



XD METRICS SERVICE



Feedback Mechanisms

- Feature Request button in portal
- XDMoD Help
- XDMoD Users Group
- **XDMoD Roadmap** – provides feature request and shows future development (coming

The screenshot shows the XDMoD portal interface. The top navigation bar includes links for Summary, Usage, Metric Explorer, Allocations, App Kernels, Report Generator, Compliance, Custom Queries, Sci Impact, Job Viewer, and About. The main content area is titled 'XDMoD - Roadmap' and displays a grid of feature requests organized by release version. A blue arrow points from the 'Request new features' box to the 'How to request a feature' link in the 'Help' section.

Help	Release 6.0 (July 2016)	Release 6.5 (Nov 2016)	Release 7.0 (July 2017)	Release 8.0 (July 2018)	Feature Requests
How to get back to XDMoD	Add User Authorization Capability to Open XDMoD	Federated Open XDMoD	Report Storage Metrics	Scientific Impact Metrics Realm	X-Y Plots
How to request a feature	Fix XDMoD OSG reporting	Auto Detection of Poorly Performing End User Jobs	Report on HPC Center Help Desk Ticket Handling in XDMoD	Add Value Based Analytics ROI Realm	Add Comments to Plots
How to vote for a roadmap feature	Include Cloud-based Metrics in XDMoD		Histogram Plot Capability	XDMoD Data Warehouse API	
	Improve API client documentation for XDMoD		X-Y Plot Capability		
	App Kernel Scheduling GUI for Open XDMoD				
	Provide Public github repo				

Request new features

User community can vote

Comments welcome

This block shows a detailed view of a feature request card. The card title is 'Include Cloud-based Metrics in XDMoD in list Release 6.0 (July 2016)'. It includes a 'Votes' section with a '1 vote' button, a 'Description' section with text about metrics reporting for cloud-based computing, an 'Activity' section with user comments, and a list of recent actions.

Include Cloud-based Metrics in XDMoD in list Release 6.0 (July 2016)

Votes [Share and more...](#)

[1 vote](#)

Description

Metrics reporting for cloud based computing is substantially different than that for the typical cluster based computing and a variety of new types of metrics will need to be introduced. For example, VM's can run for many months and can go inactive for periods of time during a user "job" requiring interim progress reporting. In addition, this work will necessitate re-engineering the back-end database for XDMoD.

Activity

EW Ed Walker

I vote for this as being an important feature to incorporate in Release 6.

Feb 8 at 2:16 PM

BD Bob DeLeon moved this card from Feature List to Release 6.0 (July 2016)

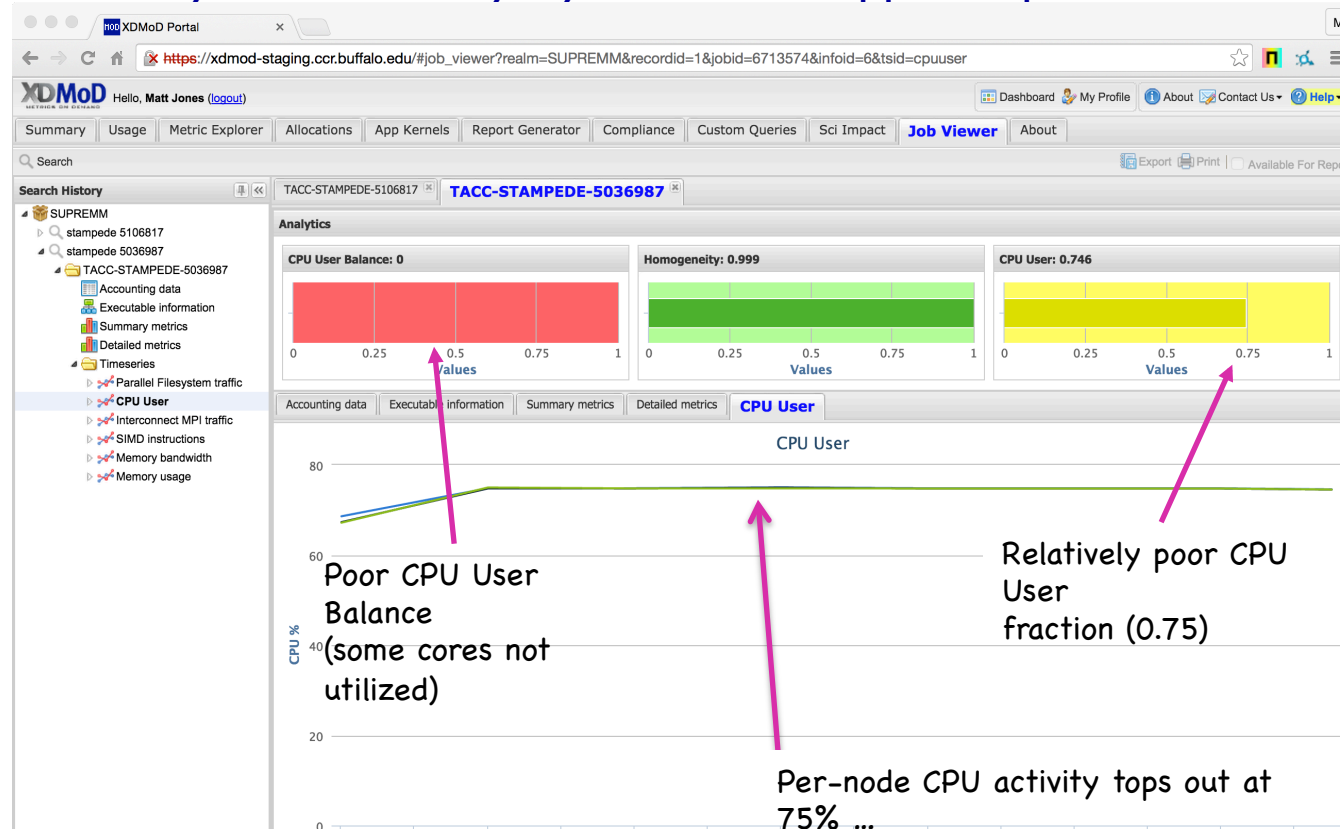
Jan 15 at 1:55 PM

TF Tom Furlani added this card to Feature List Jan 15 at 11:25 AM



Analyzing Jobs in XDMoD thru Job Viewer Tab

- Analyze the performance of individual jobs
- Basic search by job number
- Advanced search by job characteristics
- Many available metrics including charts on CPU usage, memory usage, network usage
 - Broken down by node or by CPU
- Currently utilized daily by CCR HPC support Specialist



XDMoD Job Viewer Demo

- Primarily focus on Job Viewer tab
- Analyze several XSEDE jobs
- Please feel free to ask questions



Single Job Viewer

- What is it?
 - New tab in XDMoD (first available 2015-07)
 - Displays single job metrics, both summary and time dependent:
 - CPU, memory, network, filesystem, as collected by TACC_Stats, pcp, ...
 - Includes identification and overview of potential “bad” (poor efficiency) jobs



Requirements

- XDMoD (version ≥ 5.0 , 2015-07)
 - or Open XDMoD (version ≥ 5.0 , 2015-11)
- Data collection:
 - TACC_Stats (Stampede, SuperMIC, Comet, Gordon, UB CCR)
 - Performance Co-Pilot (UB CCR)
 - Cray RUR (prototype, S. Su at NICS/Darter)
 - Ganglia (prototype, no sites yet)
 - TBD
- Collect/Manage/Ingest node level data from at least one of the supported collection mechanisms (see, e.g., previous TACC_Stats presentation)

Metrics

- Metrics gathered per node:
 - Anything available, really – cpu, i/o, memory, filesystem, network, etc.
 - Application identification – requires either Lariat/XALT or PCP/TACC_Stats with process capture. Performs matches against list of common community codes
 - Extensible – measurable quantities can be included with some development work (e.g. CUDA, MIC, panFS, gpfs, script capture, etc.)
 - Overhead: so far we have not been able to measure it compared to the variability inherent in running jobs (order of percent), but keep in mind the potential for overhead when extending metrics

Derived Metrics

- Derived metrics for job compute efficiency analysis:
 - CPU User (job length > 1h):
 - CPU user average, normalized to unity
 - CPU User balance (job length > 1h):
 - Ratio of best cpu user average to worst, normalized to unity (1 = uniform)
 - CPU Homogeneity (job length > 1h):
 - Inverse ratio of largest drop in L1 data cache rate, normalized to one (zero = inhomogeneous)
 - (graphical header currently only if all 3 available, User, User Balance, Homogeneity)
 - CPI (counter availability): clocks per instruction
 - Intel fixed counters: CLOCKS_UNHALTED_REF, INSTRUCTIONS_RETIRED
 - CPLD (counter availability):
 - clocks per L1 data cache loads (CLOCKS_UNHALTED_REF, LOAD_L1D_ALL, MEM_LOAD_RETIRED_L1D_HIT)
 - Flop/s (counter availability):
 - Varies by CPU: Intel: SIMD_DOUBLE_256, SSE_DOUBLE_ALL (SSE_DOUBLE_SCALAR, SSE_DOUBLE_PACKED)
 - (nada for Haswell – blame Intel)



Aside: XDMoD Roles

- Roles within XDMoD give access to broader/more sensitive data:
 1. NSF/XD Management
 - All data
 2. Center Director/Service Provider
 - All data at SP, public data elsewhere
 3. PI/Allocation
 - All data per allocation
 4. User
- 5. Contact us: xdmod-help@ccr.buffalo.edu if you need your role adjusted (may require approval from XD/SP/PI)

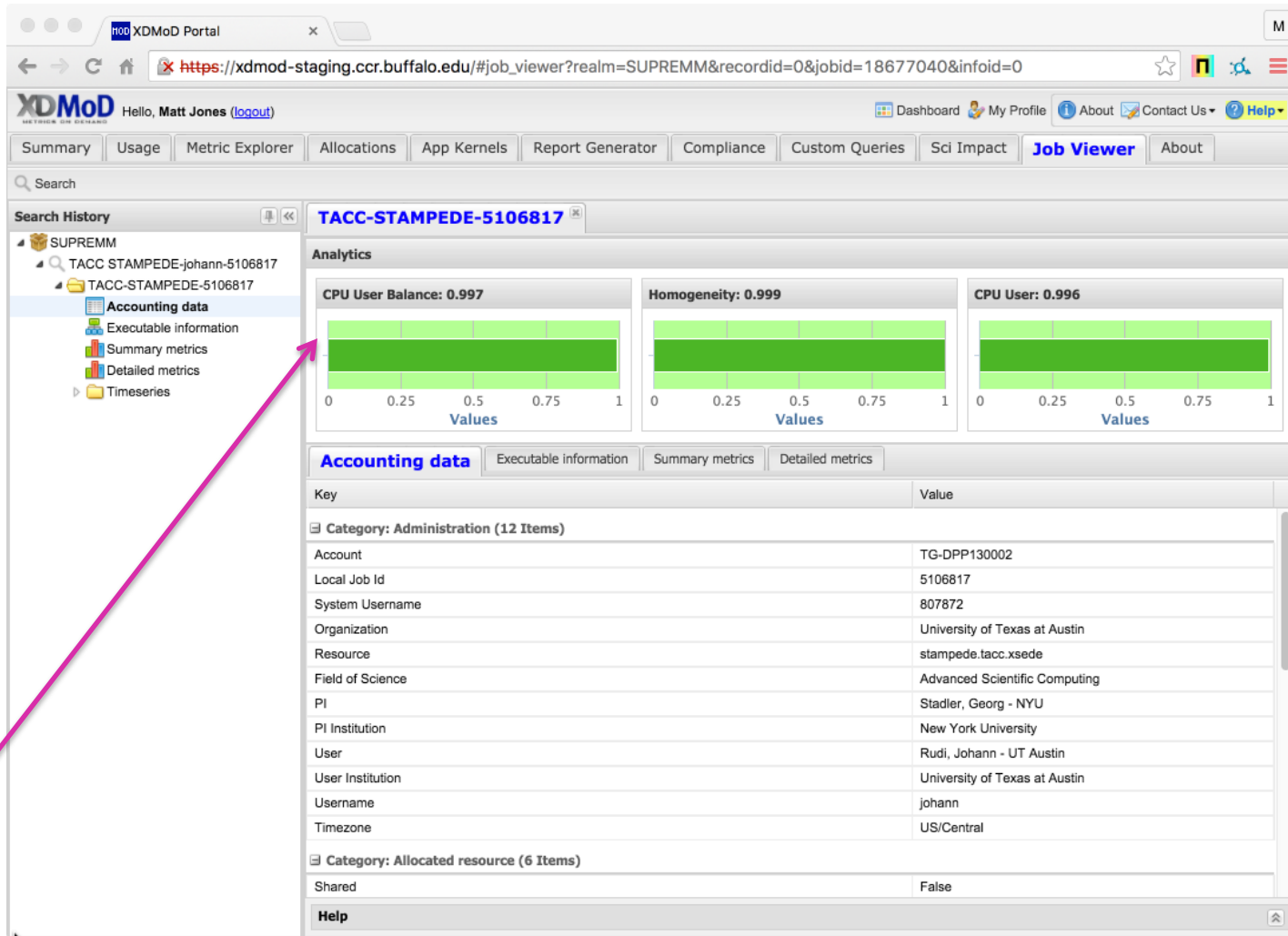


Illustrative Sample Jobs

1. 5106817 Stampede: computationally efficient large job
2. 2640365 Lonestar4: relatively inefficient job reflected in load imbalance
3. 5036987 Stampede: Cpu user cv reflects imbalance in core usage



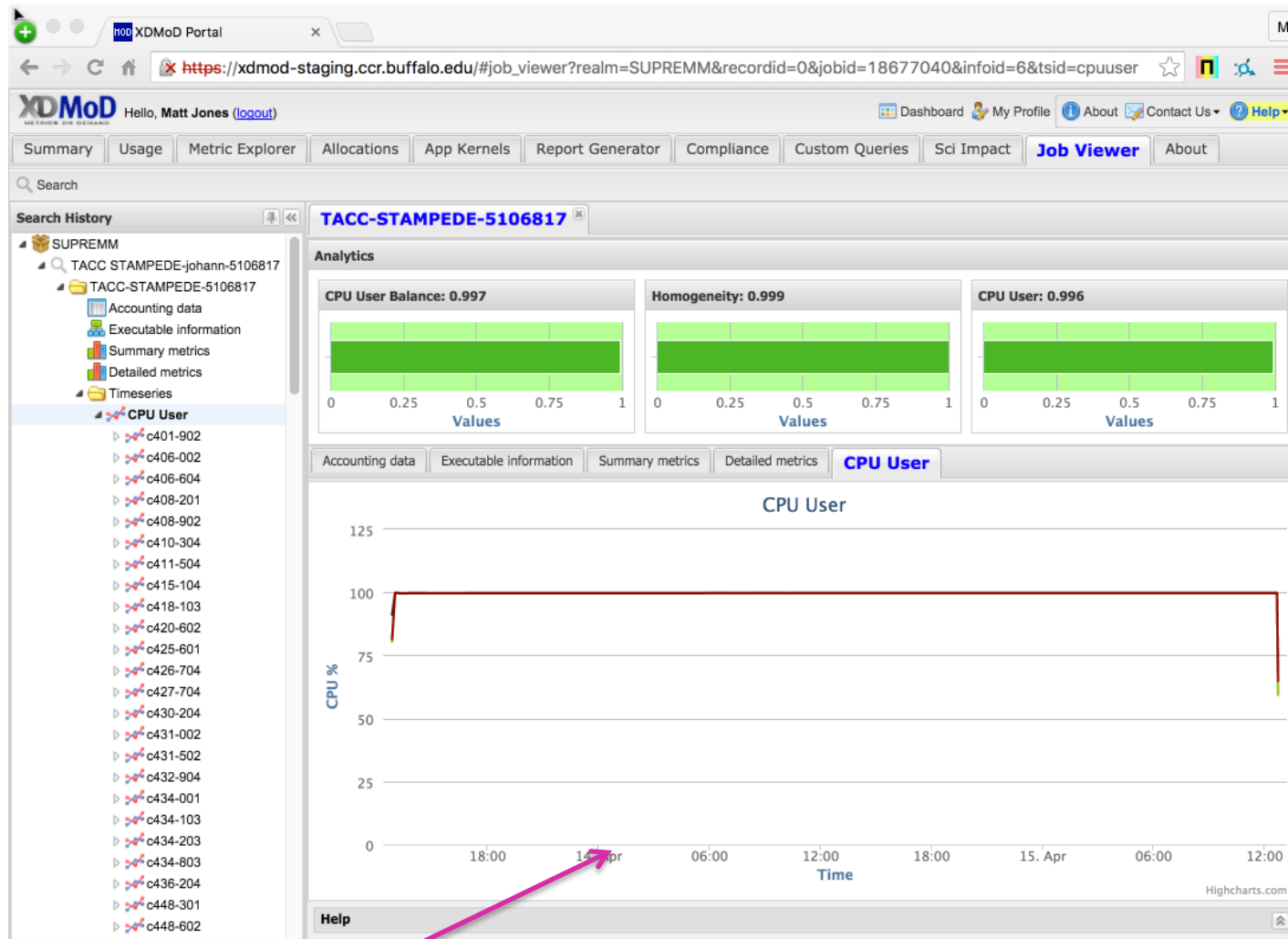
XDMoD Example 1



Graphical Header for 1st 3 derived metrics on job efficiency (only for walltime ≥ 1 h, and all included metrics present)



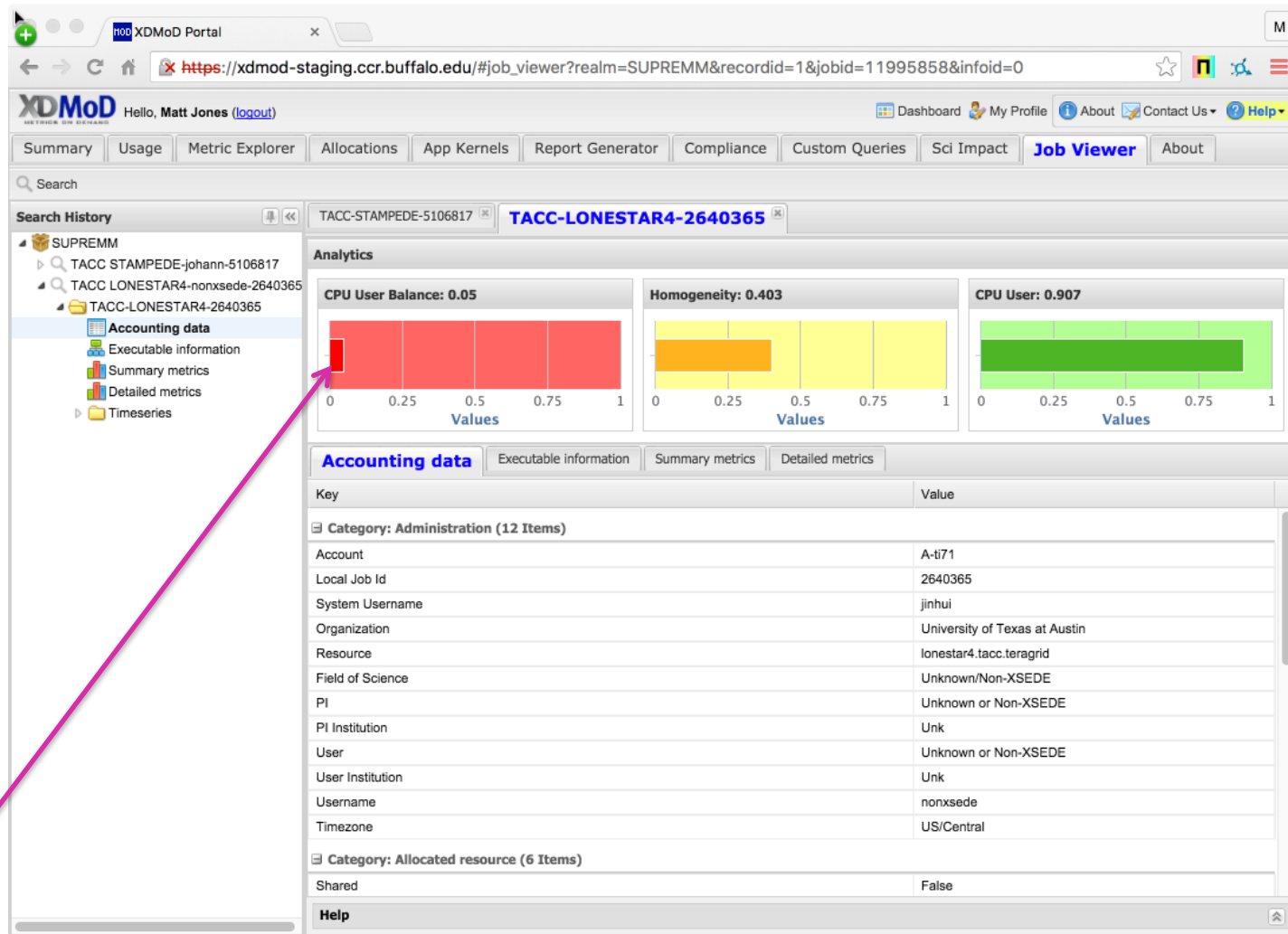
XDMoD Example 1.1



Time scale is in local browser TZ not resource TZ, will be fixed in future release



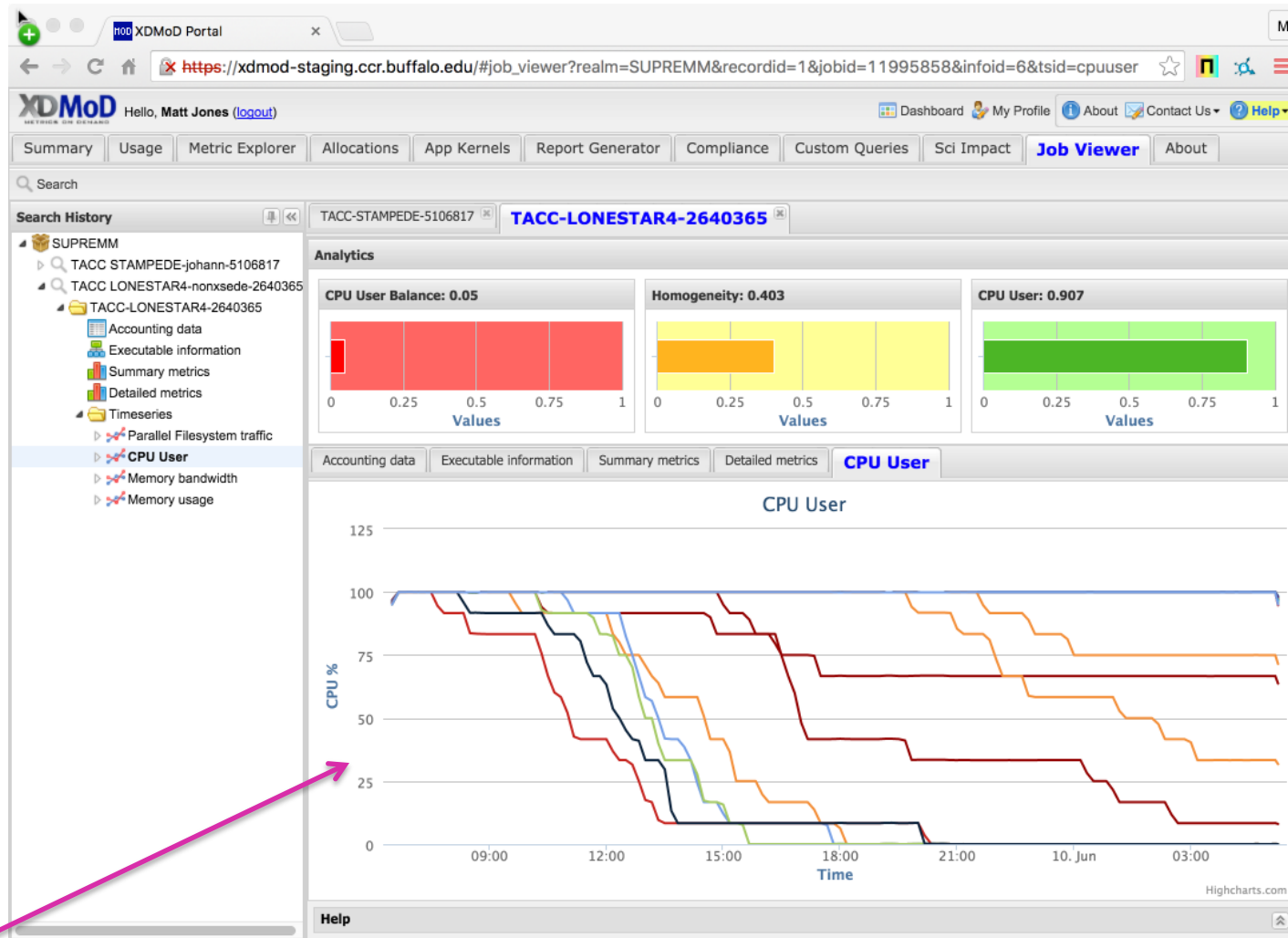
XDMoD Example 2



Job has ok overall User time, but relatively poor homogeneity and quite low User Balance



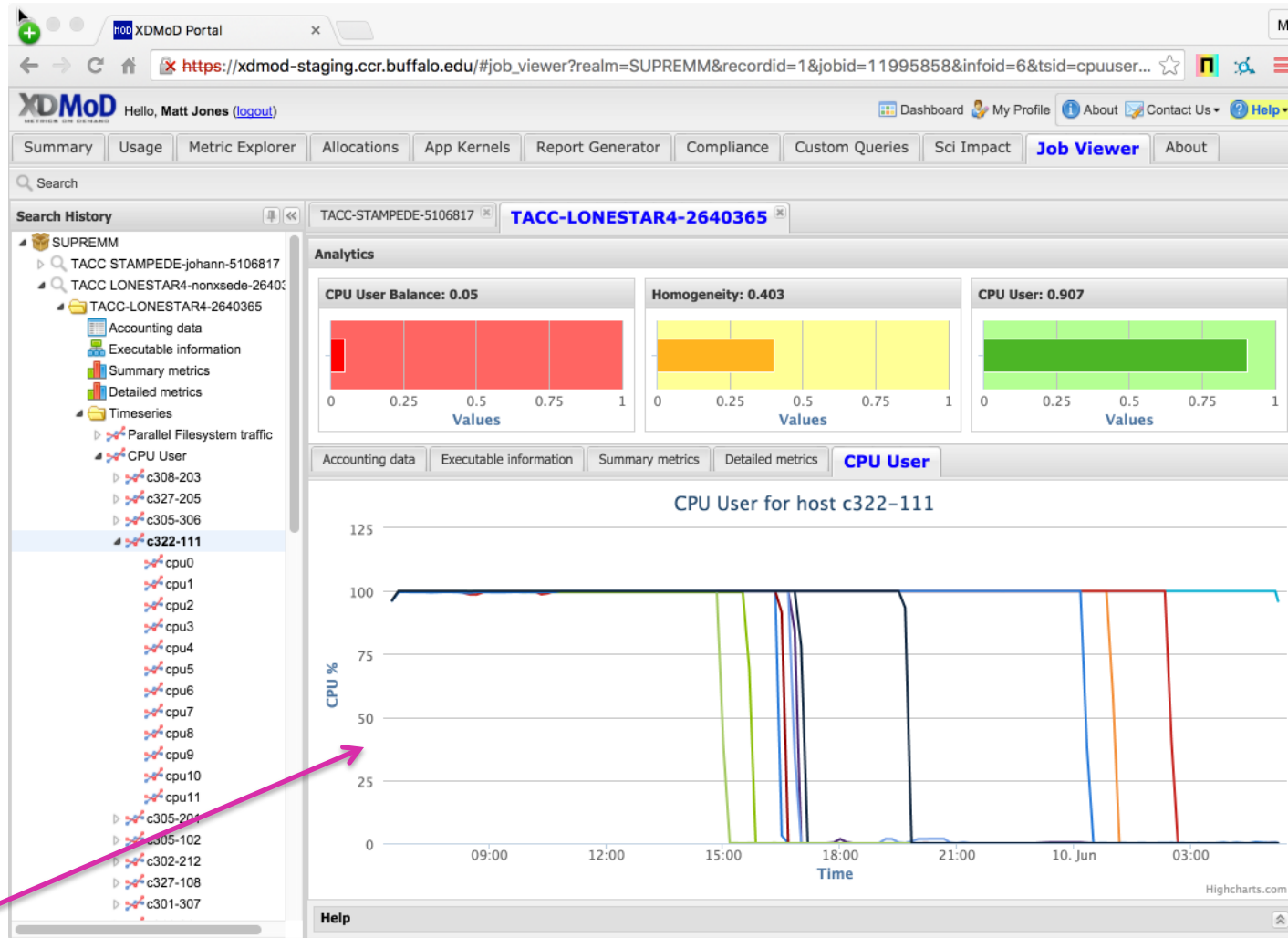
XDMoD Example 2.1



Problem: load imbalance as nodes drop CPU activity



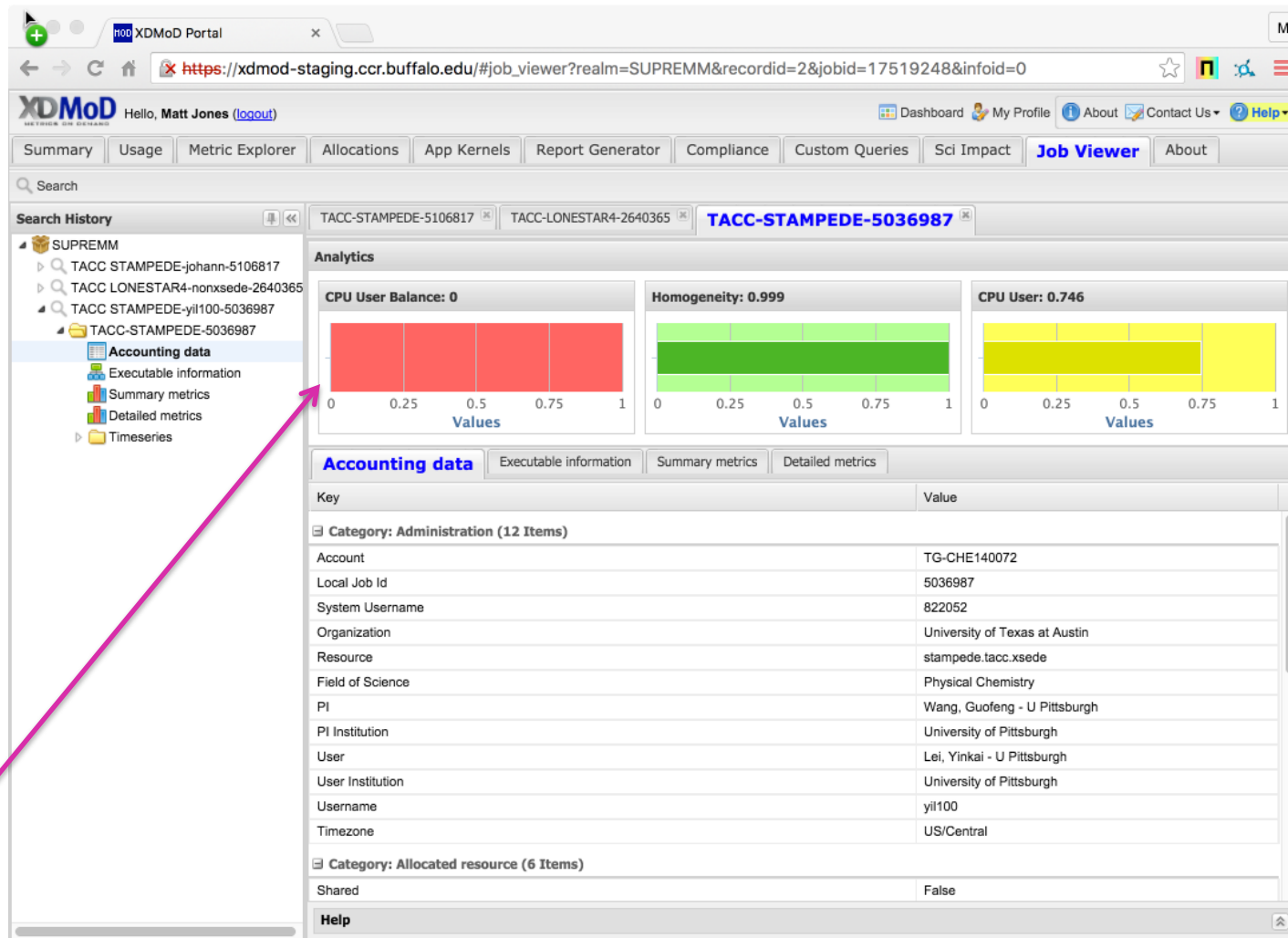
XDMoD Example 2.2



... and drilling down to nodes shows individual processes dropping out at various times (algorithmic?)



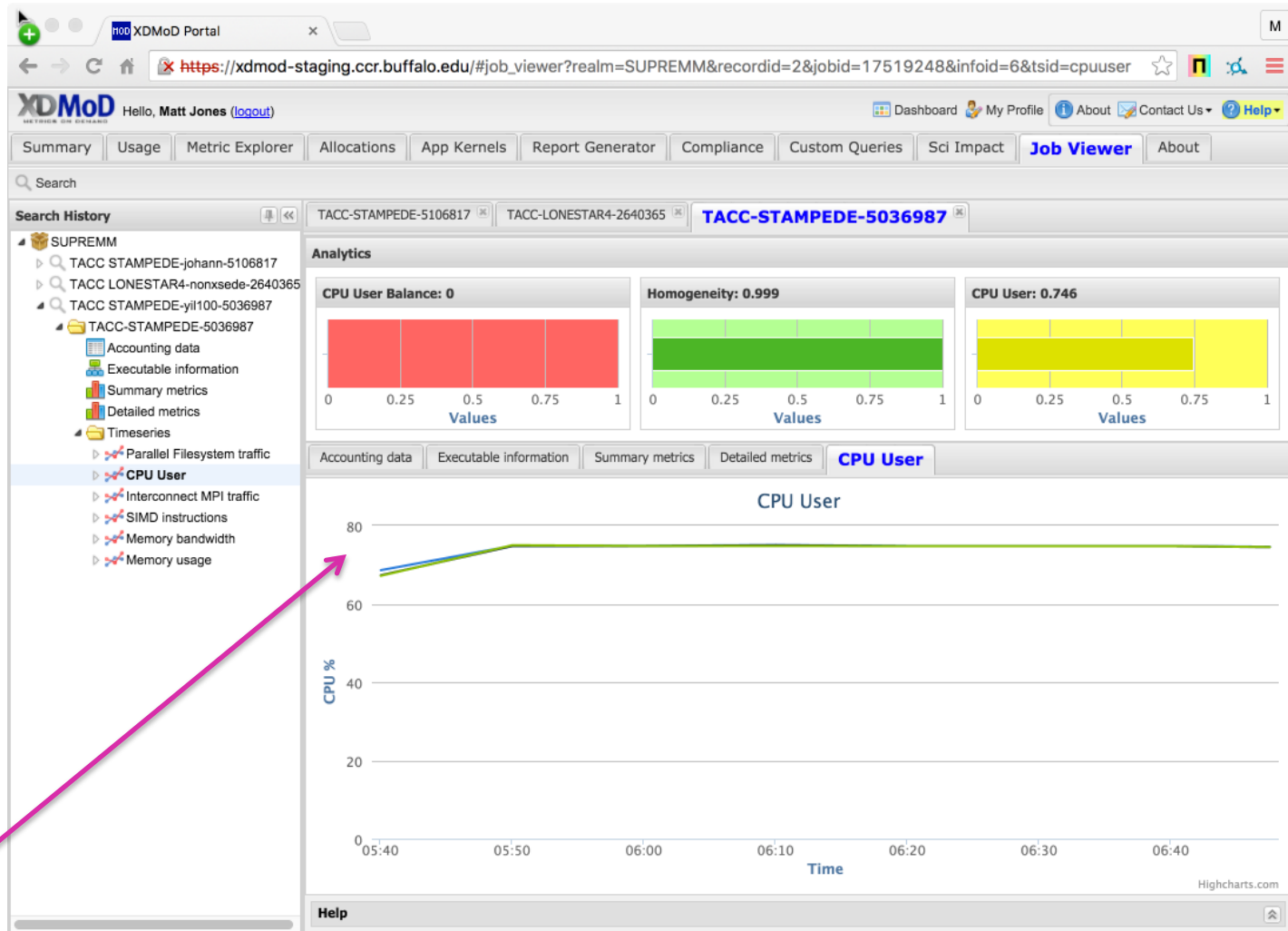
XDMoD Example 3



Relatively poor User time, poor User Balance



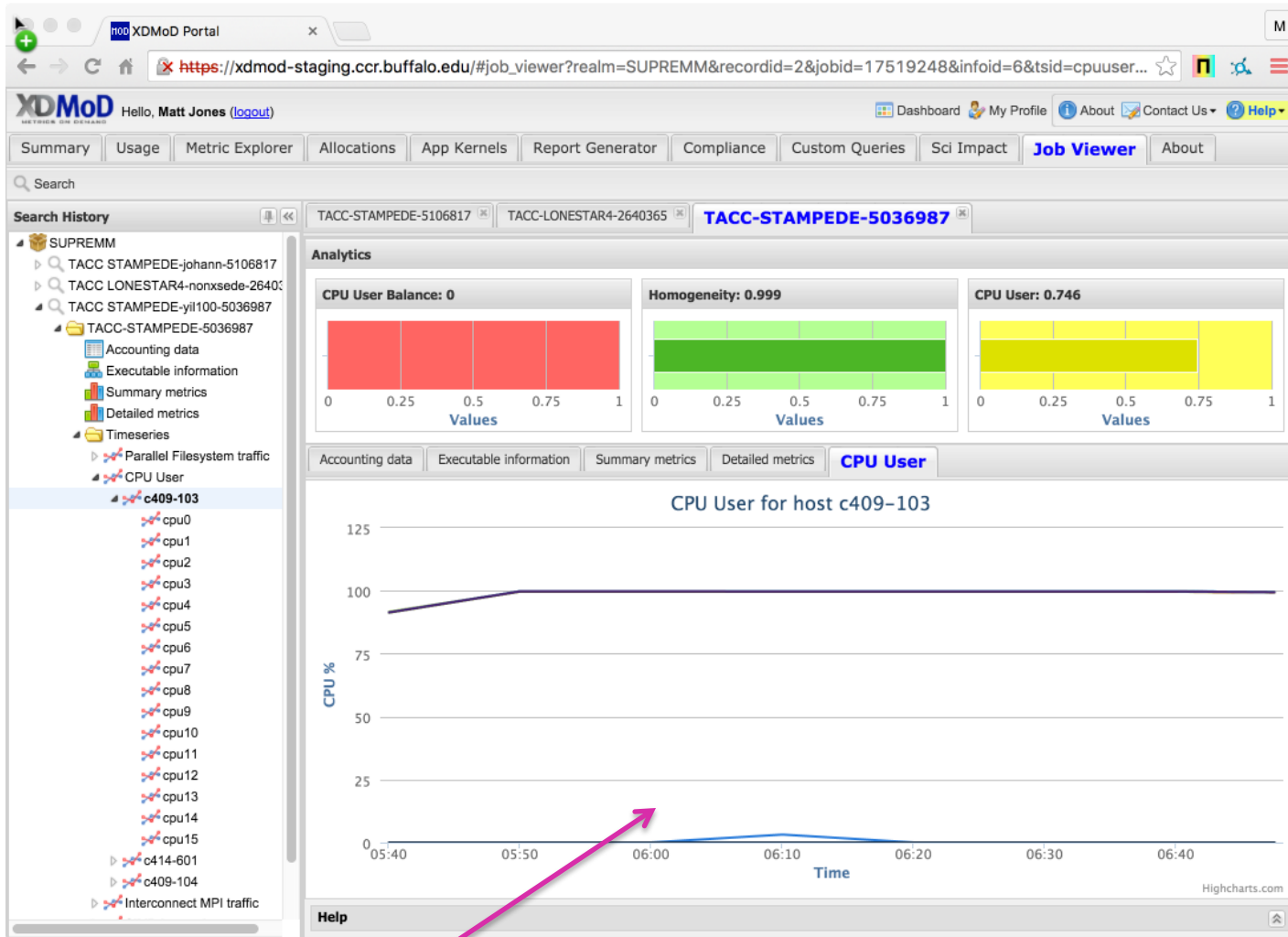
XDMoD Example 3.1



Per-node CPU activity tops out at 75% ...



XDMoD Example 3.2



Drilldown per node reveals underutilized cores (12/16) ...



What's Next?

- More thorough coverage (cloud, new XSEDE resources)
- More detailed drilldowns – especially for time dependent data
- GPU, MIC support (preliminary MIC support for Stampede)
- More search/analytics support
 - Comparison charts for jobs (combined plots)
- Export options (data, charts)
- Better node sharing support (heavily dependent on underlying collection method and limitations)
- It's a work in progress that needs user feedback!
 - xdmod-help@ccr.buffalo.edu
 - Looking for volunteers for usability study (by phone/email)



XD METRICS SERVICE



XD MoD
METRICS ON DEMAND

XDMoD Sites

<https://xdmod.ccr.buffalo.edu> [XSEDE]

<http://xdmod.sourceforge.net> [Open XDMoD]



XD METRICS SERVICE



XD MoD
METRICS ON DEMAND