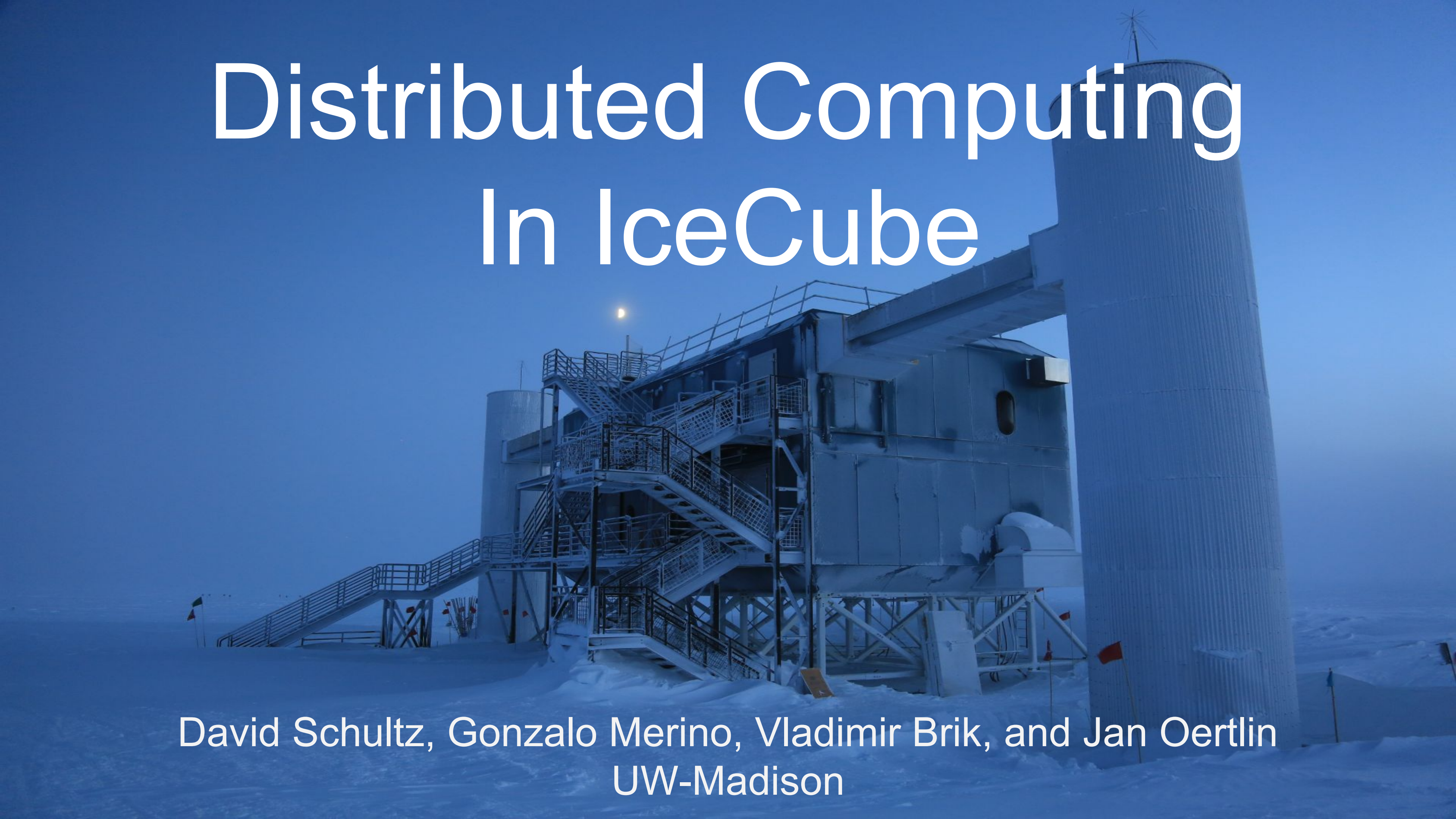
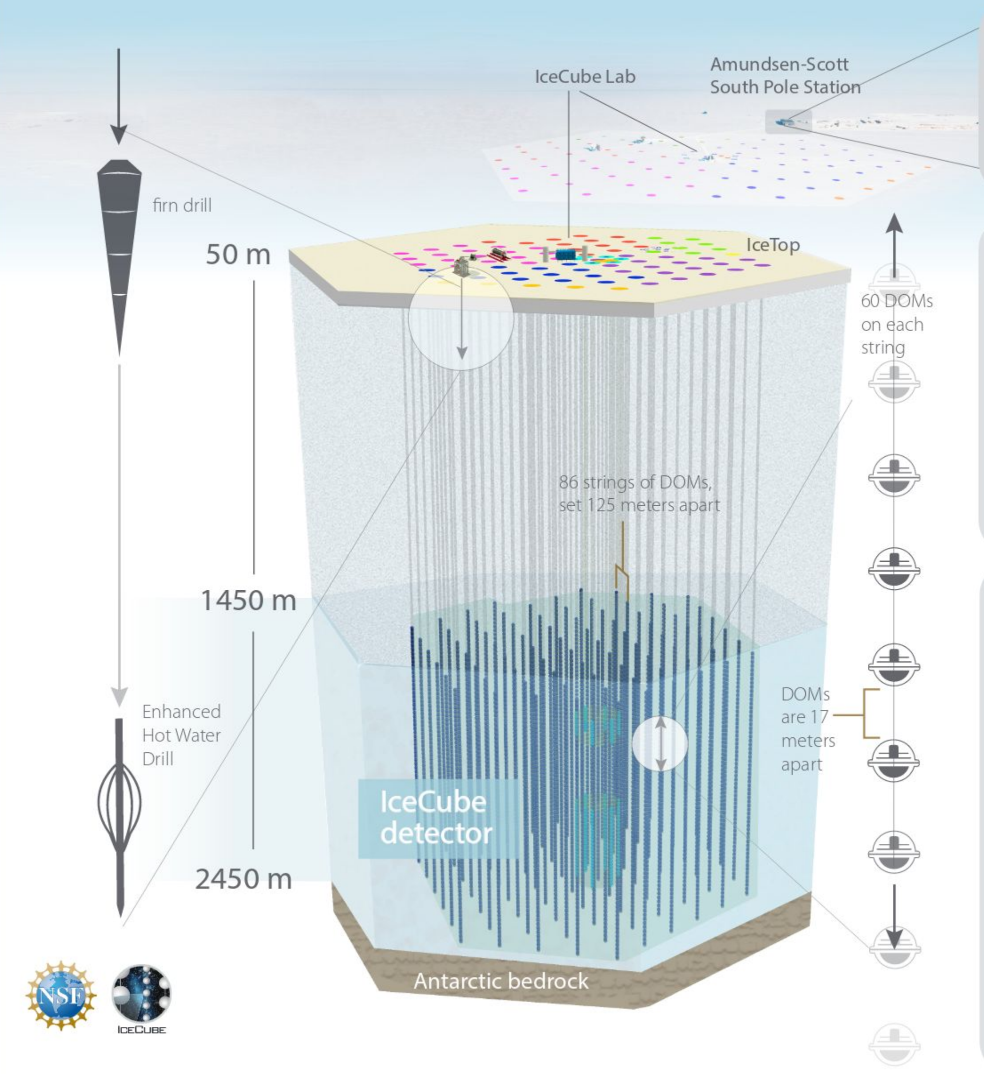


Distributed Computing In IceCube

David Schultz, Gonzalo Merino, Vladimir Brik, and Jan Oertlin
UW-Madison

The IceCube Neutrino Observatory

Design and construction

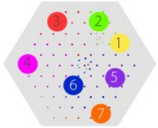


Detector Design

- 1 gigaton of instrumented ice
- 5,160 light sensors, or digital optical modules (DOMs), digitize and time-stamp signals
- 1 square kilometer surface array, IceTop, with 324 DOMs
- 2 nanosecond time resolution
- IceCube Lab (ICL) houses data processing and storage and sends 100 GB of data north by satellite daily

Detector Construction

7 seasons of construction, 2004-2011



- 28,000 person-days to complete construction, or 77 years of continuous work
- 4.7 million pounds of cargo shipped, 1.2 million of which was the drill
- 48 hours to drill and 11 hours to deploy sensors per hole
- 4.7 megawatts of drill thermal power with 200 gallons of water per minute delivered at 88 °C and1,000 psi

The IceCube Collaboration

USA
Clark Atlanta University
Drexel University
Georgia Institute of Technology
Lawrence Berkeley National Laboratory
Marquette University
Massachusetts Institute of Technology
Michigan State University
Ohio State University
Pennsylvania State University
South Dakota School of Mines & Technology
Southern University and A&M College
Stony Brook University
University of Alabama
University of Alaska Anchorage
University of California, Berkeley
University of California, Irvine
University of Delaware
University of Kansas
University of Maryland
University of Rochester
University of Texas at Arlington
University of Wisconsin–Madison
University of Wisconsin–River Falls
Yale University

Canada
SNOLAB
University of Alberta–Edmonton

University of Copenhagen, **Denmark**

Chiba University, **Japan**

Sungkyunkwan University, **Korea**

University of Oxford, **UK**

Belgium
Université Libre de Bruxelles
Universiteit Gent
Vrije Universiteit Brussel

Sweden
Stockholms universitet
Uppsala universitet

Germany
Deutsches Elektronen-Synchrotron
Friedrich-Alexander-Universität Erlangen-Nürnberg
Humboldt-Universität zu Berlin
Ruhr-Universität Bochum
RWTH Aachen
Technische Universität Dortmund
Technische Universität München
Universität Mainz
Universität Münster
Universität Wuppertal

Université de Genève, **Switzerland**

University of Adelaide, **Australia**

University of Canterbury, **New Zealand**

Funding Agencies

Fonds de la Recherche Scientifique (FRS-FNRS)
Fonds Wetenschappelijk Onderzoek-Vlaanderen (FWO-Vlaanderen)
Federal Ministry of Education & Research (BMBF)
German Research Foundation (DFG)

Deutsches Elektronen-Synchrotron (DESY)
Japan Society for the Promotion of Science (JSPS)
Knut and Alice Wallenberg Foundation
Swedish Polar Research Secretariat
The Swedish Research Council (VR)

University of Wisconsin Alumni Research Foundation (WARF)
US National Science Foundation (NSF)

Outline

- Grid History and CVMFS
- Usage / Plots
- Pyglidein
- Issues / Events:
  - High memory GPU jobs
  - Data reprocessing
  - XSEDE allocations
  - Long Term Archive

Grid History

Pre-2014 Setup

- Flock to UW
  - CHTC, HEP, CS, ...
  - GLOW VOFrontend (GLOW VO)
- IceCube simulation framework doing local submissions at ~20 sites

2014 to 2015 Setup

- Flock to UW
  - CHTC, HEP, CS, ...
  - GLOW VOFrontend (IceCube VO)
    - Some EGI, CA sites via OSG glideins
- IceCube simulation framework doing local submissions at ~10 sites

2016 Setup

- Flock to UW
  - HEP, CS, …
  - GLOW VOFrontend (IceCube VO)
    - Some EGI, CA sites via OSG glideins
- Pyglidein to all other sites
  - CHTC for better control of priorities

Sites on GLOW VOFrontend (IceCube VO)

- IceCube Sites
  - CA-Toronto
  - CA-McGill
  - Manchester
  - Brussels
  - DESY
  - Dortmund
  - Aachen
  - Wuppertal
- Notable OSG Sites
  - Fermilab
  - Nebraska
  - CIT_CMS_T2
  - SU-OG
  - MWT2
  - BNL-ATLAS

Sites on Pyglidein

- IceCube Sites
  - CA-Toronto
  - CA-Alberta
  - CA-McGill
  - Delaware
  - Tokyo
  - DESY
  - Mainz
  - Dortmund
  - Brussels
  - Uppsala
- XSEDE
  - Comet
  - Bridges
  - XStream

CVMFS

CVMFS History

- icecube.opensciencegrid.org
  - Started: 2014-08-13
  - Using OSG Stratum 1s: 2014-10-29
- Stats
  - Total file size: 300GB
  - Spool size: 45GB
  - Num files: 2.9M
- Yearly growth
  - Total file size: 120GB
  - Spool size: 10GB
  - Num files: 1.2M

CVMFS Future

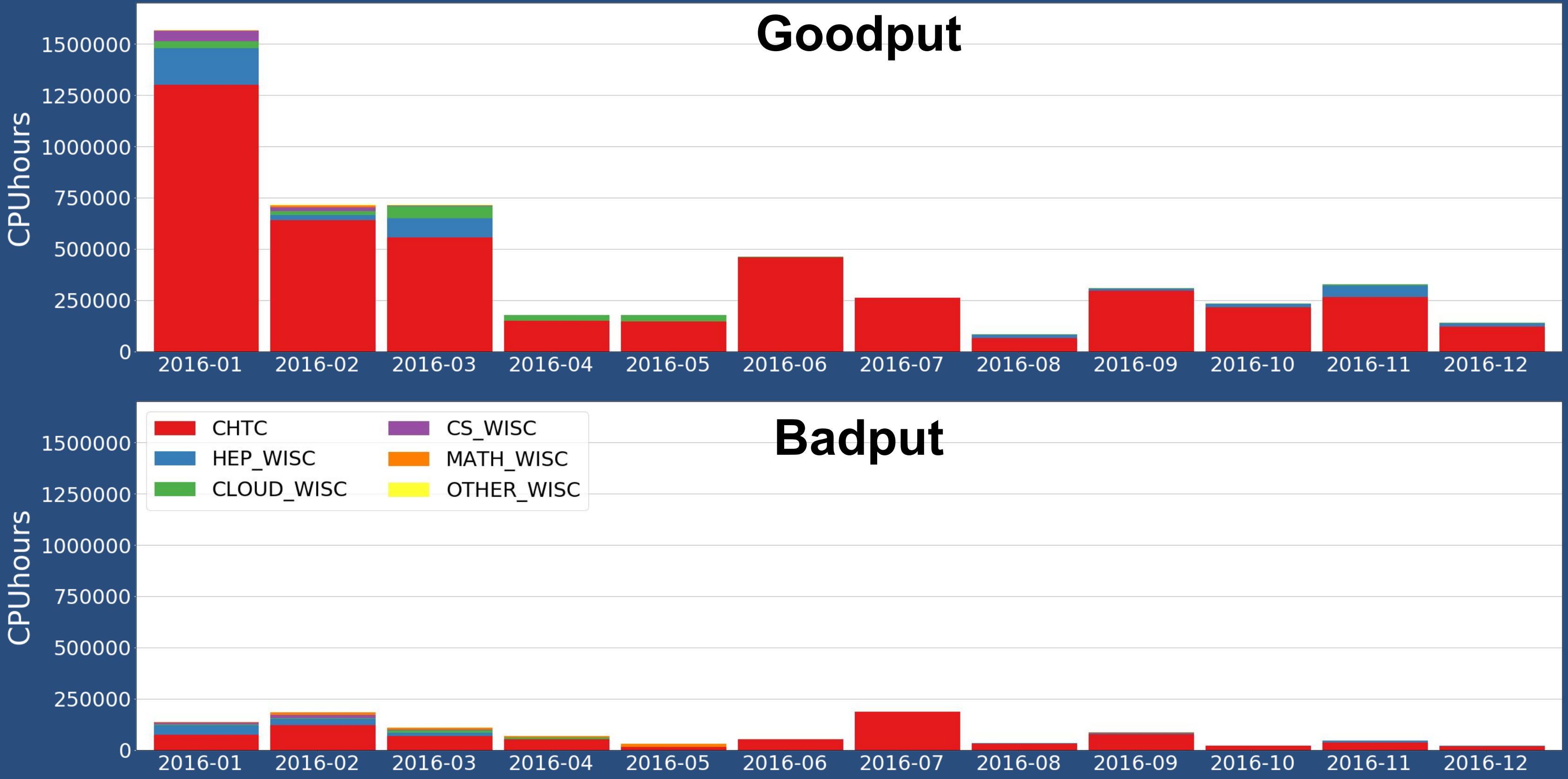
- Data federation /cvmfs/icecube.osgstorage.org?
  - Data processing and analysis: no use case
    - Most data files are single job, or small set of jobs
  - One possible use case: realtime alerts
    - Problem: they need the data instantly
    - No time for file catalog to update

CVMFS Future

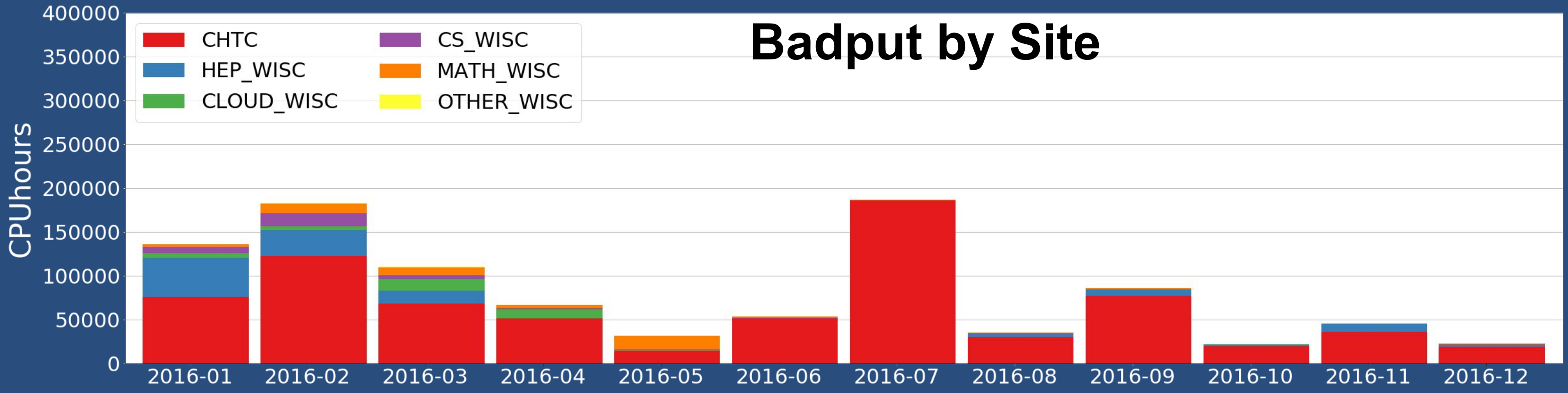
- User software distribution
  - ~300 analysis users
    - ~40 currently use the grid
  - Currently transfer ~100MB tarfiles
    - Mostly duplicates, with small additions
  - Plan: hourly rsync from user filesystem
    - Use a directory in the existing repository?
    - Make a new repository?

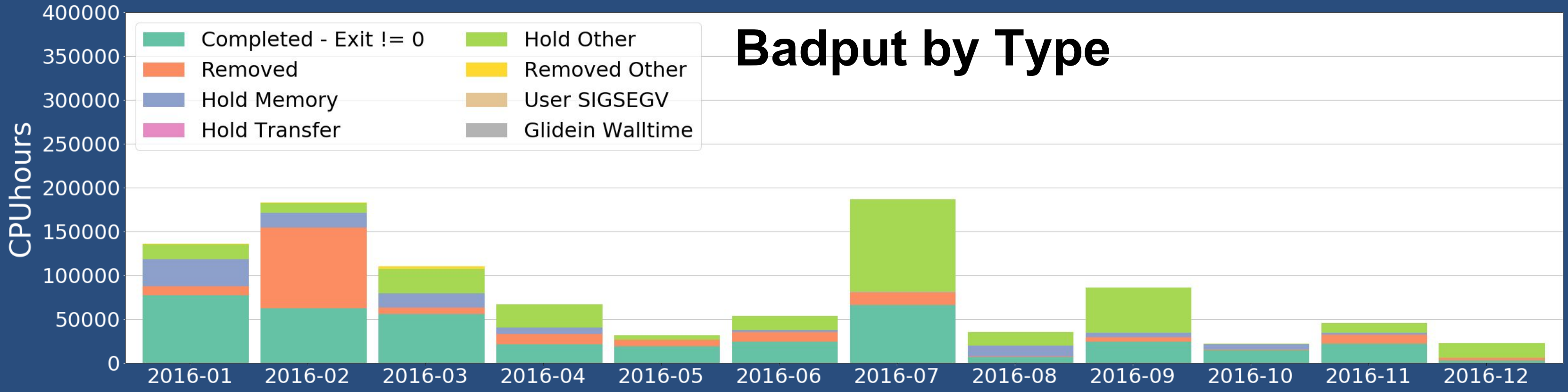
Grid Usage

CPU - Campus Pool

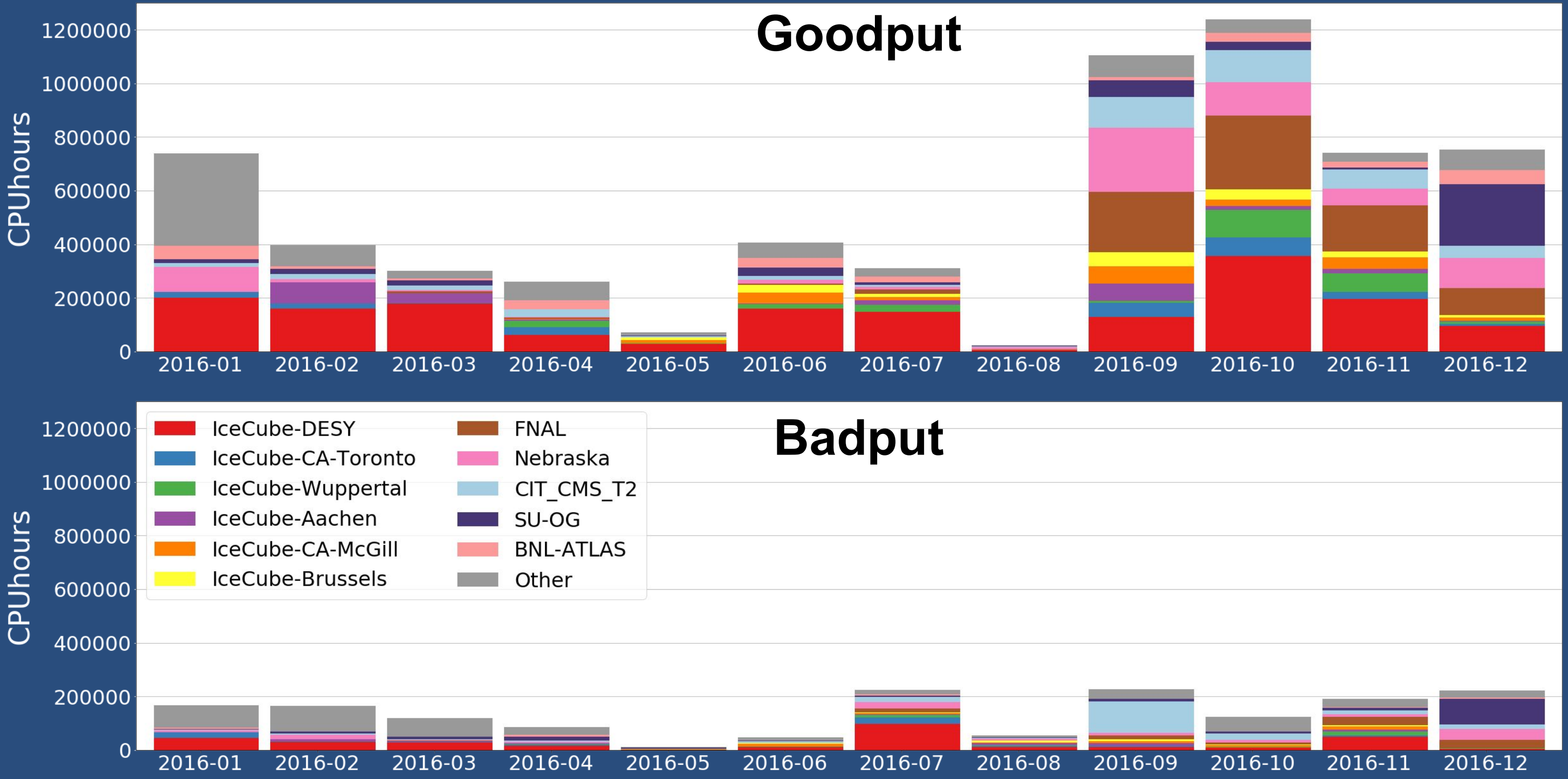


CPU - Campus Pool

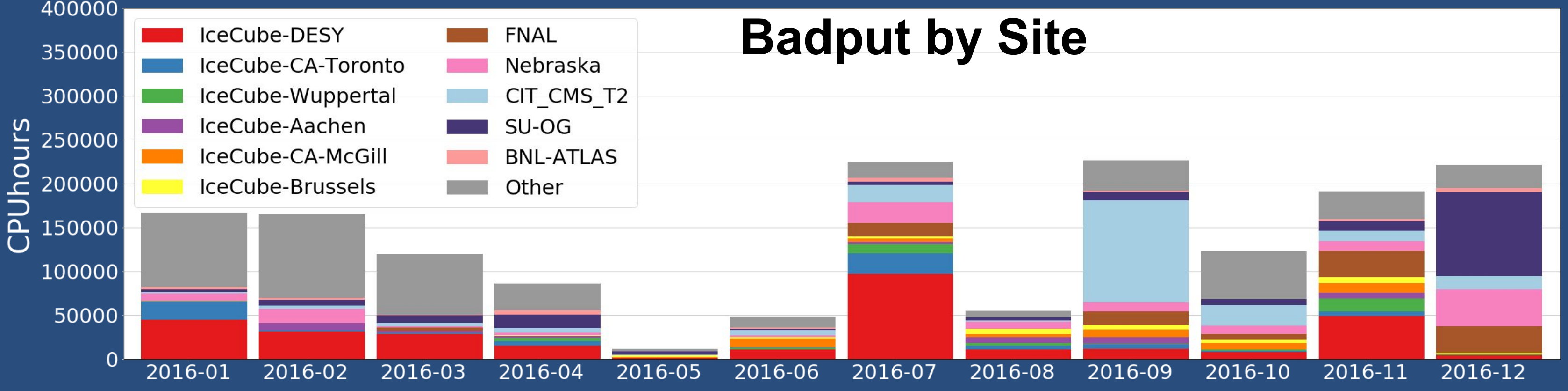


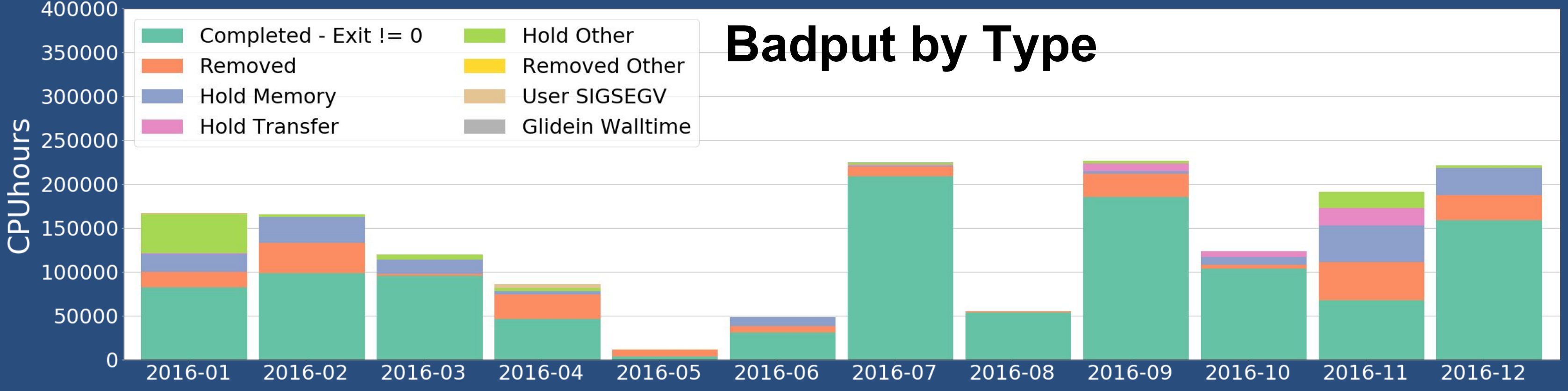


CPU - GLOW VOFrontend (IceCube VO)

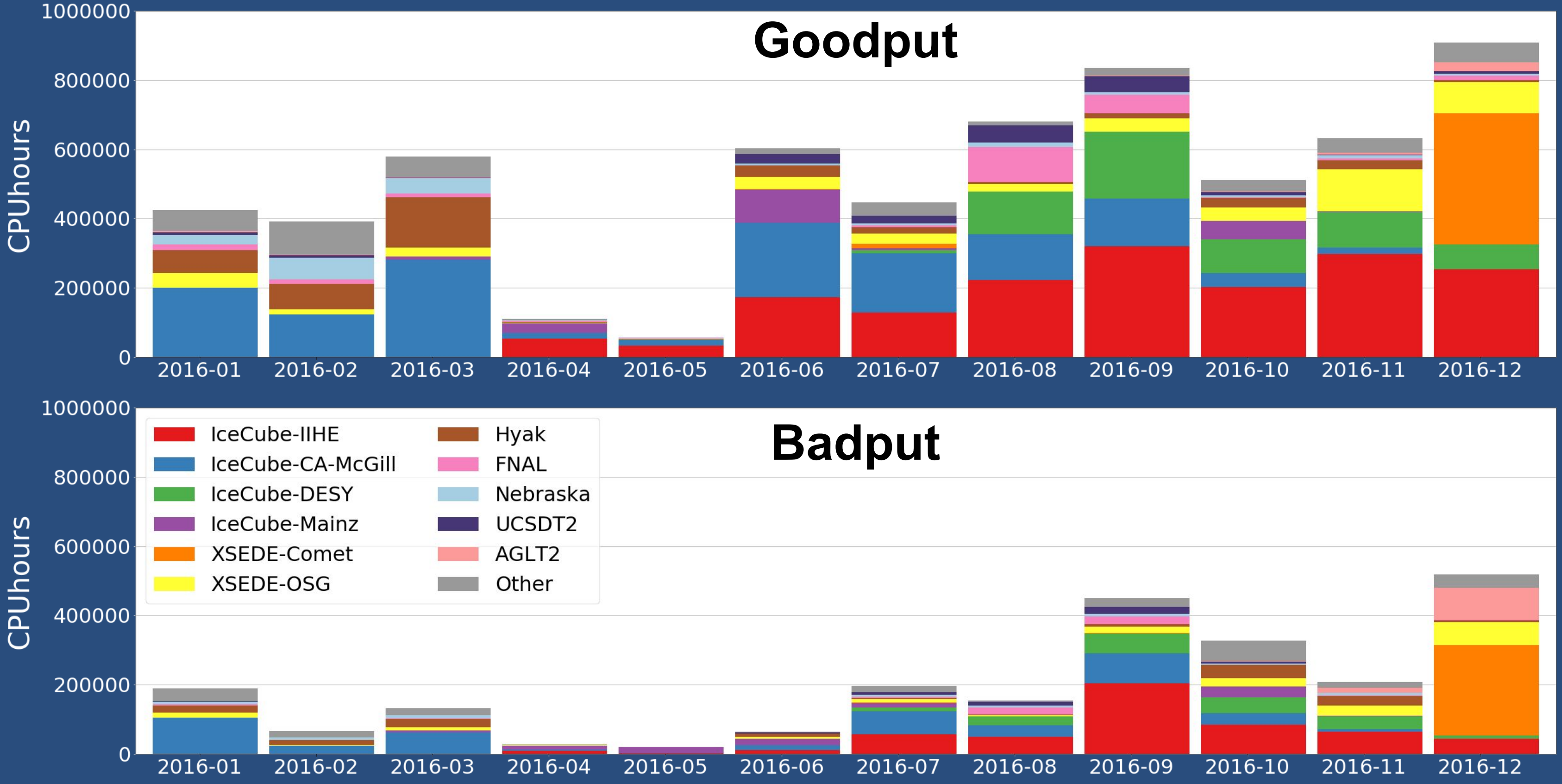


CPU - GLOW VOFrontend (IceCube VO)

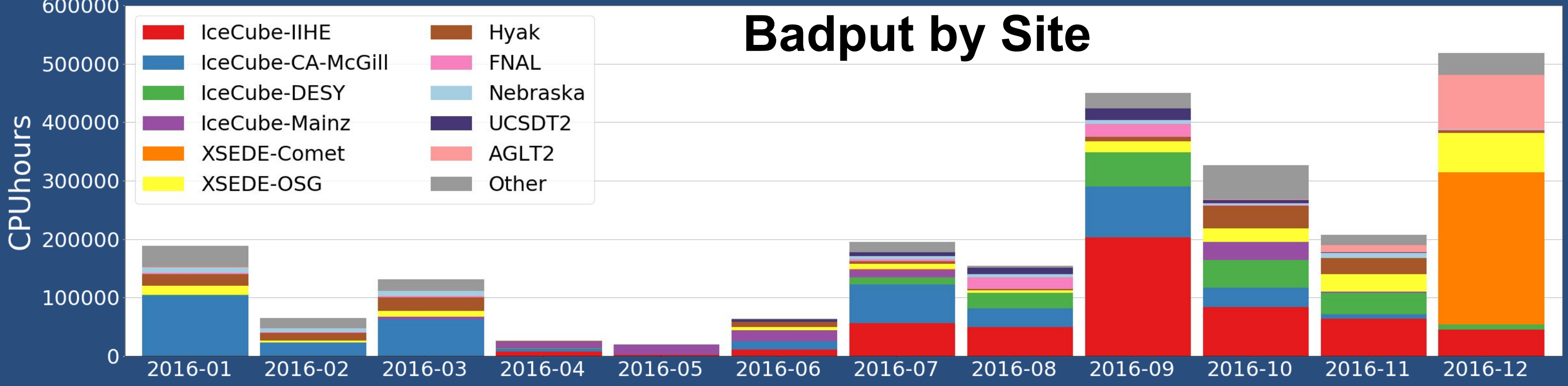


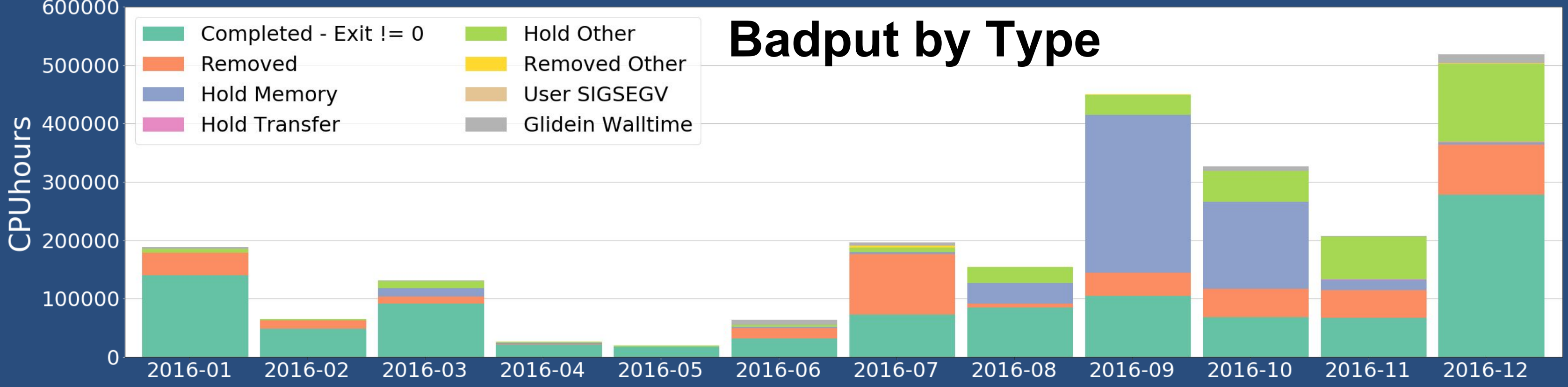


CPU - Pyglidein

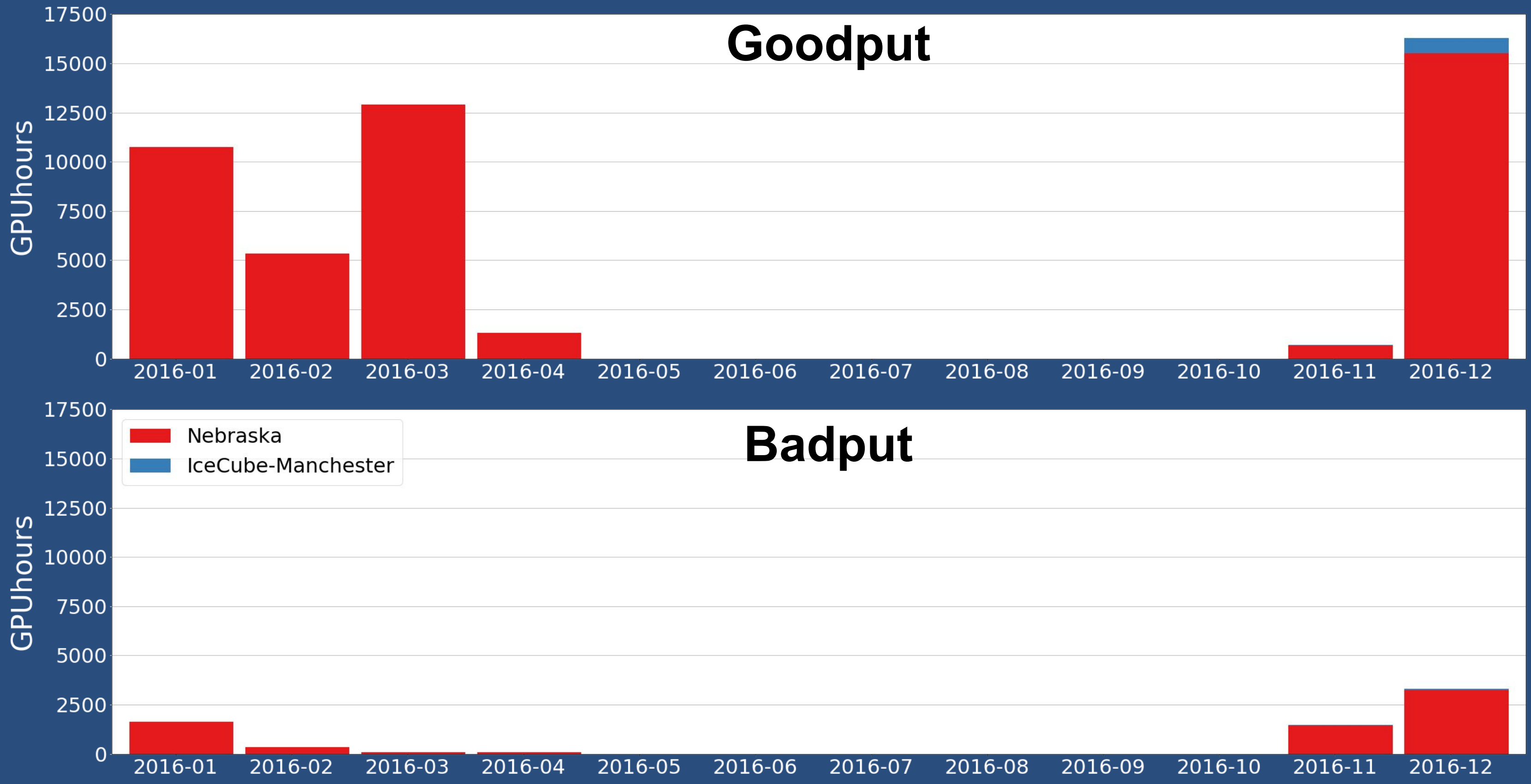


CPU - Pyglidein

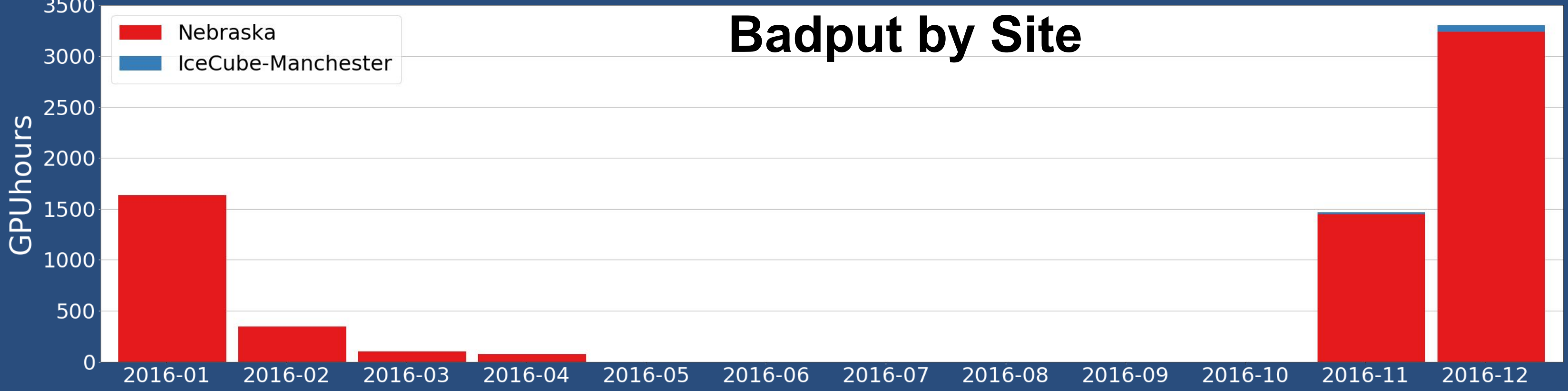


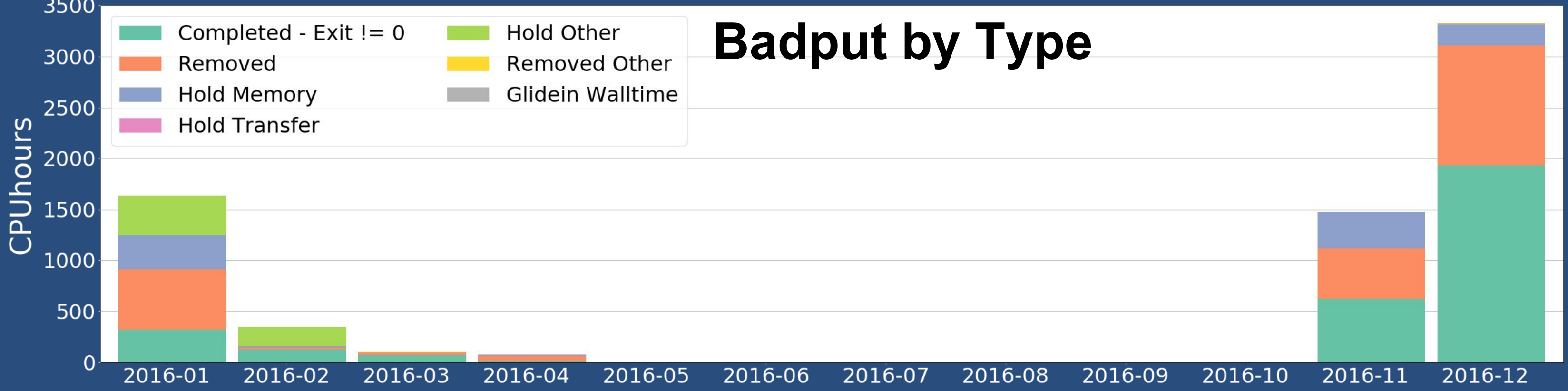


GPU - GLOW VOFrontend (IceCube VO)

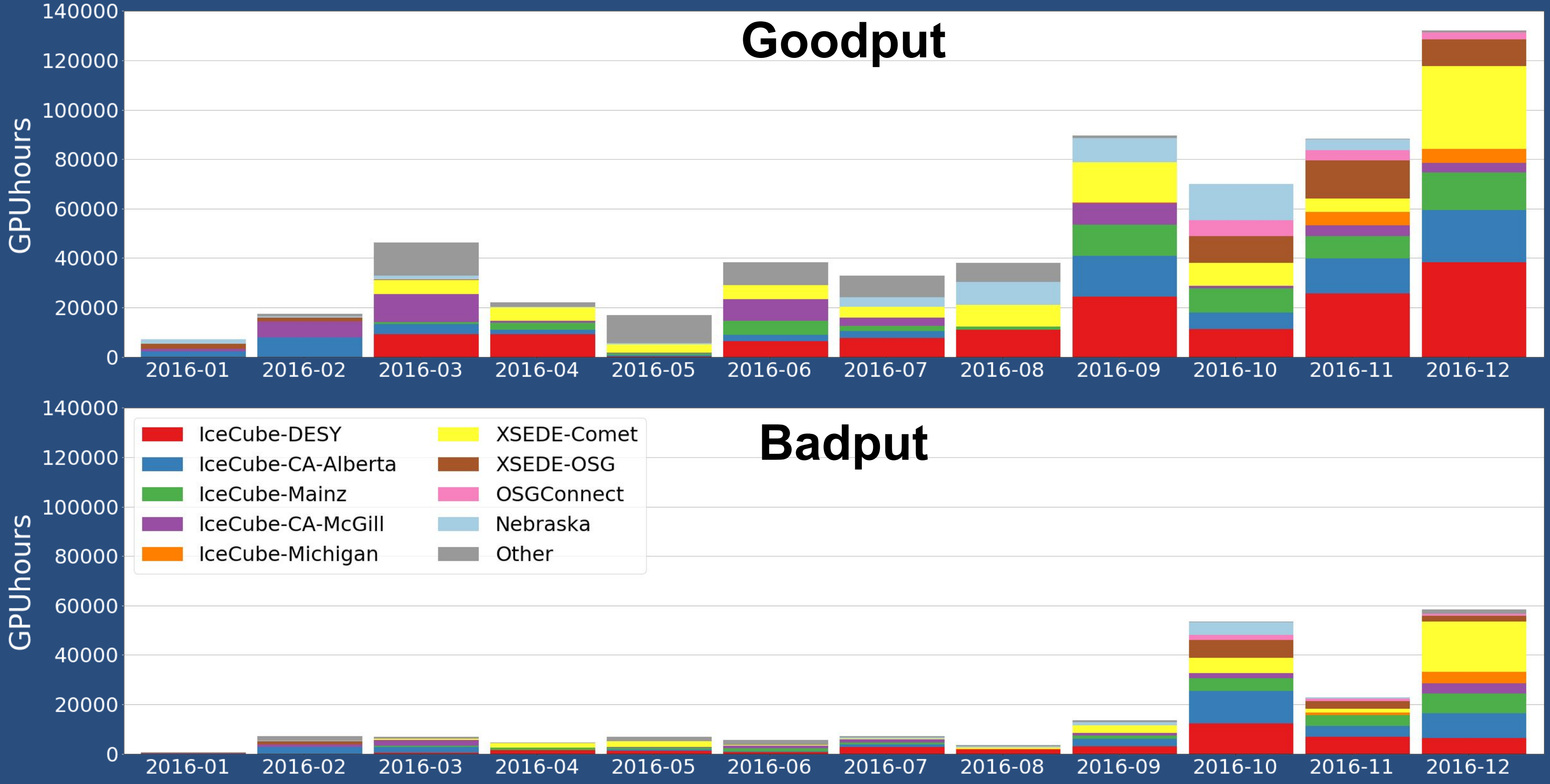


GPU - GLOW VOFrontend (IceCube VO)

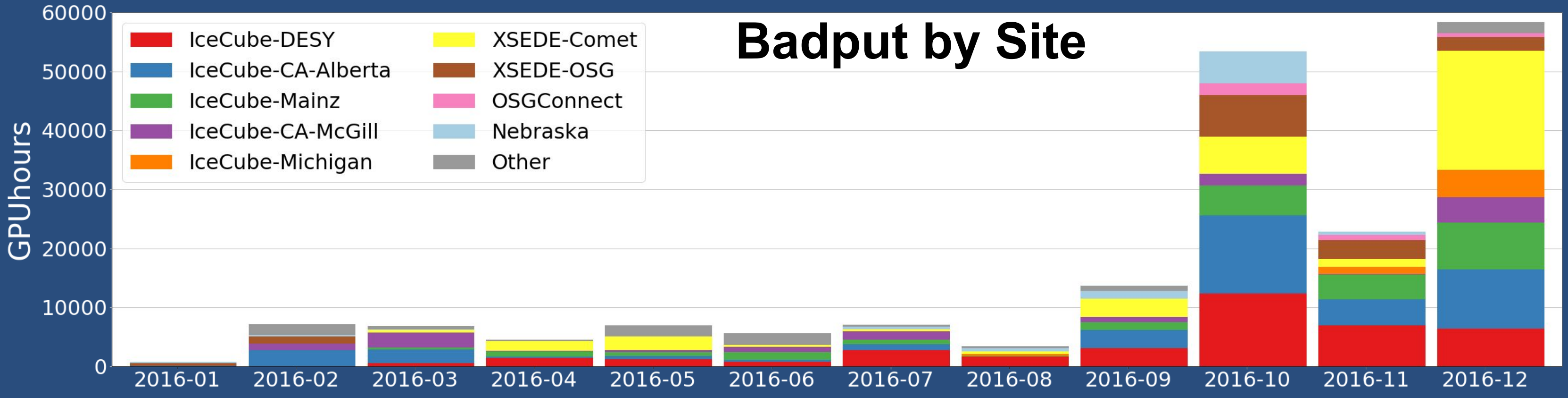


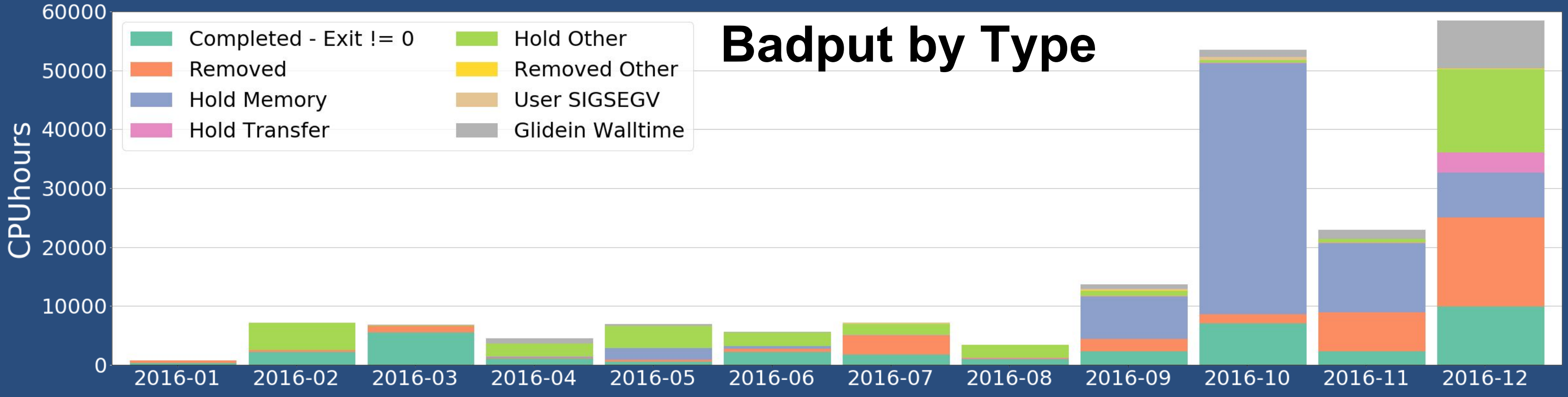


GPU - Pyglidein

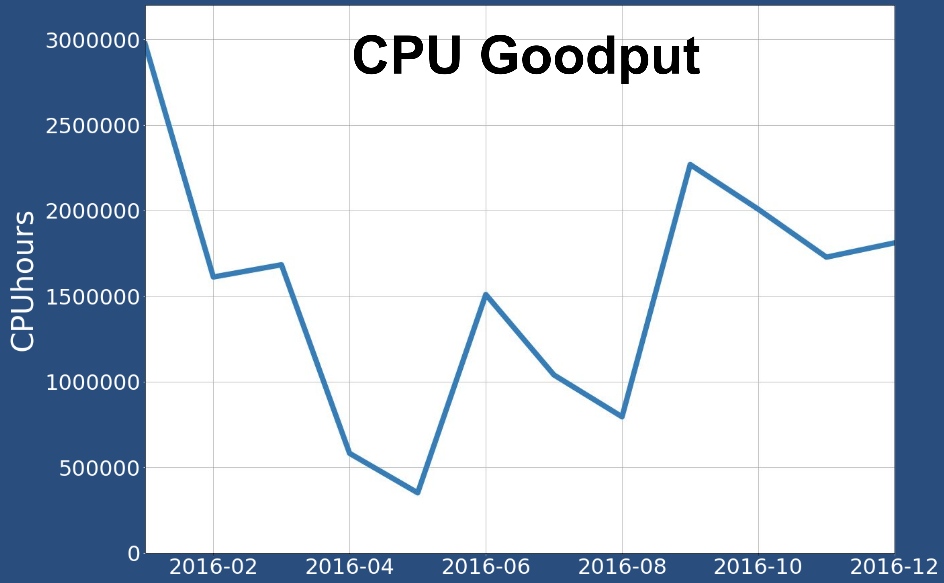


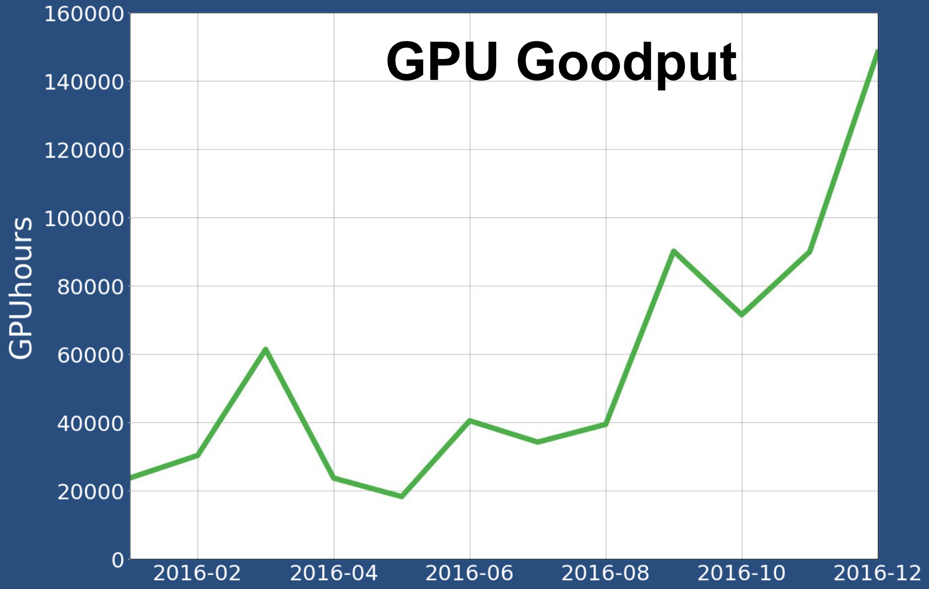
GPU - Pyglidein





Grid Usage Totals





CPU: 18.3M hours GPU: 650K hours Badput: 20%

Pyglidein

Pyglidein Advantages

- All IceCube sites in a single HTCondor pool
  - Priority is easier with one control point
- Simplified process for new sites to “join” pool
  - Feedback is positive
    - “Much better than the old system”
  - Useful for integrating XSEDE sites

Use Case - CHTC

- Main shared cluster on campus
  - We used 6M hours in 2016
- Before: flock to CHTC
  - Priority control on CHTC side, no control locally
- Now using pyglidein
  - Priority control locally
  - UW resource: prefer UW users before collaboration

Some Central Manager Problems

- Lots of disconnects
  - VM running collector, negotiator, shared_port, CCB:
    - 8 cpus, 12GB memory
    - Pool password authentication
    - 5k-10k startds connected
    - 10k-40k established TCP connections

Some Central Manager Problems

- Suspect a scalability issue
  - Frequent shared_port blocks and failures
  - Frequent CCB rejects and failures
  - Suspicious number of lease expirations
- Pyglidein idle timeout is 20 minutes
  - Lots of timeouts even with idle jobs in queue
- Ideas welcome

Future Work

- Troubleshooting
  - Easier gathering of glidein logs
  - Better error messages
  - Ways to address black holes
    - Remotely stop the startd
    - Watchdog inside glidein

Future Work

- Monitoring
  - Store more information in condor_history job records
    - GLIDEIN_Site, GPU_Type ...
  - Better analyzing tools for condor_history
    - All plots today using MongoDB + matplotlib
    - Interested in other options (ELK?)
    - Any options for getting real-time plots?
  - Dashboard showing site status (similar to SAM, RSV)

Future Work

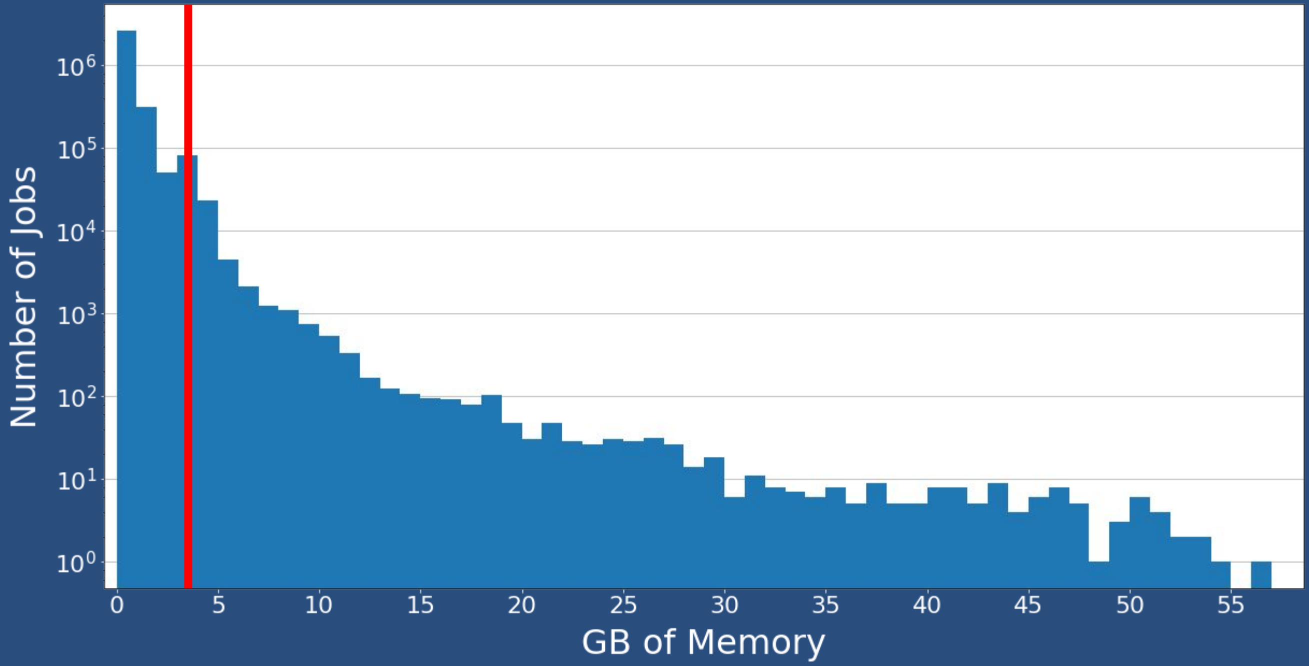
- Wishlist for this year
  - Automatic updating of the client
  - Restrict a glidein to specific users
    - Add special classad to match on?
  - Use “time to live” to make better matching decisions
  - Work better inside containers

Issues / Events Highlights

GPU Job Memory Overuse

GPU Job Memory Overuse

- 2.5% of GPU jobs go over memory request



GPU Job Memory Overuse

- No way to pre-determine memory requirements
- But we do have access to large partitionable slots (and we control the startd on Pyglidein)
  - Dynamically resize the slot with available memory?
  - Evict CPU jobs so the GPU job can continue?
  - Can we do this with HTCondor?

Data Reprocessing - “Pass2”

Data Reprocessing - “Pass2”

- IceCube will reprocess data from 2010 to 2015
  - Improved calibration, updated software
  - Uniform multi-year dataset
  - First time we went back to RAW data
    - Previous analyses all used the online filtered data
  - We want to use the Grid
    - First time data processing will use the Grid (only simulation and user analysis so far)

Data Reprocessing - “Pass2”

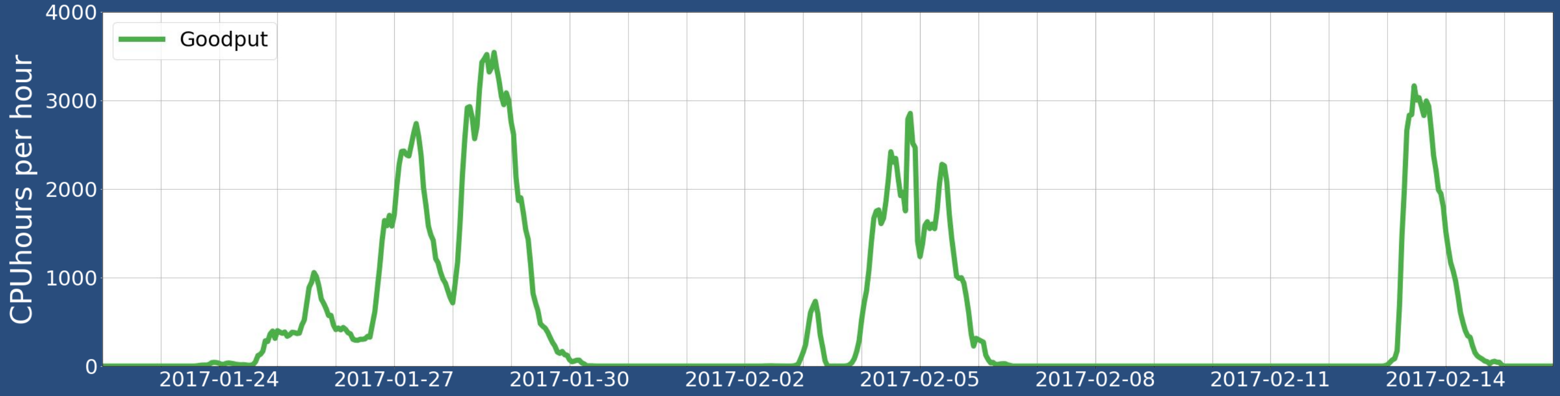
Season	Input Data	Output Data	Estimated CPU Hrs
2010	148 TB	44 TB	1,250,000
2011	97 TB	47 TB	1,263,000
2012	163 TB	53 TB	1,237,000
2013	139 TB	61 TB	1,739,000
2014	149 TB	58 TB	1,544,000
2015	78 TB	56 TB	1,513,000
Totals	*774 TB*	*319 TB*	*8,546,000*

Data Reprocessing - “Pass2”

- Requirements per job:
  - 500 MB input, 200 MB output
  - 4.2 GB memory
  - 5-8 hours
  - Currently SL6-only

Data Reprocessing - “Pass2”

- 10% sample already processed for verification
  - Have been able to access 3000+ slots



- Full reprocessing estimated to take 3 months

XSEDE Allocations

2016 XSEDE Allocations

	GPUs in System	Allocated SUs	Used SUs (2/27/2017)	%
Comet	72 K80	5,543,895	3,132,072	57
Bridges	16 K80 +32 P100 in Jan	512,665	172,025	34

2016 XSEDE Allocations

- Issue: large Comet allocation compared to actual GPU resources
  - We did only ask for GPUs in the request
  - Impossible to use all allocated time as GPUhours
- Extended allocation through June 2017
  - A chance at using more of the allocation

Future Allocations

- Experience with Comet / Bridges very useful
  - Better understanding of XSEDE XRAS process
  - Navigating setup issues at different sites
- Next focus: larger GPU systems
  - Xstream
  - Titan?
  - Bluewaters?

Long Term Archive

Long Term Archive

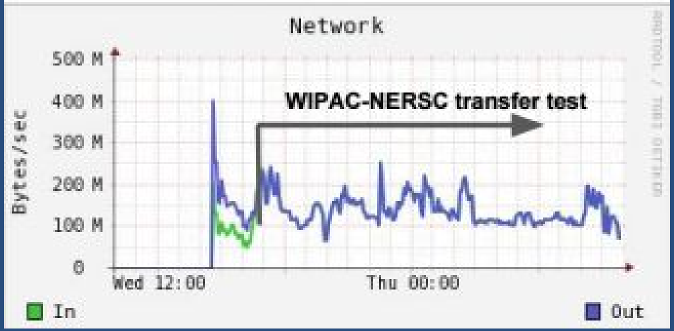
- Data products to be preserved for long time
  - RAW, DST, Level2, Level3 ...
- Two collaborating sites providing tape archive
  - DESY-ZN and NERSC
- Added functionality to existing data handling sw
  - Index and bundle files in the Madison data warehouse
  - Manage WAN transfers via globus.org
  - Bookkeeping

Long Term Archive

- Goal is to get ~40TB/day (~500MB/s)
  - ~3 PB initial upload
  - +700 TB/yr
    - ~400 TB/yr bulk upload in April (disks from South Pole)
    - ~300 TB/yr constant throughout the year

Long Term Archive

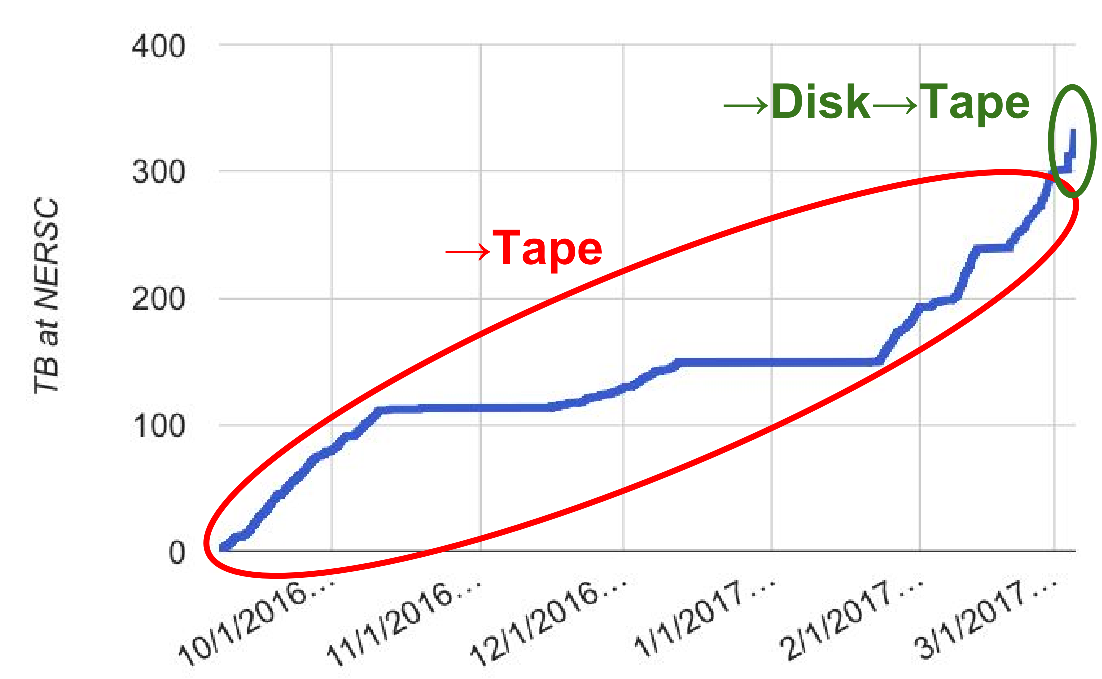
- Started archiving files in Sept 2016
- uw → nersc#hpss:
  - Direct gridftp to tape endpoint
  - ~100MB/s: 12 concurrent files, 1 stream/file



Long Term Archive

- Now trying two-step transfer
  - Buffer on NERSC disk before transfer to tape
- uw → nersc#dtn:
  - Gridftp to disk endpoint
  - ~600-800 MB/s: 24 concurrent files, 4 streams/file
- NERSC internal disk→tape: >600MB/s

Long Term Archive



Summary

- CVMFS
  - Working well for production
  - Potential expansion to users
- Grid
  - IceCube using 2 glidein types
  - More resources than ever
  - Still much work to be done
- Issues & Events
  - GPU memory problem
  - “Pass2” data reprocessing
  - XSEDE allocations
  - Long term archive