TACC
TEXAS ADVANCED COMPUTING CENTER

STAMPEDE 2, THE MIGRATION TO MANY CORE AND OTHER BURNING ISSUES FOR HPC PROVIDERS

2016 Oklahoma Supercomputing Symposium

Dan Stanzione

Executive Director, TACC

9/21/2016

OUTLINE

- What we did with Stampede
- What we're doing now to change architectures
- What's coming with Stampede-2
- The changing use cases: Where we aren't using Stampede-2, and a word about CI Architecture.

STAMPEDE

- Awarded by the NSF for a production run of January 7th, 2013 to January 6th, 2017.
- A 6,400 node Dell cluster of Intel Sandy Bridge Nodes with Knights Corner accelerator cards
  - Also subsystems of nVidia K20 GPUs and Large Memory Nodes.
- A national resource through XSEDE.
- *By all measures, this project has been remarkably successful…*

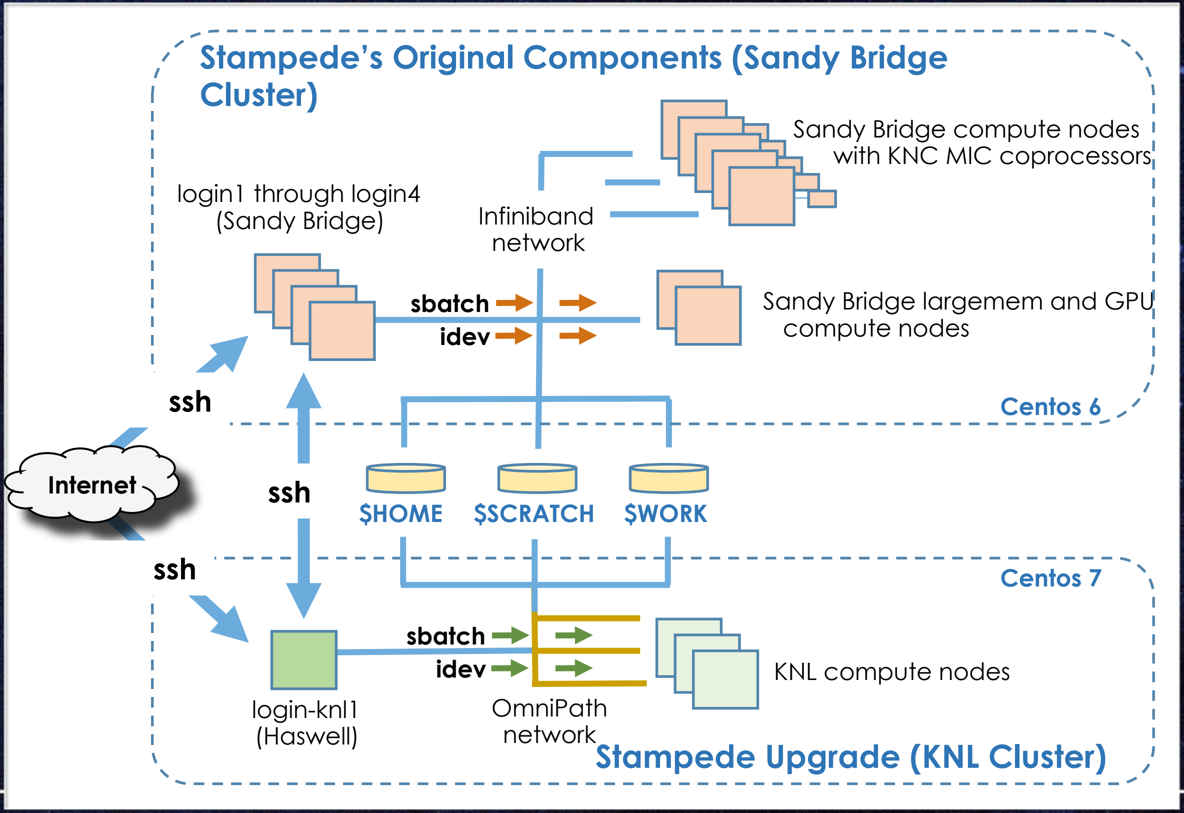
STAMPEDE OPERATIONAL STATS

Metric	12 months thru 7/2016	Project Totals to Date
# of users running jobs:	**5,303**	**10,458**
# of projects:	1,795	3,145
#of jobs:	**1,761,366**	**7,071,847**
SU’s delivered:	**829,804,026**	**2,649,762,345**
SU’s allocated to XSEDE	815,191,387	3,357,469,282
System Uptime:	**98.88%**	**97.71%**
Tickets resolved by staff:	4,720	14,495

STAMPEDE 1.5
PIONEERING A SECOND GENERATION OF MANY CORE

- Revision of upgrade plan in original proposal.
- Add 500 KNL *nodes* to the existing Stampede
  - Provided by Dell/Intel
  - Replace original plan for KNL cards
- OmniPath connected (bridge to IB for Lustre).
  - Omnipath in-house now for evaluation.
  - Along with “first wave” of KNL systems
- **First KNL system on Top 500 (#112 all by itself)**
- *Smooth the transition to Stampede 2*
- **Available now. . .**

STAMPEDE KNL SUBSYSTEM



Stampede's Original Components (Sandy Bridge Cluster)

login1 through login4
(Sandy Bridge)

Infiniband
network

Sandy Bridge compute nodes
with KNC MIC coprocessors

sbatch

idev

Sandy Bridge largemem and GPU
compute nodes

Centos 6

ssh

Internet

ssh

$HOME

$SCRATCH

$WORK

ssh

Centos 7

sbatch

idev

KNL compute nodes

login-knl1
(Haswell)

OmniPath
network

Stampede Upgrade (KNL Cluster)

STAMPEDE 2

- Officially Awarded June 6th of 2016 (after a long wait).
- A renewal of the National Science Foundation Stampede project award for 2017-2021.
- $30M for the acquisition of a replacement for Stampede
  - Additional operational funding to follow.
- TACC's 3rd leadership class supercomputer:
  - 2008 Ranger: the fastest open science system in the world.
  - 2013 Stampede: 7M jobs for 10,000 users and counting.
  - 2017 Stampede-2: The largest US University system, and among the fastest in the world
- Projecting ~18PF peak performance
  - 2x Stampede with accelerators – 9x Stampede without accelerators.

STAMPEDE 2
TIMELINES AND COMPONENTS

- Phase 1: Spring 2017
  - Roll in the “1.5” components with new Dell KNL sleds and Dell Director-class Omnipath switches.
  - Total of 4,200 KNL Nodes.
  - Replace filesystem
  - (~60% of Original Stampede will keep running through this phase).
- Phase 2: Fall 2017
  - Add 1,736 Dell nodes with Intel “Sky Lake” Processors
- Phase 3: Spring 2018
  - Add 3D Xpoint memory to a small subset of nodes.

STAMPEDE 2 POINTS OUT SOME OF THE CHALLENGES OF MODERN SYSTEMS

- Each processor has 68 cores.
- Each core runs 4 hardware threads
- Each instruction can operate on 8 operands at a time (512-bit vectors).
- 68*4= 272 ; 272*8 = 2,176 ; your code needs to process 2,176 instructions *at a time* to run on *one processor* of Stampede2.

CONTINUING ON THE PATH TO MANY CORE – BUT CONTINUING TO MAKE LIFE HARDER FOR SOFTWARE

- Our most aggressive early user program ever.
- We continue to invest in consulting, code optimization, low level performance modeling.
- The KNL Upgrade gives us our longest runway to a new technology for a large system. We will need it!
- The good news:
  - The highest use codes are already exceeding our expectations on KNL.
- For the others – they can run on Sky Lake – but will they even do as well there?

KNL PERFORMANCE AND APPLICATION EXPERIENCES

- We have some early experience with applications, though not with any code tuning.
  - If the past few years are any indication "no code changes" is the most appropriate mode to think about.
- The memory architecture makes performance a non-trivial question
  - No longer is it "how fast does my code go", but
  - "How fast does my code go in what mode with what options at what size and task/thread count?"

RETHINKING NOTIONS OF PERFORMANCE

- While we're at it, KNL-as-a-processor means we need to rethink what our value proposition is for performance:
  - It used to be "Is it worth adding an expensive accelerator to my already expensive node?".
  - Now it is "Should I buy a 'regular' server node, or a *less expensive* Xeon Phi node?".
    - Single socket Phi nodes should cost ~30% less than traditional dual-socket Xeon nodes ; less DIMMS and one less processor.
- Another odd notion – for throughput-oriented apps, single thread performance is lower than traditional Xeons ; but, you have 2-4x as many cores -- *node* throughput may be higher.

RETHINKING NOTIONS OF PERFORMANCE

- Power management combined with how we are pushing silicon increases variability.
  - For Ranger and Stampede-1, we would tune nodes to an HPL number of +/- 1% (across all 6,416 nodes).
  - For Haswell and Broadwell, we could only get to within +/- 3%
  - For Knights Landing, we are at +/-5%
- *(Note, this type of variability applies to Linpack only, as you have to be near the power threshold).*
- But, we have mapped other kinds of variability – every chip has certain cores turned off in at least one quadrant, so memory access time is not perfectly balanced.
  - Applications and benchmarks which rely on everything repeating perfectly each time (from a performance perspective) are becoming outmoded.

EARLY USER PERIOD

- Opened to early users in July
  - Staff access provided in mid June
- 17 projects, 73 users
- Projects range from molecular dynamics to astrophysics, CFD to machine learning
- Software stack
  - Intel 16.0.3
  - Intel MPI 5.1.3
  - Upgrading to Intel 17 and Intel MPI 17 (hmm, Intel MPI just jumped 12 versions!)

EARLY USER PERIOD

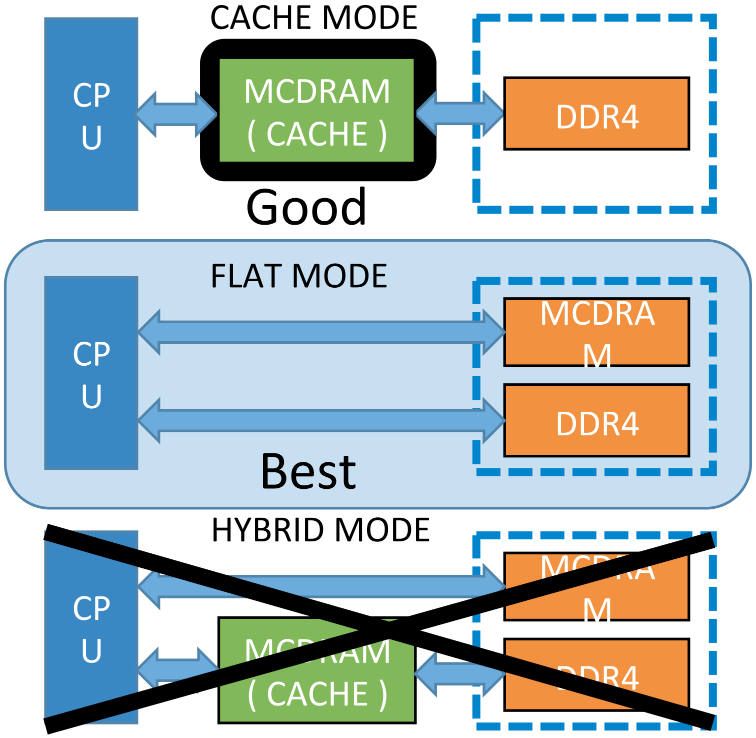
- Like all new platforms, the very early bring up is filled with identifying and removing a host of issues – that's why they don't immediately go out in quantities from the big vendors.
  - Determining validation/quality checks
  - Firmware issues
  - Linux kernel issues (older Linux was not a fan of 272 hyperthreads)
  - Compiler/libraries issues (mostly performance).
- We work closely with the vendor and other early customers – we have closed half a dozen notable issues, and many tinier fixes, across the various modes, hardware combinations, and performance
- Still 1 squirrely issue on very specific codes that we are fighting, but basically ready to go "full production"
- We're also exploring the right user environment – i.e.
  - switching memory modes requires a node reboot prior to job start.
  - Enable or disable huge pages in the kernel? C-states?

Memory Architecture

- Two main memory types
  - DDR4
  - MCDRAM

- Three memory modes
  - Cache
  - Flat
  - Hybrid

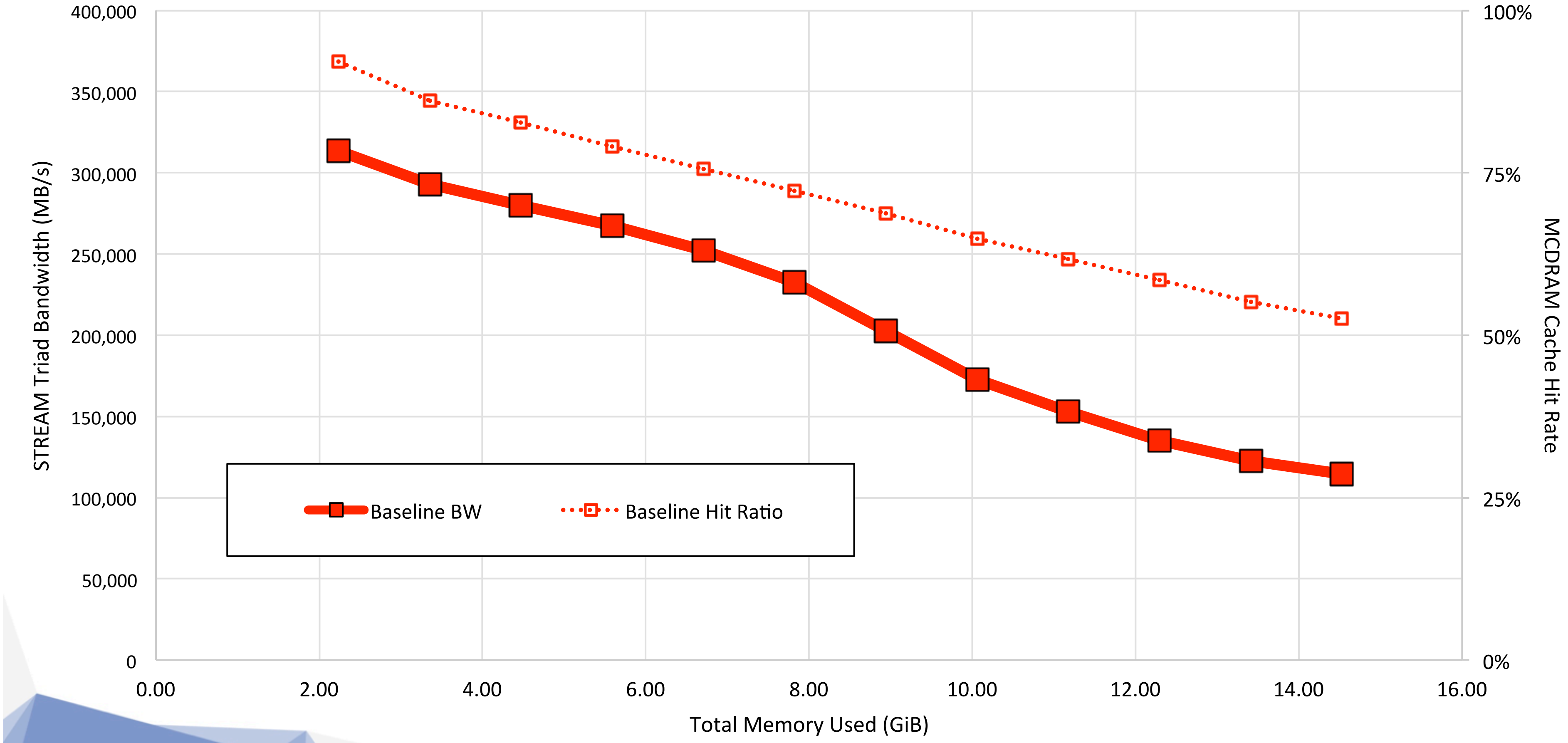
- Hybrid mode
  - Three choices
  - 25% / 50% / 75%
  - 4GB / 8 GB / 12GB



KNL MCDRAM Cache Mode

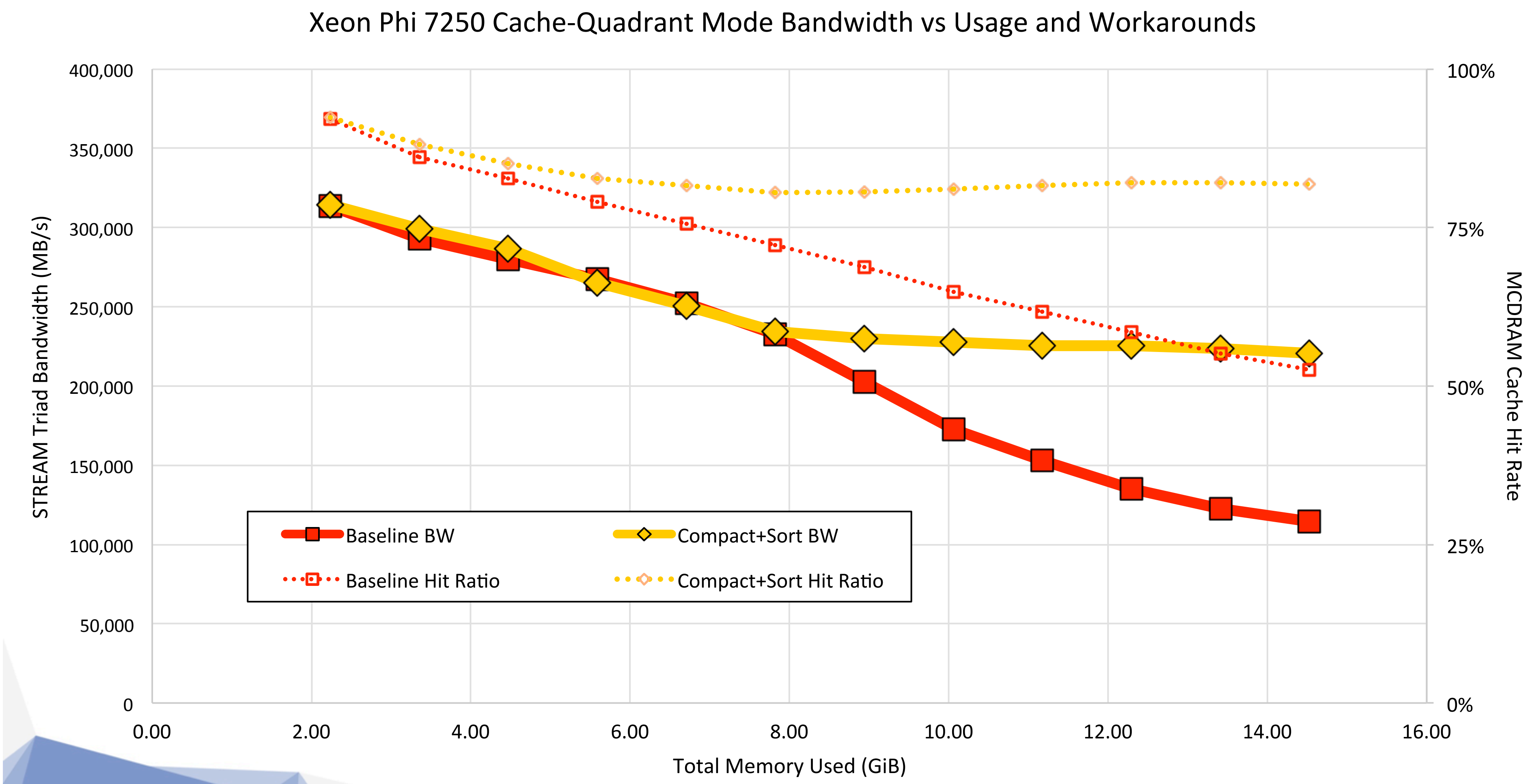
- The 16 GiB MCDRAM on Xeon Phi x200 can be configured as a direct-mapped cache
- Such a cache can be very effective, for *transparent* good performance.
- This is an example of where we have worked in early user to find workarounds:
  - Performance was observed to degrade over time (days to weeks)
  - Directed testing has confirmed this with both performance counter measurements and cache modeling using physical addresses
  - Workarounds are of varying degrees of usefulness
    - It does not take a large miss rate to degrade throughput significantly
    - But rebooting periodically between jobs is highly effective ☺.

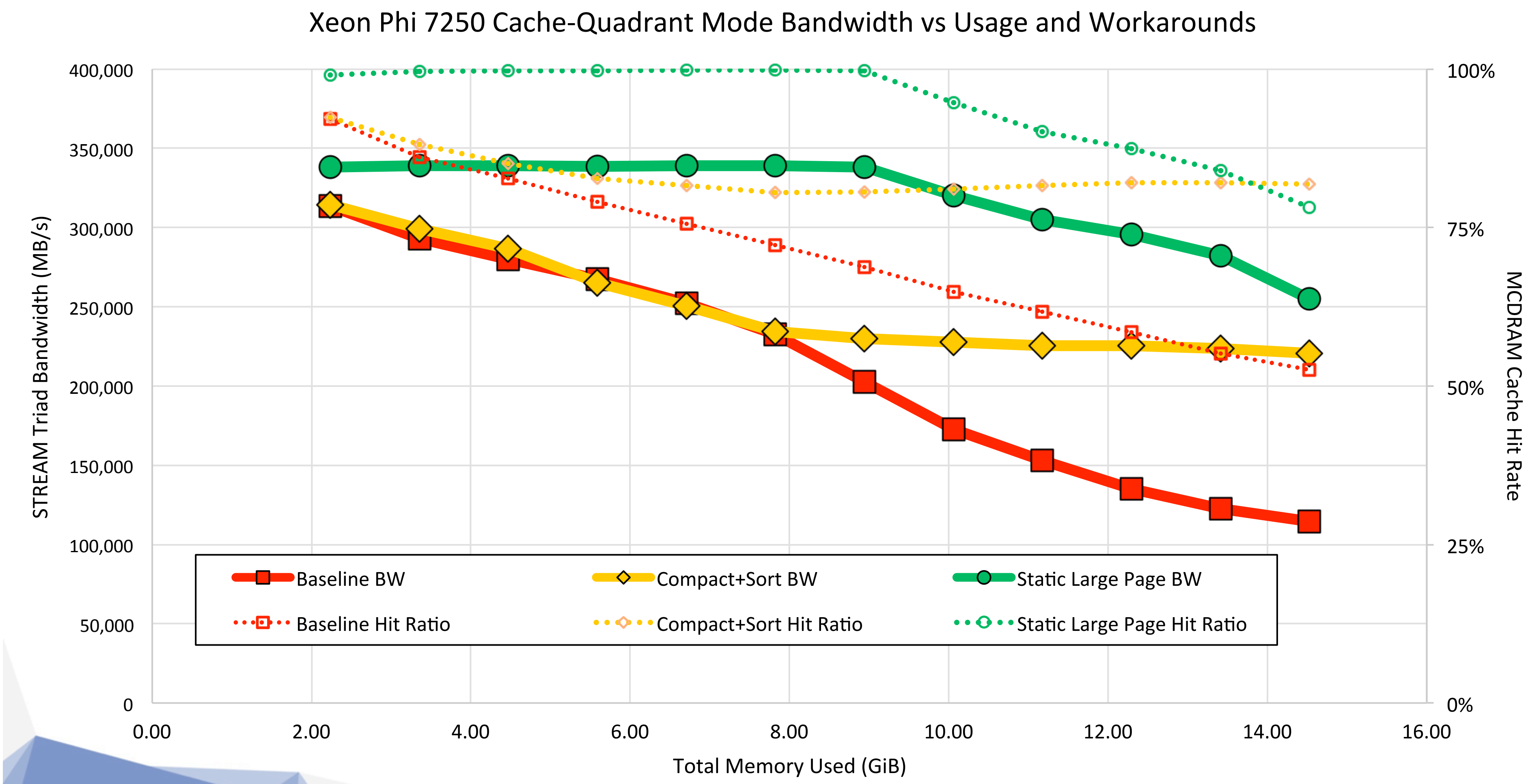
Xeon Phi 7250 Cache-Quadrant Mode Bandwidth vs Usage and Workarounds



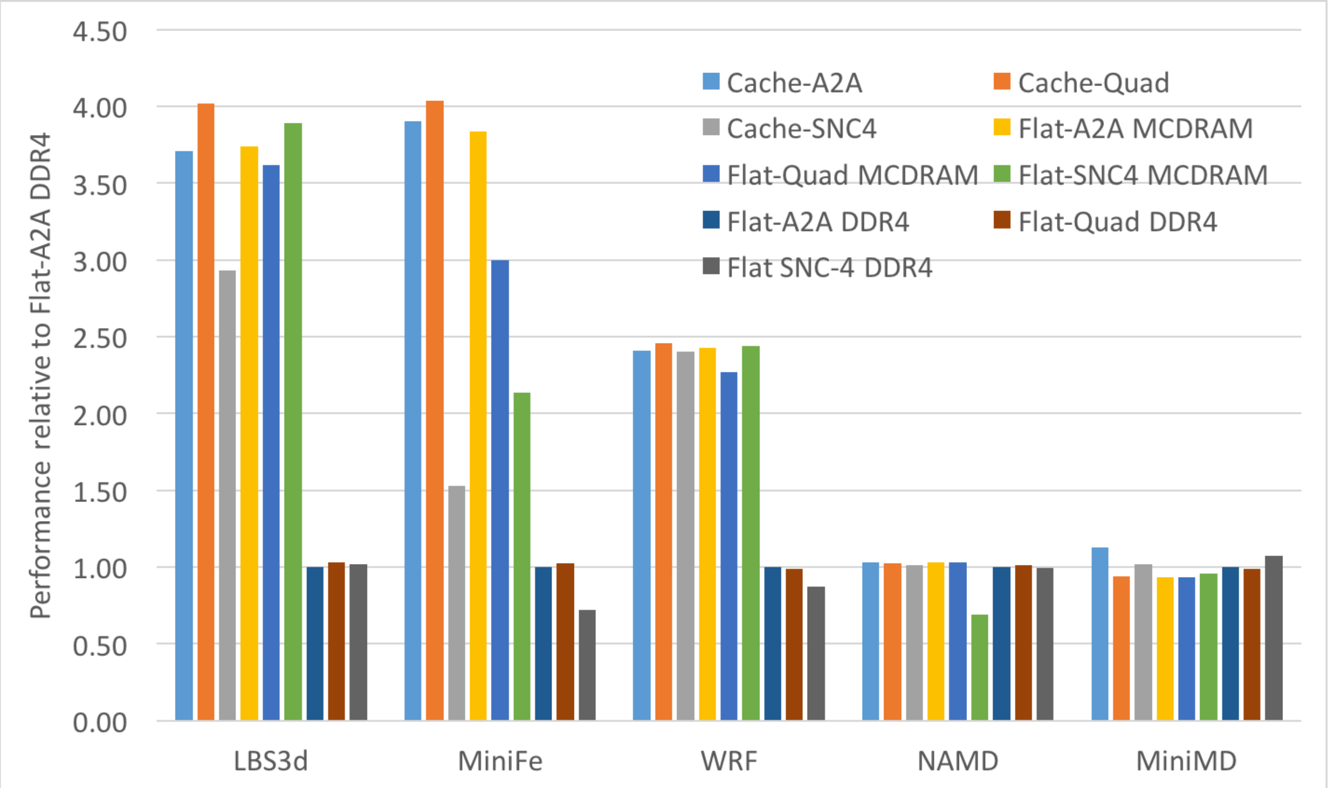
Cache Mode Performance Workarounds

- Without some workarounds, cache hit rate rapidly falls off as OS page free lists become randomized
  - Behavior approaches expected analytical result for random page assignment in a direct-mapped cache after several days of active use
- Intel XPSSL 1.4.1 provides a “free list sorter” as a workaround
  - Pages from the same 16 GiB memory region cannot conflict in the cache, so sorting the free lists into ascending order should reduce conflicts
  - Not entirely effective – especially with Transparent Huge Pages
- Pre-allocated Large Pages work better, but still show significant variability over time
- Rebooting before each job results in the best performance


Xeon Phi 7250 Cache-Quadrant Mode Bandwidth vs Usage and Workarounds
STREAM Triad Bandwidth (MB/s)
400,000
350,000
300,000
250,000
200,000
150,000
100,000
50,000
0
MCDRAM Cache Hit Rate
100%
75%
50%
25%
0%
0.00
2.00
4.00
6.00
8.00
10.00
12.00
14.00
16.00
Total Memory Used (GiB)
Baseline BW
Compact+Sort BW
Baseline Hit Ratio
Compact+Sort Hit Ratio

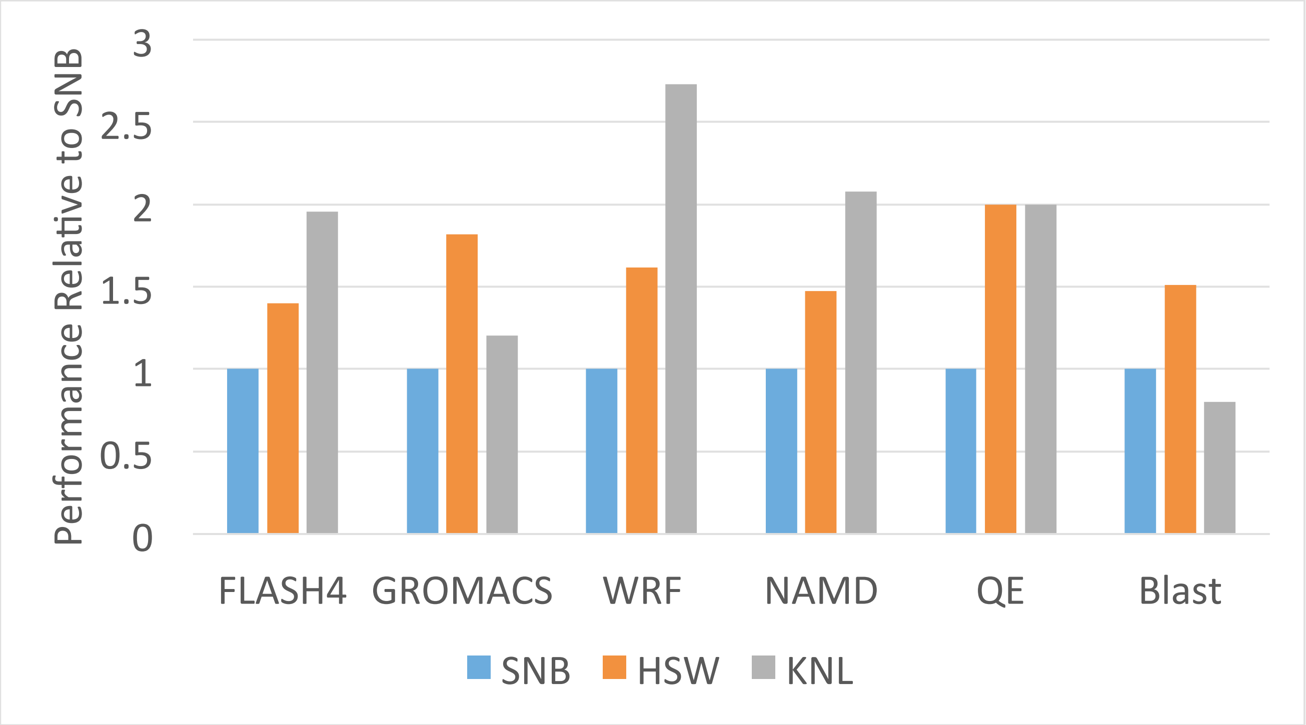

Xeon Phi 7250 Cache-Quadrant Mode Bandwidth vs Usage and Workarounds
STREAM Triad Bandwidth (MB/s)
400,000
350,000
300,000
250,000
200,000
150,000
100,000
50,000
0
MCDRAM Cache Hit Rate
100%
75%
50%
25%
0%
0.00
2.00
4.00
6.00
8.00
10.00
12.00
14.00
16.00
Total Memory Used (GiB)
Baseline BW
Compact+Sort BW
Static Large Page BW
Baseline Hit Ratio
Compact+Sort Hit Ratio
Static Large Page Hit Ratio

KNL Configuration Benchmarks (ISC 16)



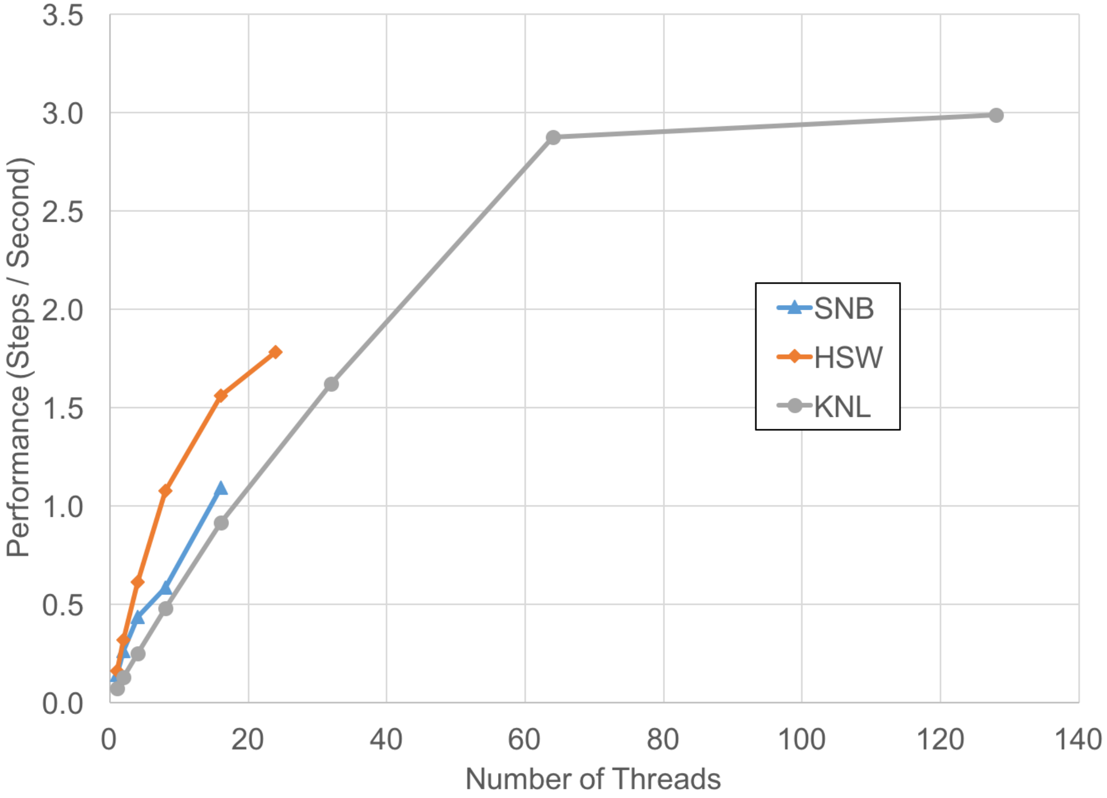
(higher is better)

Performance vs Stampede & Lonestar-5



(higher is better)

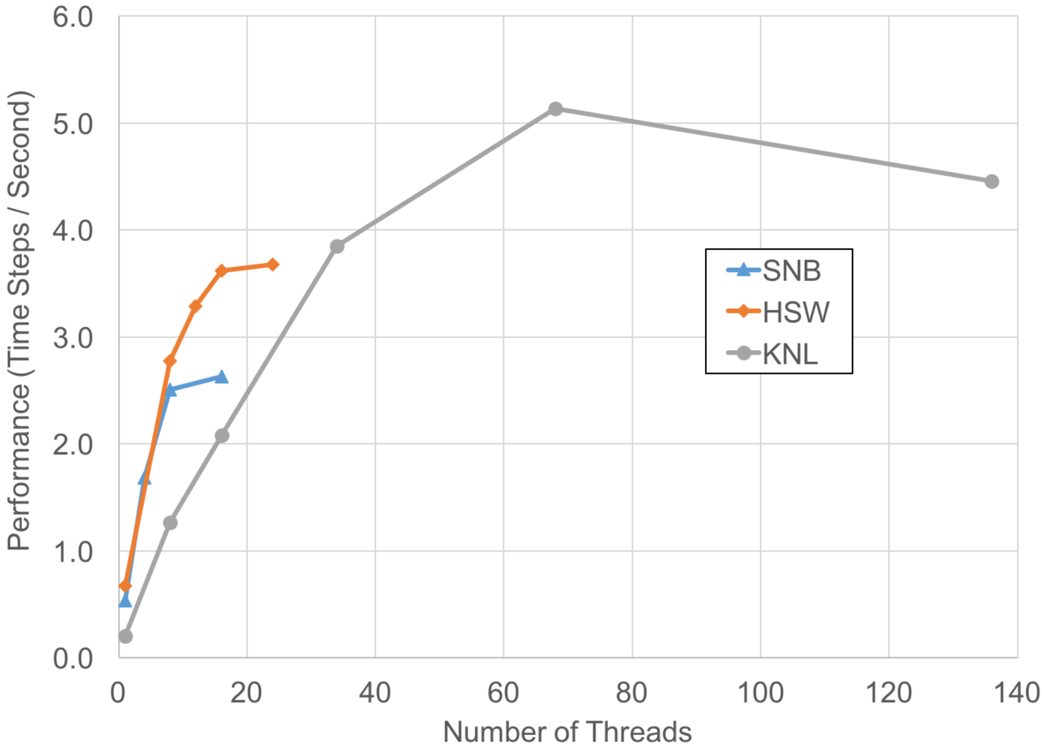
WRF Results



(higher is better)

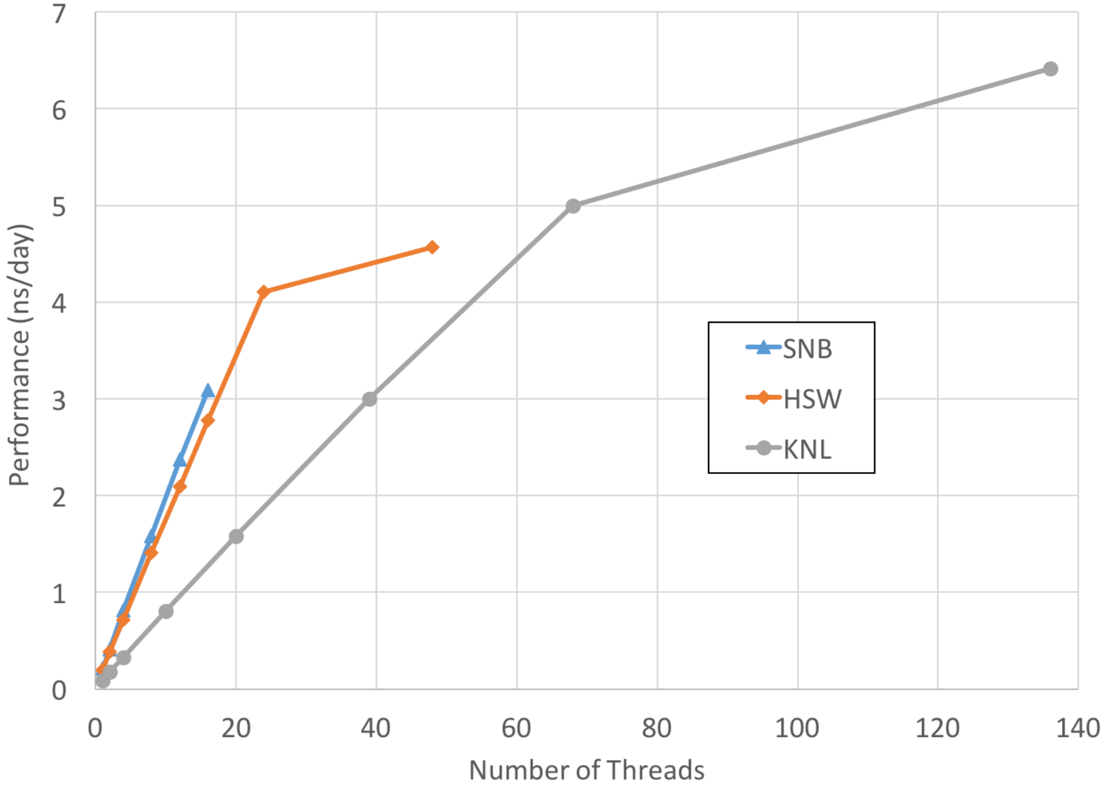
Flash Results

Minor modification to one loop to add OpenMP



(higher is better)

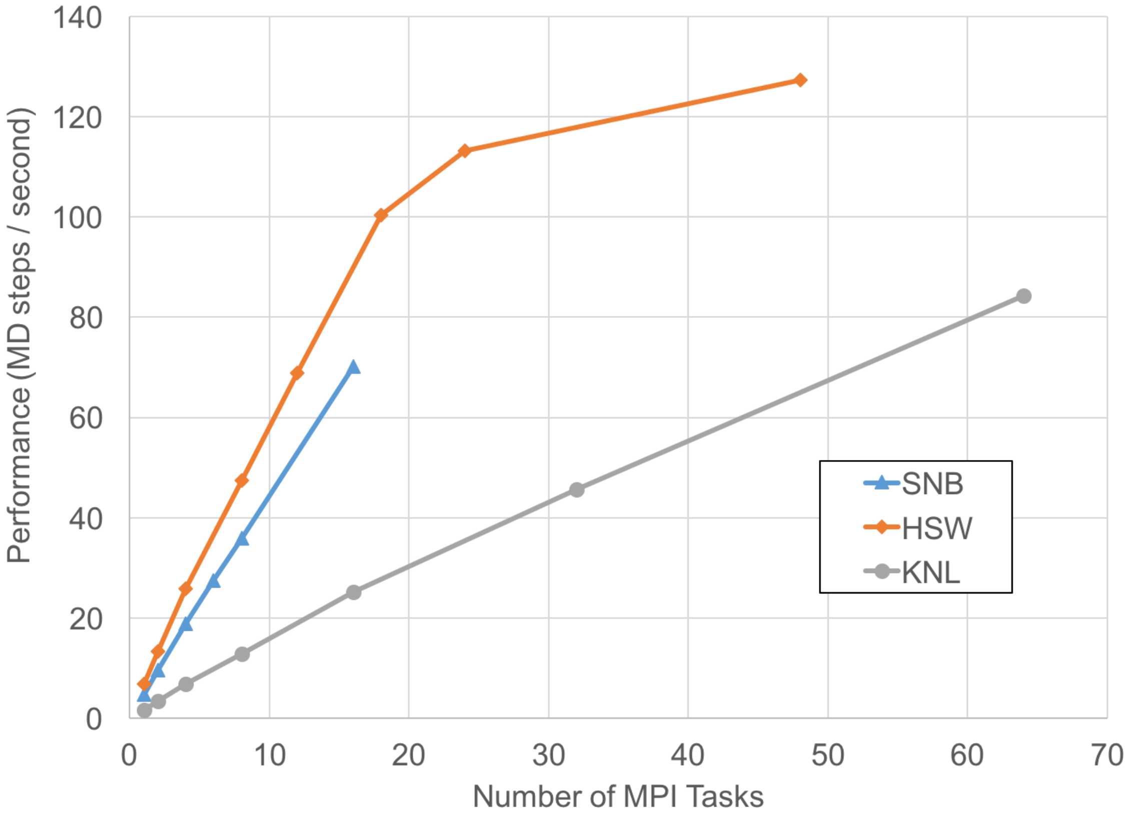
NAMD Results



(higher is better)

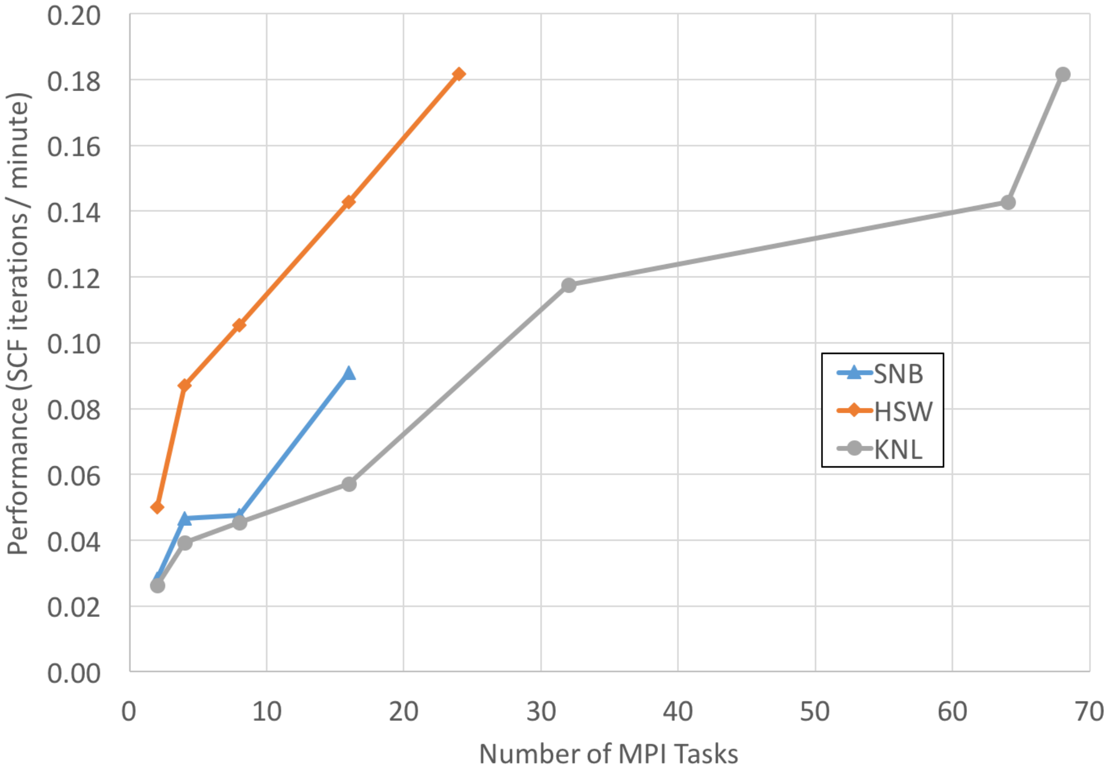
Gromacs Results

Change in the OMP part of the code to increase max thread count



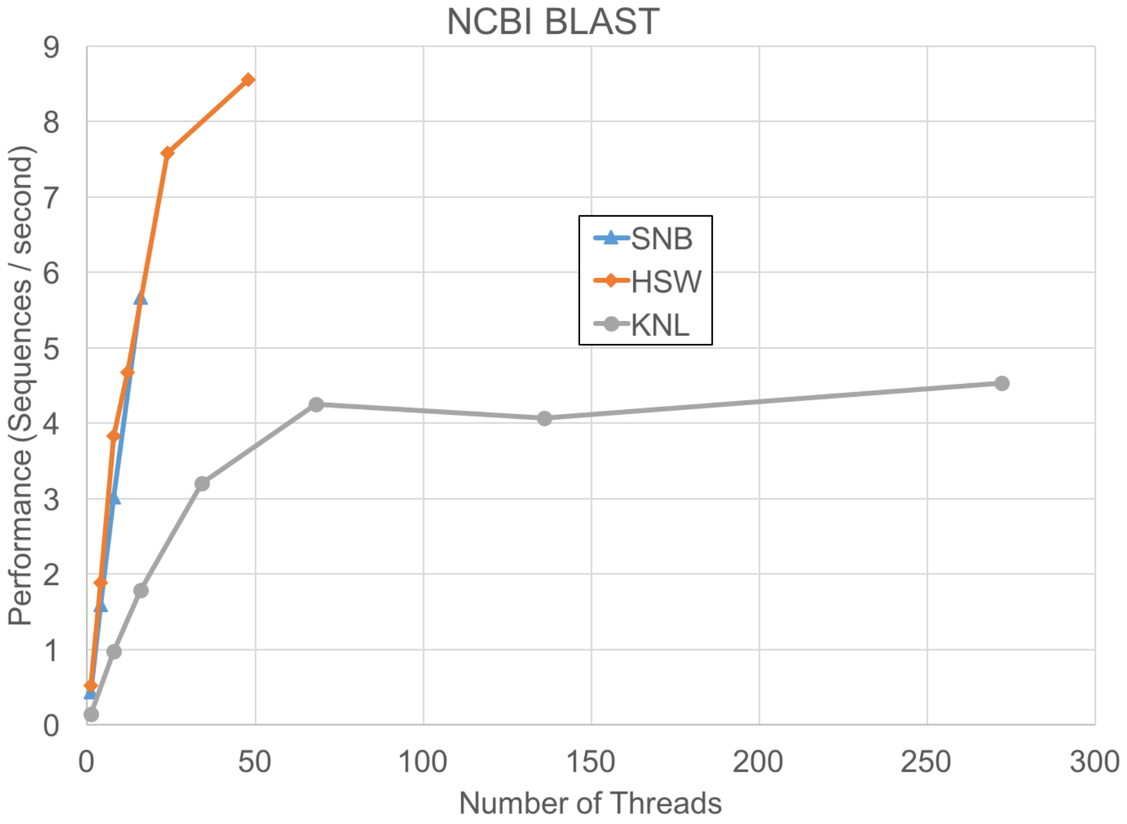
(higher is better)

Quantum Espresso Results



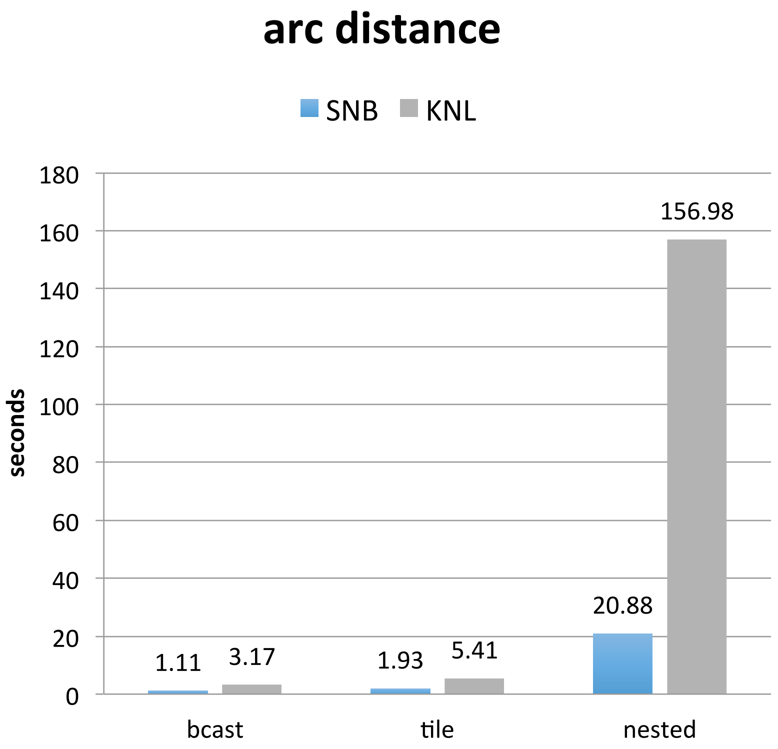
(higher is better)

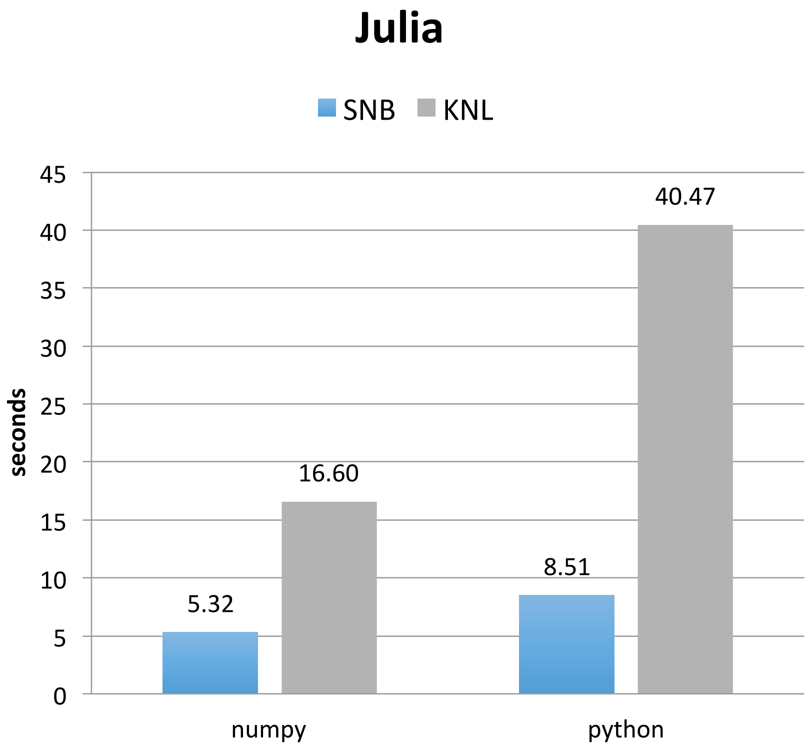
tblastx Results



(higher is better)

Python Results





(lower is better)
Note, this is per core – we have 4x more cores on KNL Nodes, so in some cases throughput is higher

NOT EVERYTHING FITS ON STAMPEDE

- Colin Morningstar is a long-time XSEDE user, working in Quantum Chromodynamics, a “traditional” simulation-intensive scientific discipline.
    - He uses several million compute hours per year on jobs that average 4.5hours apiece.
- Moving him from Sandy Bridge to top end Haswell made his code 1.5x faster 50% better.
- Moving his code to Wrangler’s filesystem made his runs 20 minutes, 13.5x faster or a an improvement of 1,350%.

- There are a variety of other problems in these categories – interactive usage,I/O intensive, GPU-friendly learning libraries, strange software stacks, persistent services, etc.

SOMETHING ELSE TO CONSIDER

- All of these results are on codes that have received immense tuning over decades (if not on this platform).
- In my experience, "normal" code is different:
  - It sucks much, much more.
  - So much so, low level hardware performance characteristics tell you nothing.
  - Arguing hardware tuning and modes can often be about 10 – 50%. Good coding practice is often 10,000-50,000%.
- Worry about having good code, then if you really want to worry about chips.
  - Tasks. Threads. Vectorization. Cache Blocking. ALGORITHM.

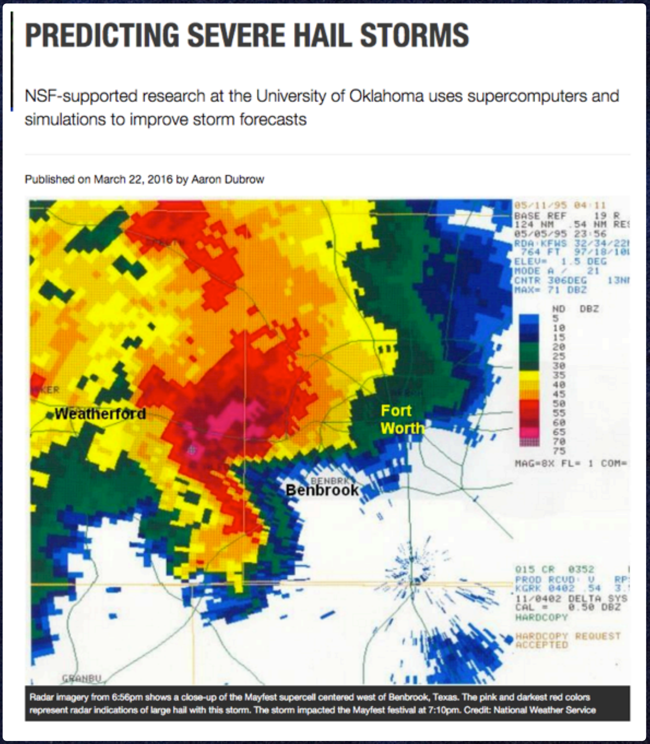
TOWARDS A COMPLETE ECOSYSTEM

- So, Stampede-2 will fit a whole bunch of people’s needs (10,000+, we hope), but there are still lots of other needs we need to fill through other systems.
- And, we have to deal with things beyond systems.

TODAY'S CYBERINFRASTRUCTURE PROBLEMS ARE COMPLEX

- It's almost always too simplistic to classify a problem as a "data problem", a "computing problem", a "machine learning" problem or a "cloud problem"
  - (or even, often, a technical problem – get a large group to adopt a new standard!!!).
- The kinds of challenges we face today usually blend many elements of what we call "Cyberinfrastructure".

E-SCIENCE ADVANCES RESEARCH PACE AND OUTCOMES



PREDICTING SEVERE HAIL STORMS

NSF-supported research at the University of Oklahoma uses supercomputers and simulations to improve storm forecasts

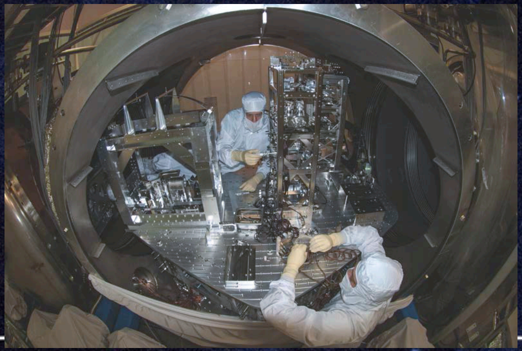
Published on March 22, 2016 by Aaron Dubrow

Radar imagery from 6:56pm shows a close-up of the Mayfest supercell centered west of Benbrook, Texas. The pink and darkest red colors represent radar indications of large hail with this storm. The storm impacted the Mayfest festival at 7:10pm. Credit: National Weather Service

BLENDING ENSEMBLE SIMULATIONS, DATA ASSIMILATION+INTEGRATION, AND MACHINE LEARNING EXTENDS HAIL FORECASTS FROM 2 HOURS TO 24 HOURS

EVEN TACC'S SMALL CONTRIBUTION ON LIGO WAS MORE THAN JUST COMPUTING

- LIGO discovered evidence of Gravitational Waves!
  - One of the most important physics/cosmology discovery in decades.
  - >3 Million core hours from TACC.
  - More important was the software support:
    - Doubled their efficiency on FFTs (and have since worked with them on further optimizations).
    - Improved their filesystem performance while supporting their OSG workflows.
    - Helped them change their internal culture on code performance!

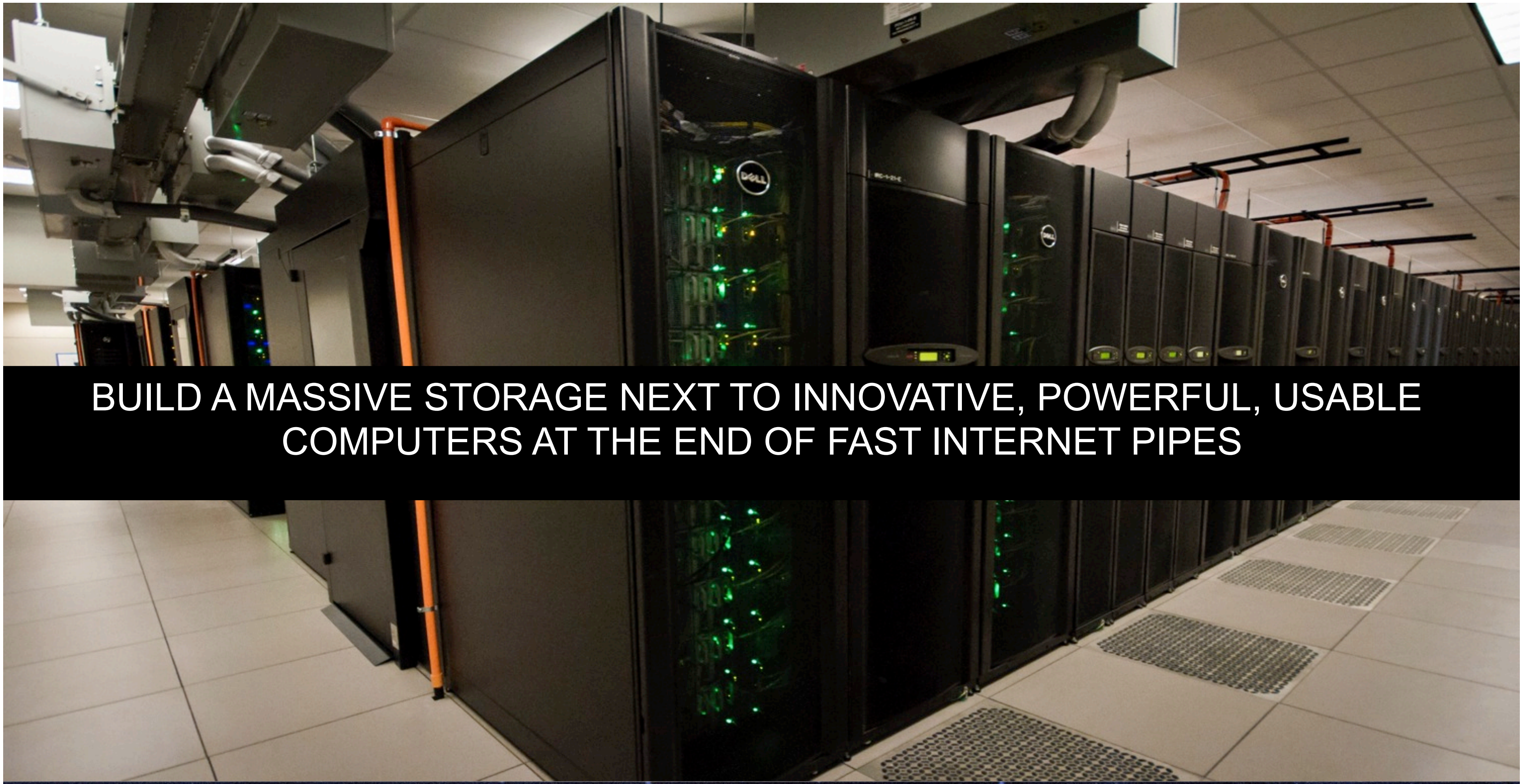


THE NEW E-SCIENCE

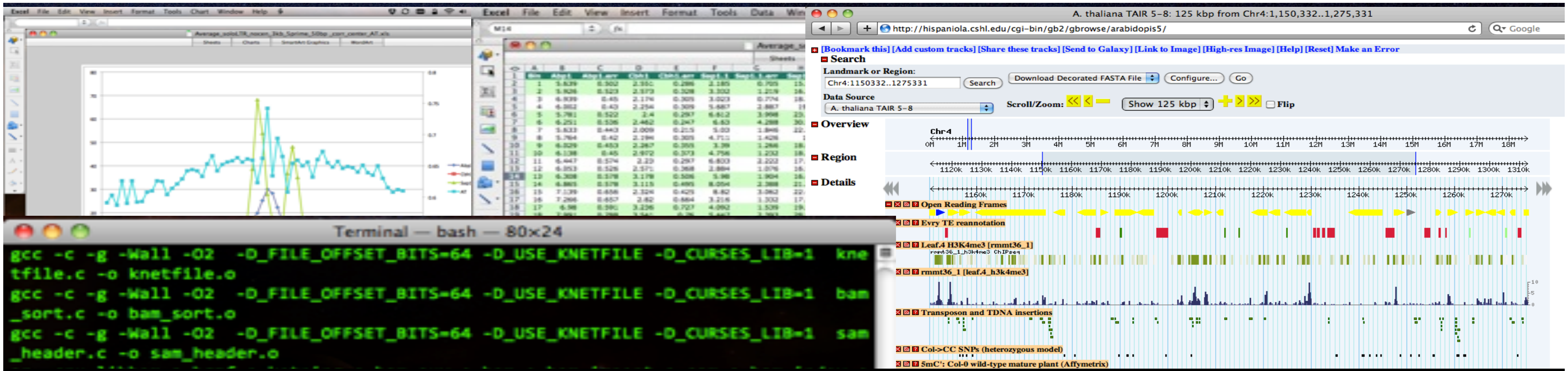
- Both the examples I just mentioned are largely driven by large scale computation – And computation centers have of course focused on these kinds of problems for decades, with tremendous success – but many new kinds of problems are not just about computing.

- The new E-science is largely a problem of integrating, at scale, data collection, curation, and storage with advanced computing and analysis (mining, visualization, machine learning).

NOT JUST ONE DATA TSUNAMI BUT THOUSANDS OF THEM



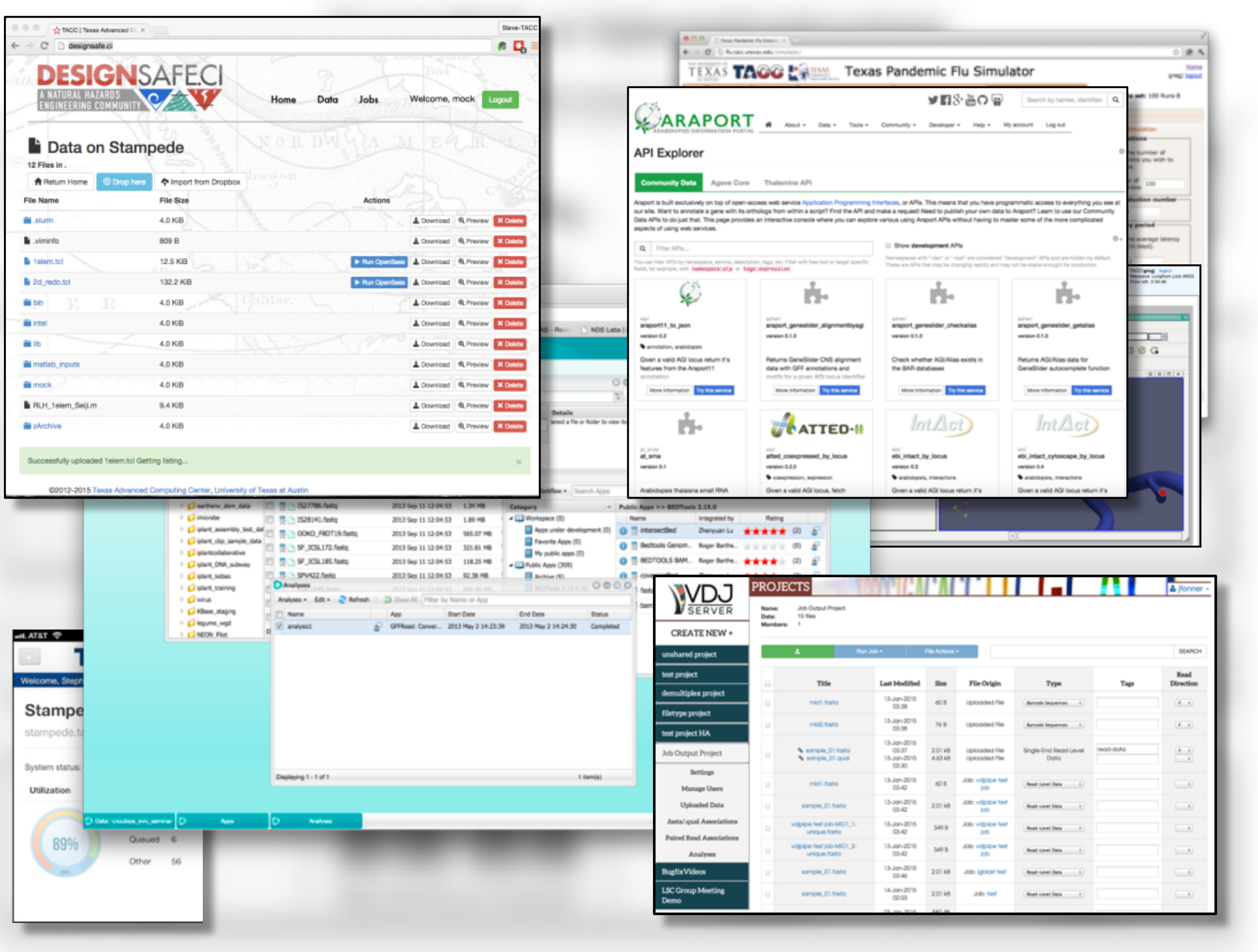
BUILD A MASSIVE STORAGE NEXT TO INNOVATIVE, POWERFUL, USABLE COMPUTERS AT THE END OF FAST INTERNET PIPES



```
gcc -c -g -Wall -O2  -D_FILE_OFFSET_BITS=64 -D_USE_KNETFILE -D_CURSES_LIB=1  kne
tfile.c -o knetfile.o
gcc -c -g -Wall -O2  -D_FILE_OFFSET_BITS=64 -D_USE_KNETFILE -D_CURSES_LIB=1  bam
_sort.c -o bam_sort.o
gcc -c -g -Wall -O2  -D_FILE_OFFSET_BITS=64 -D_USE_KNETFILE -D_CURSES_LIB=1  sam
_header.c -o sam_header.o
```

Terminal — bash — 80×24
A. thaliana TAIR 5-8: 125 kbp from Chr4:1,150,332..1,275,331
http://hispaniola.cshl.edu/cgi-bin/gb2/gbrowse/arabidopsis5/
[Bookmark this] [Add custom tracks] [Share these tracks] [Send to Galaxy] [Link to Image] [High-res Image] [Help] [Reset] Make an Error
Search
Landmark or Region:
Chr4:1150332..1275331
Search
Download Decorated FASTA File
Configure...
Go
Data Source
A. thaliana TAIR 5-8
Scroll/Zoom:
Show 125 kbp
Flip
Overview
Region
Details
Chr4
Open Reading Frames
Evry TE reannotation
Leaf.4 H3K4me3 [rmmt36_1]
rmmt36_1 [leaf.4_h3k4me3]
Transposon and TDNA insertions
Col>CC SNPs (heterozygous model)
5mC': Col-0 wild-type mature plant (Affymetrix)

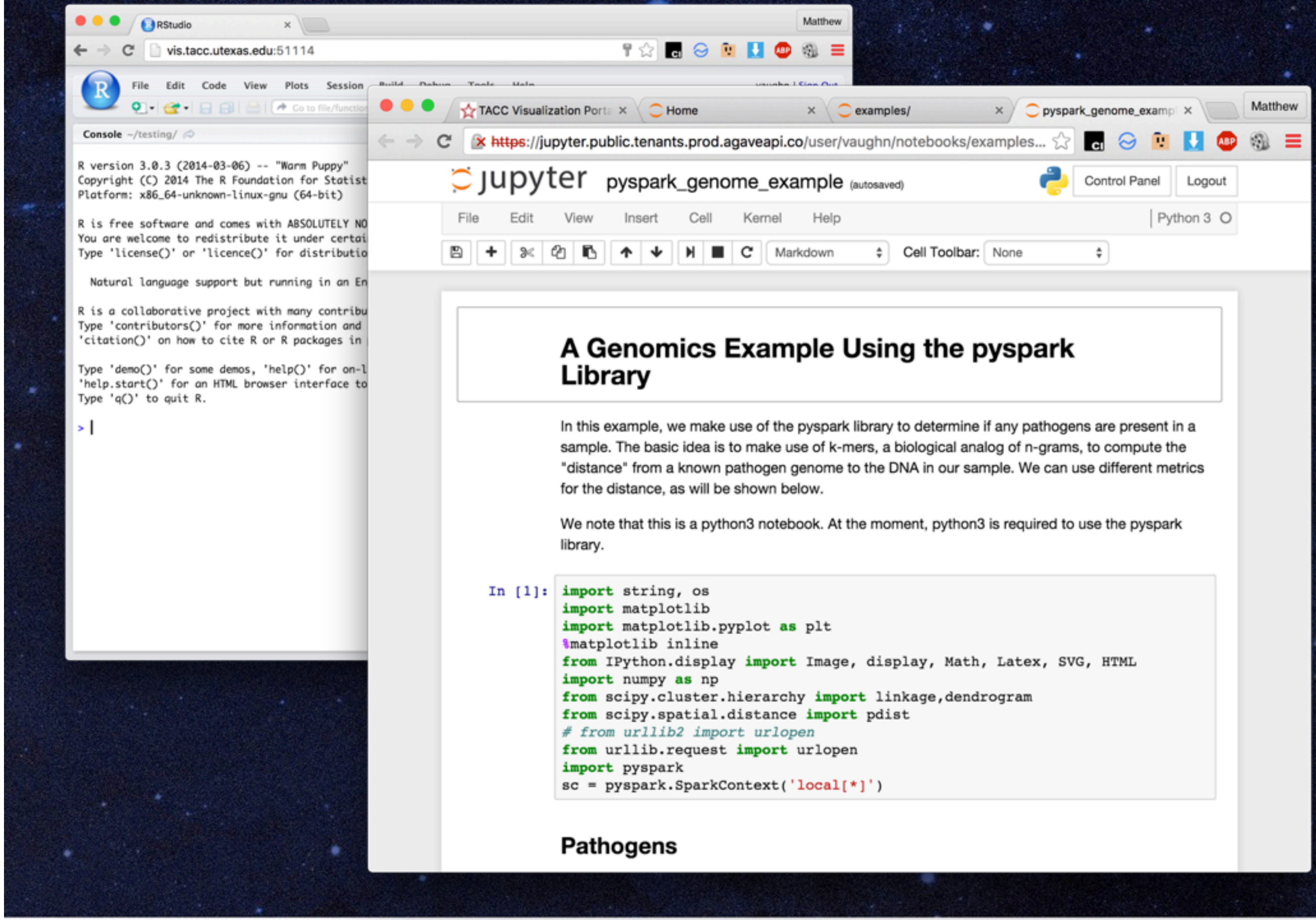
MANY DOMAIN SCIENTISTS ARE NOT EXPERTS AT COMPUTING TECHNOLOGY. CREATE PURPOSE-BUILT, SECURED, HIGHLY INTUITIVE ENVIRONMENTS



Point-and-click interfaces

- Data management, sharing, and analysis
- Publishing reproducible analysis workflows
- Discovery of new or updated tools and data
- Interactive visualization of results

Backed by world-class computing and data capacity


RStudio
vis.tacc.utexas.edu:51114
Matthew
File Edit Code View Plots Session
Console ~/testing/
TACC Visualization Porta
Home
examples/
pyspark_genome_examp
https://jupyter.public.tenants.prod.agaveapi.co/user/vaughn/notebooks/examples...
jupyter
pyspark_genome_example (autosaved)
Control Panel
Logout
File Edit View Insert Cell Kernel Help
Python 3
Markdown
Cell Toolbar: None

```
R version 3.0.3 (2014-03-06) -- "Warm Puppy"
Copyright (C) 2014 The R Foundation for Statist
Platform: x86_64-unknown-linux-gnu (64-bit)

R is free software and comes with ABSOLUTELY NO
You are welcome to redistribute it under certai
Type 'license()' or 'licence()' for distributio

  Natural language support but running in an En

R is a collaborative project with many contribu
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in

Type 'demo()' for some demos, 'help()' for on-l
'help.start()' for an HTML browser interface to
Type 'q()' to quit R.

> |
```

A Genomics Example Using the pyspark Library

In this example, we make use of the pyspark library to determine if any pathogens are present in a sample. The basic idea is to make use of k-mers, a biological analog of n-grams, to compute the "distance" from a known pathogen genome to the DNA in our sample. We can use different metrics for the distance, as will be shown below.

We note that this is a python3 notebook. At the moment, python3 is required to use the pyspark library.

In [1]:

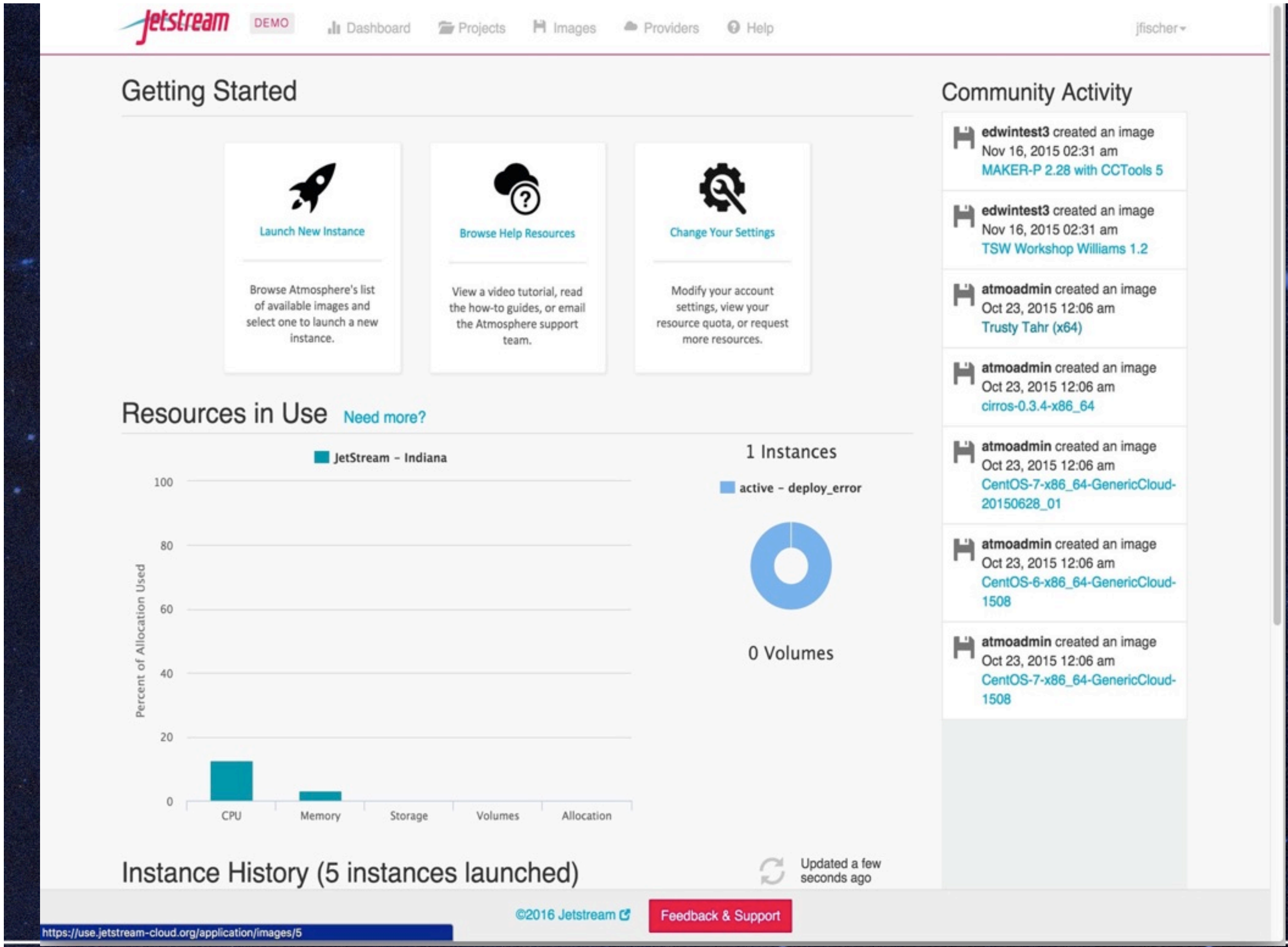
```python
import string, os
import matplotlib
import matplotlib.pyplot as plt
%matplotlib inline
from IPython.display import Image, display, Math, Latex, SVG, HTML
import numpy as np
from scipy.cluster.hierarchy import linkage,dendrogram
from scipy.spatial.distance import pdist
# from urllib2 import urlopen
from urllib.request import urlopen
import pyspark
sc = pyspark.SparkContext('local[*]')
```

Pathogens

Hosted SaaS

- JupyterHub notebooks
- Rstudio
- Web-based VNC

Also, backed by world-class computing and data capacity



Getting Started

Launch New Instance: Browse Atmosphere's list of available images and select one to launch a new instance.

Browse Help Resources: View a video tutorial, read the how-to guides, or email the Atmosphere support team.

Change Your Settings: Modify your account settings, view your resource quota, or request more resources.

Resources in Use

1 Instances

0 Volumes

Instance History (5 instances launched)

Updated a few seconds ago

Community Activity

edwintest3 created an image
Nov 16, 2015 02:31 am
MAKER-P 2.28 with CCTools 5

edwintest3 created an image
Nov 16, 2015 02:31 am
TSW Workshop Williams 1.2

atmoadmin created an image
Oct 23, 2015 12:06 am
Trusty Tahr (x64)

atmoadmin created an image
Oct 23, 2015 12:06 am
cirros-0.3.4-x86_64

atmoadmin created an image
Oct 23, 2015 12:06 am
CentOS-7-x86_64-GenericCloud-20150628_01

atmoadmin created an image
Oct 23, 2015 12:06 am
CentOS-6-x86_64-GenericCloud-1508

atmoadmin created an image
Oct 23, 2015 12:06 am
CentOS-7-x86_64-GenericCloud-1508

Easy to use Cloud Computing

- Atmosphere (Cyverse)
- Jetstream (IU,UA,TACC)
- Chameleon (UC,TACC)

Cloud consoles are aimed at sysadmins and unintutive.

We're changing that with improved UX and support

- APIs are still available
- No cost to end user

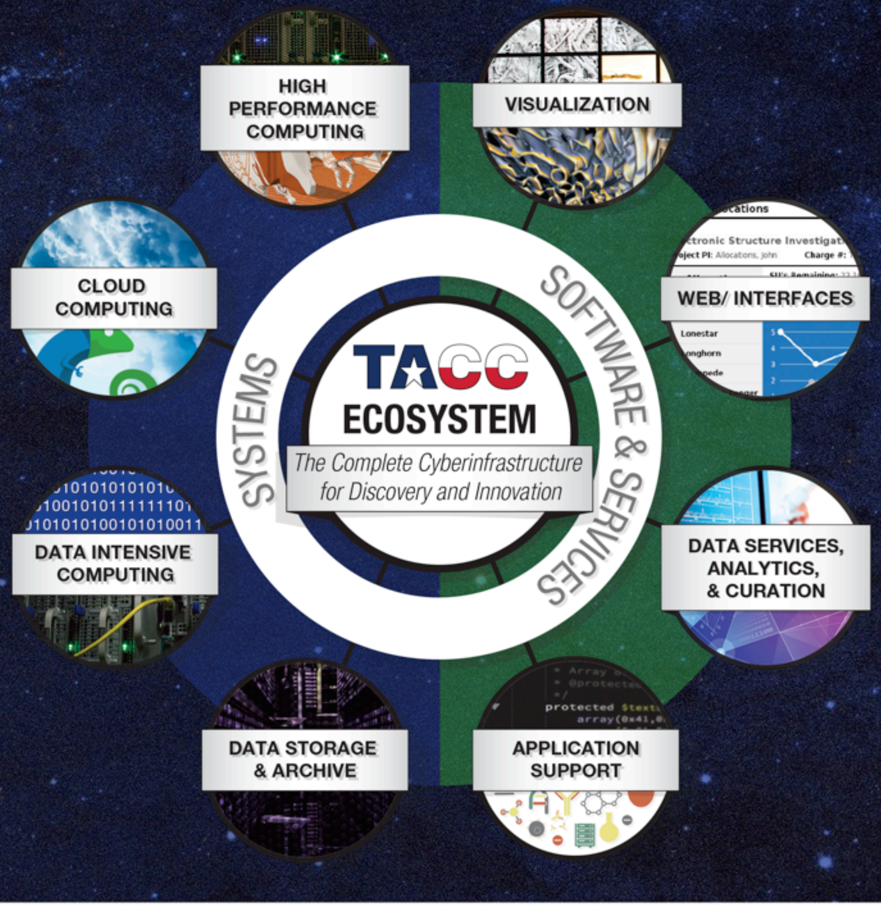
GIVE EXPERTS ACCESS TO EVERY SINGLE ONE OF YOUR BUILDING BLOCKS. WEB SERVICE APIs EVERYWHERE. AUGMENT WITH PROFESSIONAL TOOLING.

EXPERT STAFF

160+ team members

Direct, relevant experience in hundreds of technologies and research topics

We do thousands of hours of consulting, teaching, training, research, and outreach every year



ALL THIS RESEARCH (AND MORE) IS ENABLED BY A COMPREHENSIVE CYBERINFRASTRUCTURE ECOSYSTEM AT TACC

BUT ALL THE PIECES DO NOT MAKE A FINISHED PRODUCT

Happy 25th Birthday IKEA!
Here's your Cake.

No Software System in History has ever worked together accidentally!

EVEN IF YOU HAVE ALL THE SYSTEMS, YOU DON'T HAVE ALL YOU NEED. . .

- You need a coherent CI Architecture to fit the pieces together.
  - Usually built for each different domain in close collaboration with domain experts, but the principles can get reused.

PULLING IT TOGETHER

- We are going to be able to keep building much larger and faster systems, for a long time as we stay on this path to (T)exascale.
  - But, as has been the trend for a while, *real* performance is *not* transparent to software.   Peak is increasingly irrelevant.
- The most fundamental principle of software engineering is abstraction.
  - But our old abstractions have begun to fail us. SW needs to know about HW.
- We will continue to leap forward in E-Science (and bring back our wayward unruly teenage offspring, Data Science and AI).
  - But, we need a *comprehensive* approach to do this at scale – hardware, software, services, people - and we need to rebuild our abstractions.
- **Also, come use our new machine(s)   ☺.**

Firefly Space Systems

Firefly, a TACC industrial partner, will begin engine tests shortly on rocket engines designed using Stampede.

Thanks!

dan@tacc.utexas.edu