A UT/ORNL PARTNERSHIP
NATIONAL INSTITUTE FOR COMPUTATIONAL SCIENCES
NICS

Collecting Application-Level Job Completion Statistics

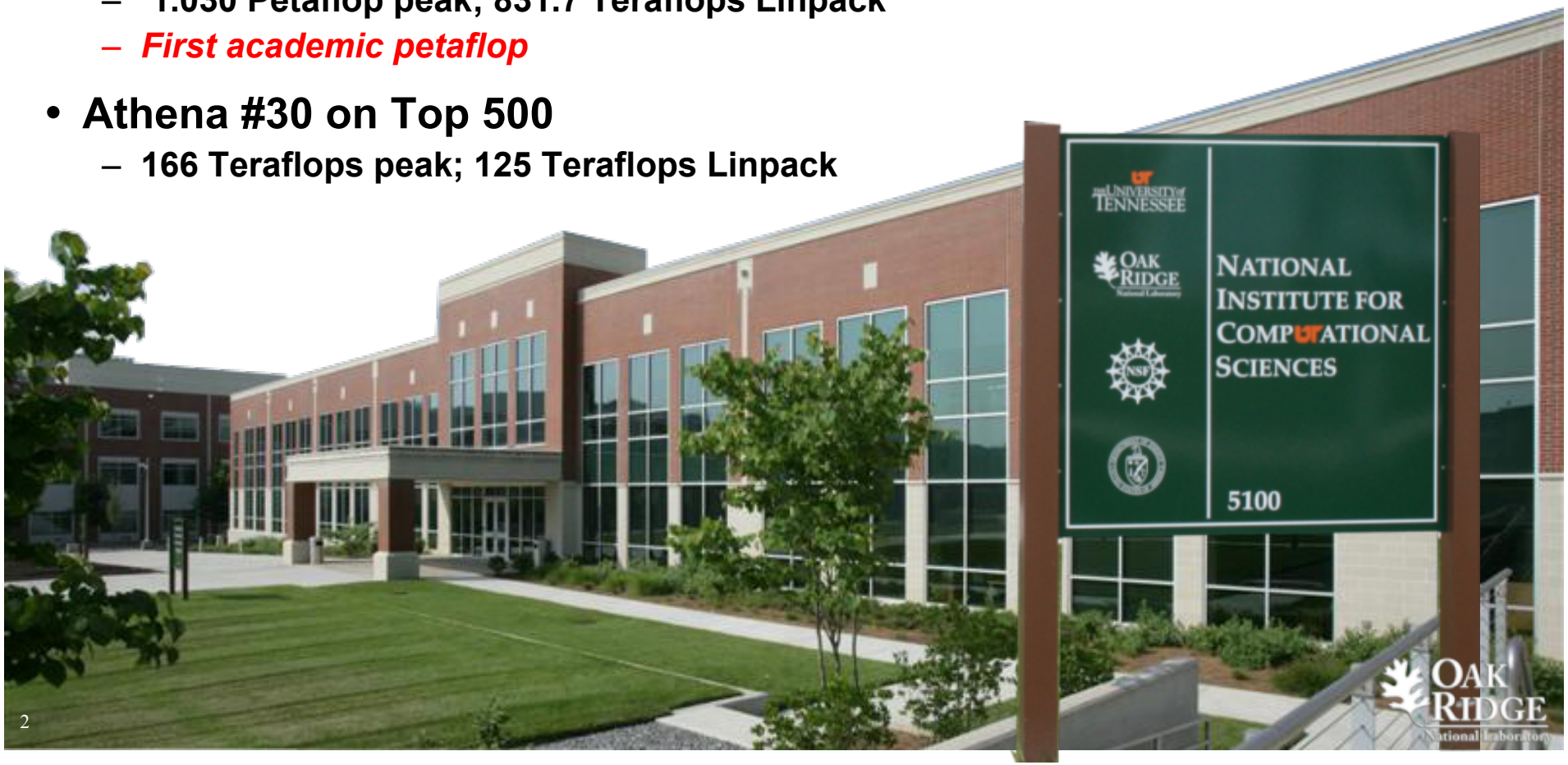
CUG 2010, Edinburgh

Matthew Ezell

HPC Systems Administrator

National Institute for Computational Sciences
University of Tennessee

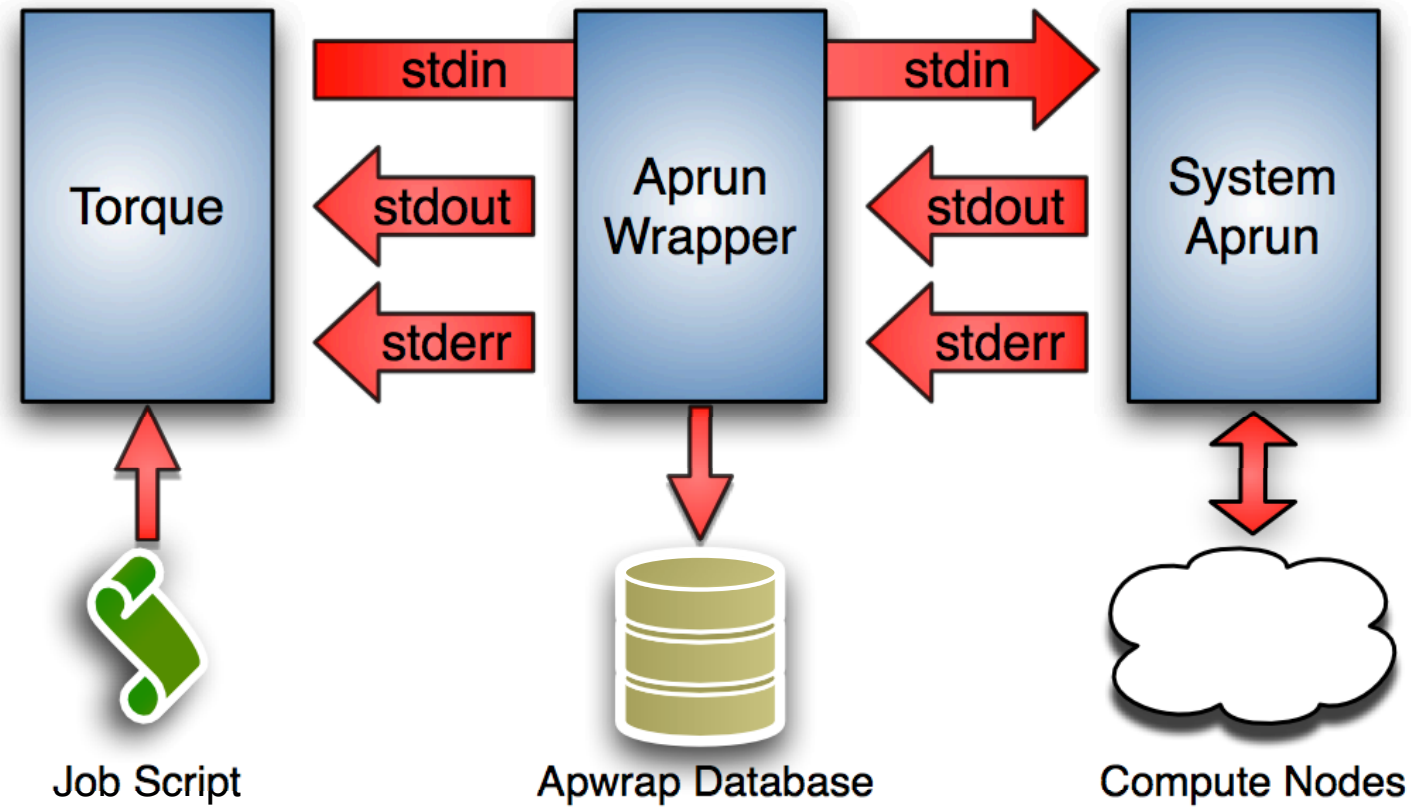
- NICS is the latest NSF HPC center
- Kraken #3 on Top 500
  - 1.030 Petaflop peak; 831.7 Teraflops Linpack
  - *First academic petaflop*
- Athena #30 on Top 500
  - 166 Teraflops peak; 125 Teraflops Linpack



Motivation and Goals

- Need for statistics on the frequency and nature of job failures
- XT Systems produce massive amounts of log data
  - Some job-level error messages are only put in job standard output or standard error
- It should have the ability to explain “cryptic” error messages to users
- Should not increase job walltime or modify user experience

Design: *apwrap* Data Flow



Design: Prologues and Epilogues

- Allow arbitrary, system-defined programs to run before and after *aprun* execution
- Should be able to send messages to the user and/or prevent the application from being launched
- Can be integrated with other tools, such as the Automatic Library Tracking Database (ALTD) at NICS

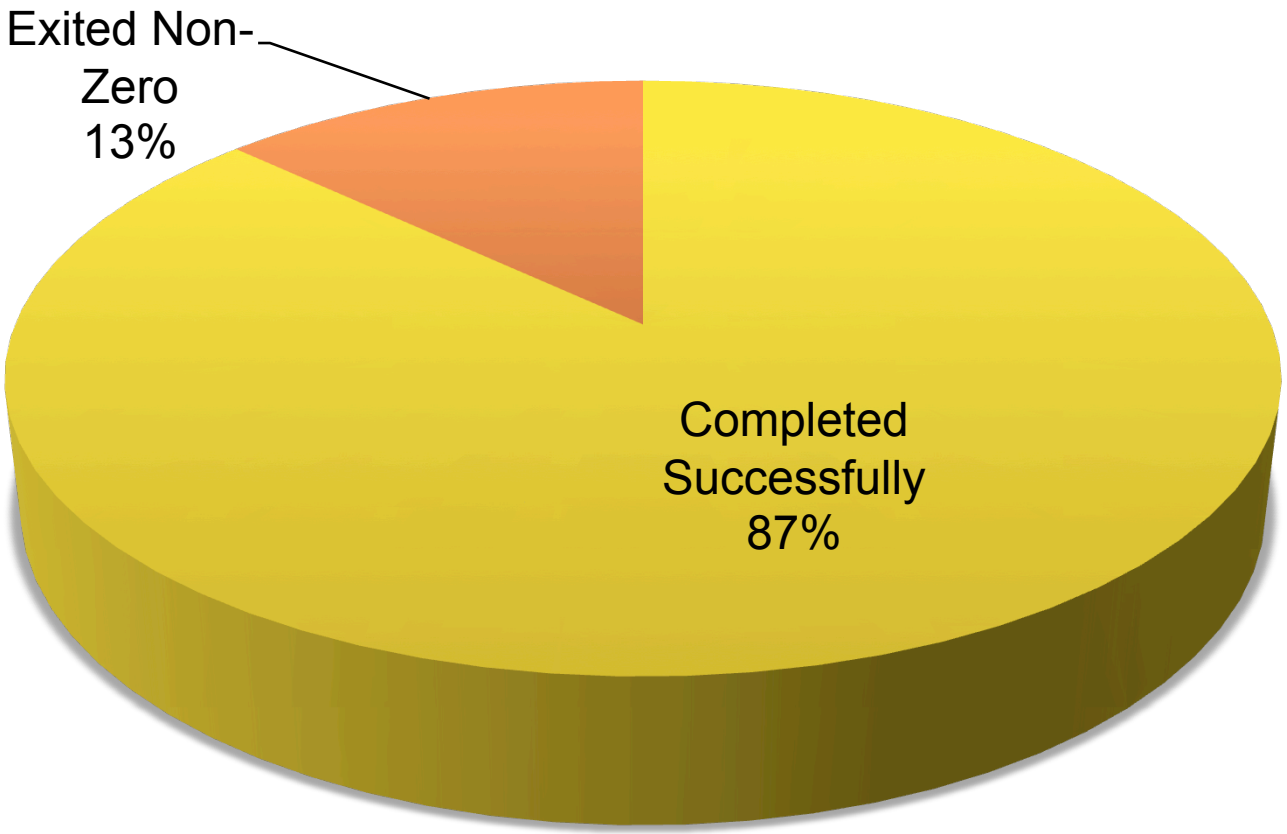
Design: Example Rules

```
rules   => [{
           name=>          'NODEFAIL',
           pattern=>       '^\[NID \d+\] \d{4}-\d{2}-\d{2} \d{2}:\d{2}:\d{2} Apid \d+
                           killed. Received node failed or halted event for nid (\d+)',
           message=>       'A compute node had a hardware failure. Please resubmit your
                           job.'
           },{
           name=>          'SEGFAULT',
           pattern=>       '^_pmii_daemon\(SIGCHLD\): PE \d+ exit signal
                           Segmentation fault',
           message=>       'A node experienced a segmentation fault. This happens when
                           the code attempts to access a memory location that it is not
                           allowed to.'
}]
```

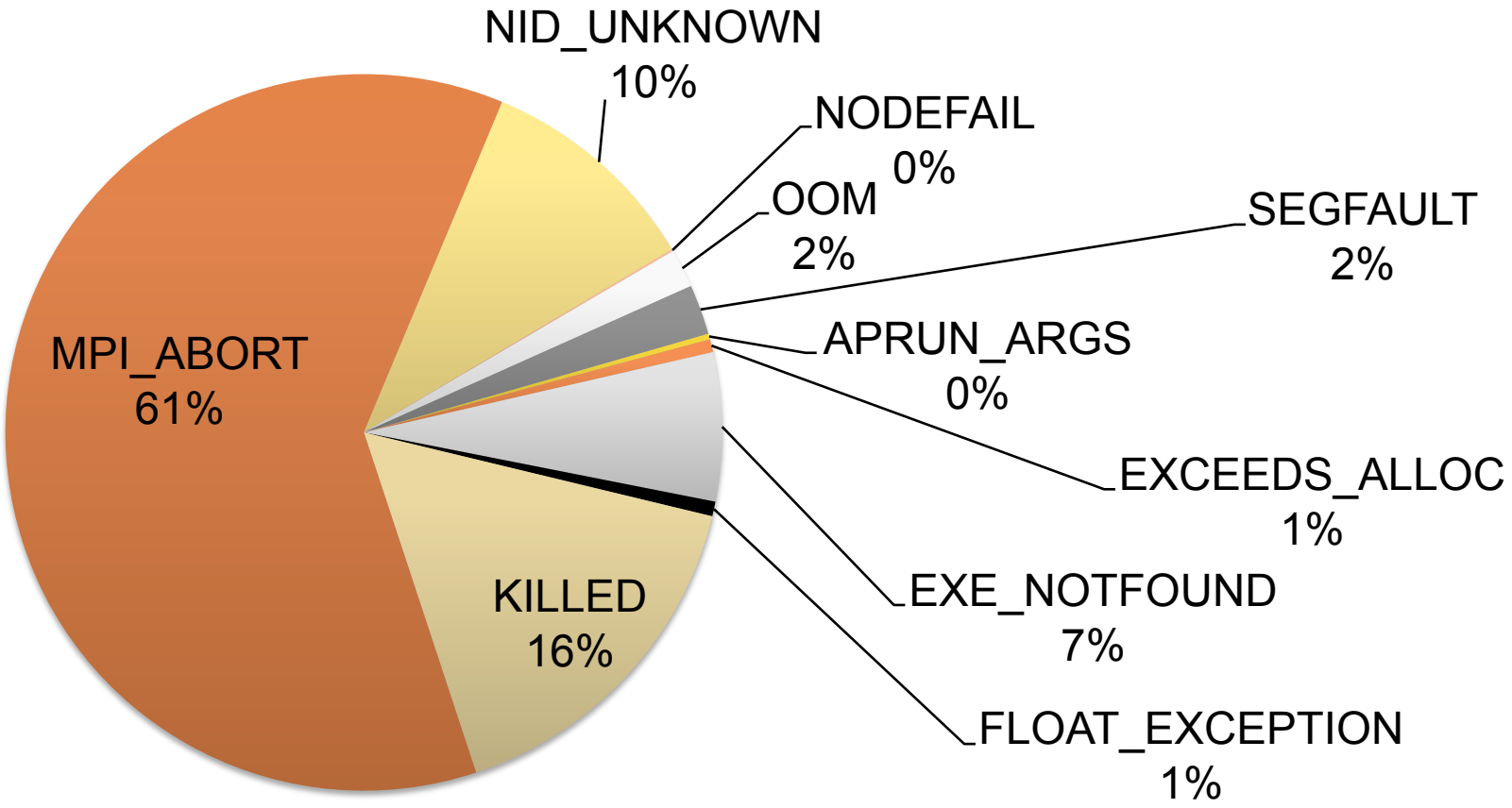
Sample Database Entry

id	189
username	*user1*
system	athena
pbsserver	nid00004
batchid	68122.nid00004
batchidnum	68122
apid	1290954
batch_node	aprun3
pwd	/lustre/scratch/*user1*
arguments	-n 4096 -N 1 -d 4 *binary*
pes	4096
pes_per_node	1
depth	4
user_binary	/lustre/scratch/*user1*/*binary*
mpmd	f
pid	18367
start_time	1270358965
exit_time	1270366985
Duration	8020
exit_code	1
error_name	NODEFAIL
error_string	[NID 15050] 2010-04-04 03:42:45 Apid 1290954 killed. Received node failed or halted event for nid 15051

Successful Completion Rate



Types of Errors Experienced



MPI_ABORT (61%)

- The code purposely calls this function
- May occur if
  - an input file could not be found
  - the algorithm reaches numeric instability
  - a call to malloc() returns a NULL pointer
  - etc…
- Usually not a system problem

KILLED (16%)

- Two Causes
  - Job runs out of walltime, batch system kills it
  - User chooses to kill the job/app
- Extended walltime *may* be due to a system problem, but it's difficult to tell

NID_UNKNOWN (10%)

- Usually code-specific

```
The last 50 lines from stderr follow:

wks.c: Error in opngks_(): Could not open "./
  20100517-gmeta/comref-2010051700_spg40-24h.gmeta"

FORTRAN STOP

[NID 00078] 2010-05-17 11:57:19 Apid 1409935:
  initiated application termination
```

Conclusions

- Most errors experienced by users are (most likely) due to users errors
- System-level errors are more rare, and require administrator involvement to debug

Questions?

Contact me at ezell@nics.utk.edu