TACC

TEXAS ADVANCED COMPUTING CENTER

WWW.TACC.UTEXAS.EDU

TEXAS

The University of Texas at Austin

How Not to Get Trampled by the Stampede in a Cloudy Season

PRESENTED BY:

Dan Stanzione

Stampede 2

Funded by NSF as a renewal of the original Stampede project.

The largest XSEDE resource (and largest university-based system).

Follow the legacy of success of the first machine as a supercomputer for a *broad* range of workloads, large and small.

Install without ever having a break in service – in the same footprint.

Stampede 2 -- Components

Phase 1

- 4,204 Intel Xeon Phi ”Knights Landing” (KNL) Processors (Intel and Dell)
- ~20PB (usable) Lustre Filesystem (Seagate), 310GB/s to scratch.
- Intel Omnipath Fabric – Fat Tree.
- Ethernet fabric and (some) management infrastructure.

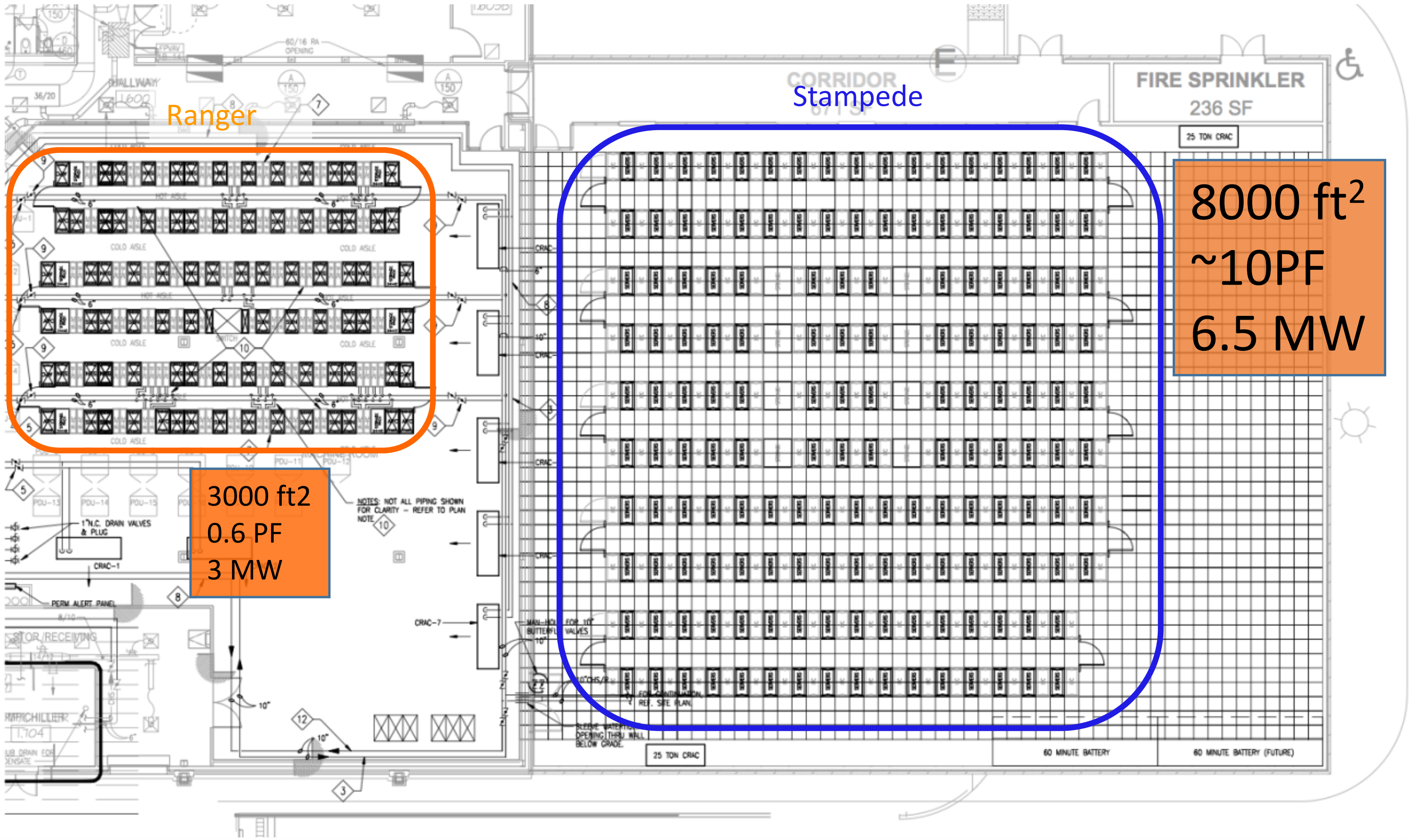
Phase 2

- 1,736 Intel Xeon (Sky Lake) Processors
- (Associated networking, but core in phase 1).
- Balance of management hardware.

Phase 3

- 3D Crosspoint DIMMS as an experimental component in a small subset of the system.

Stampede Footprint



Ranger

3000 ft2
0.6 PF
3 MW

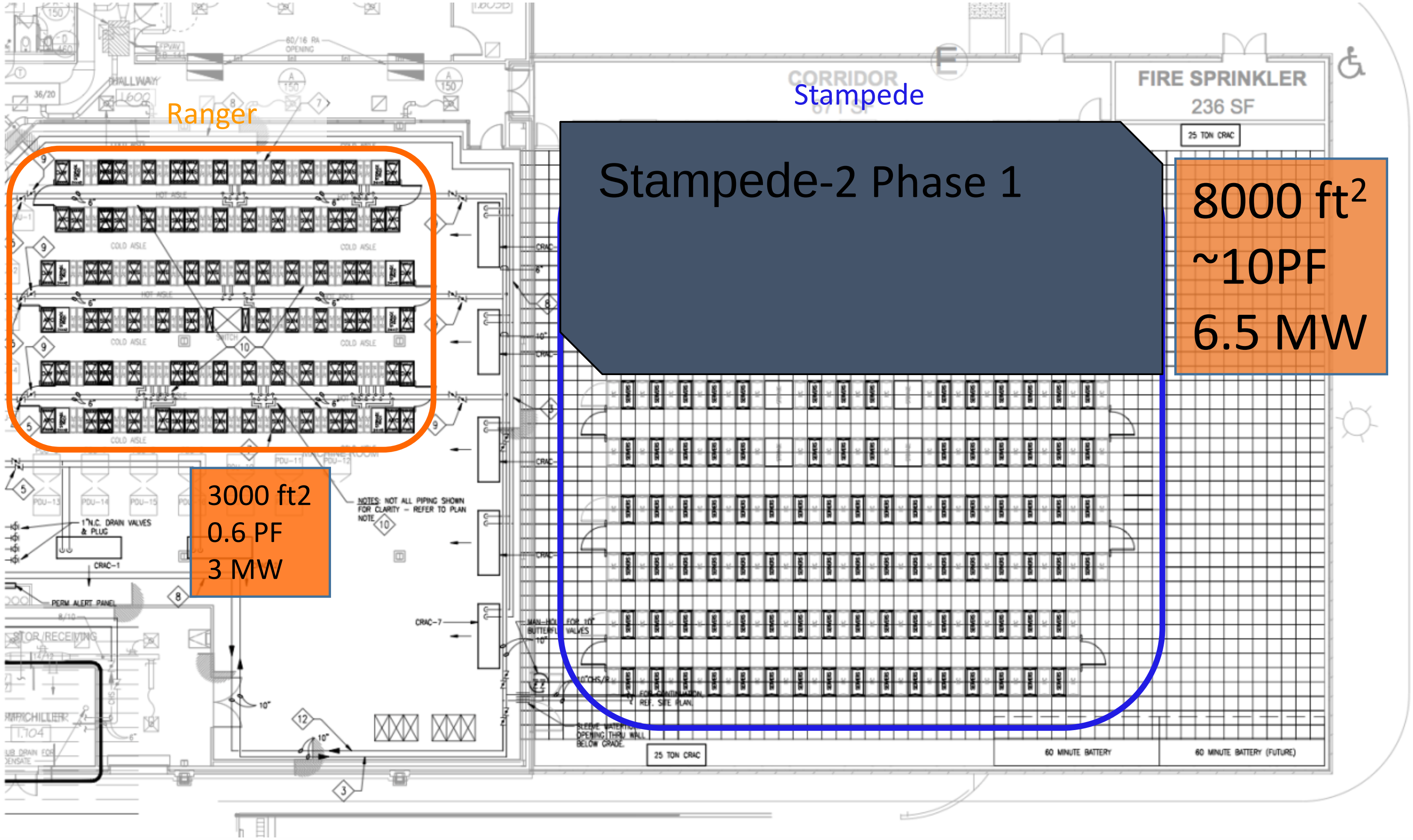
Stampede

8000 ft^2
~10PF
6.5 MW

Machine Room Expansion

Added 6.5MW of additional power

Stampede Footprint



Machine Room Expansion

Added 6.5MW of additional power

Hardware Overview

Stampede 2 Phase 1 compute nodes

- 924 Dell C6320P chassis, 4 nodes per chassis
  - 3,696 total compute nodes
    - Intel Xeon Phi 7250 CPU, 68 cores, 1.4GHz
    - 96 GB (6x16GB) 2400MHz DDR4
    - 200 GB SSD
  - Redundant 1600W power supplies
- 126 Intel PCSD chassis, 4 nodes per chassis
  - 508 total compute nodes
    - Intel Xeon Phi 7250 CPU, 68 cores, 1.4GHz
    - 96 GB (6x16GB) 2400MHz DDR4
    - 120 GB SSD

Early results

Yifeng Cui at SDSC has sped up SCEC's primary code to get a 5x performance improvement over Haswell (competitive for a P100).

Martin Berzins' team at Utah has scaling the UINTAH combustion code, outperforming Titan on the original version, and showing algorithmic improvements that can yield even more.

Rommie Amaro at SDSC is using KNL to investigate the structure, function, and dynamics of complex biological systems.

Omar Ghattas and team have recently ported in the Gordon Bell winning code from 2015 and a finalist from 2013, with good results.

George Biros has a fast solver that achieves 1.4TF per node – and scales!

Dan Bodony at Illinois is doing a massive study of for Prediction and Control of of Compressible Turbulent Flows (request equivalent to all KNL nodes for 3 months).

Real Time severe storm prediction with Ming Xue at Oklahoma has migrated to Stampede-2

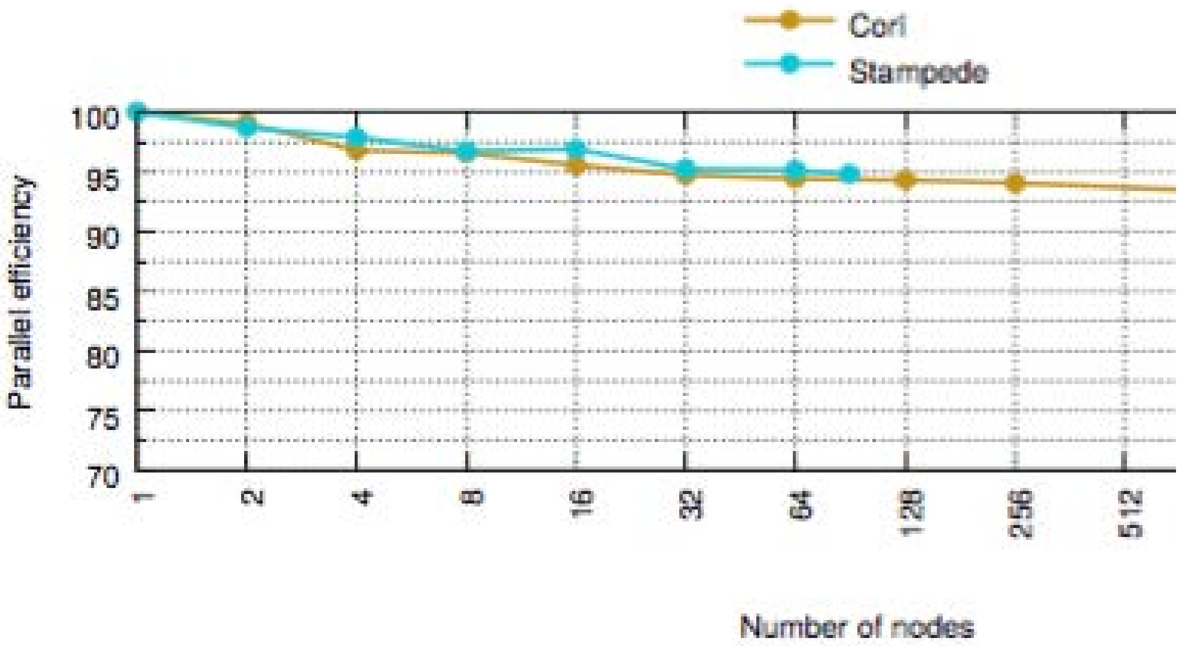
New R benchmarks have been published by IU Colleagues– with KNL reference data.

Our software defined visualization stack supported six teams in a workshop in the early science phase coupling visualization to simulation – including Stephen Hawking's team at Cambridge.

Omnipath Scaling

Seems just fine

- comparison to KNL-Cray at NERSC, just to 128 nodes (on a seismic code).
- From an ISC paper by Yifeng Cui, didn’t have higher node counts up in time to publish



Early Results --Generalizations

Everything runs, but. . .

- Carefully tuned codes are doing pretty well, but with work.
- “Traditional” MPI codes, especially with OpenMP in it do relatively well, but not great.
- Some codes, particularly, not very parallel ones, are pretty slow, and probably best run on Xeon.

Mileage varies widely – but raw performance isn’t the whole story.

Our Experience with Xeon Phi

Xeon Phi looks to be the most cost and power efficient way to deliver performance to highly parallel codes.

In many cases, it will not be the fastest. For things that only scale to a few threads, it is *definitely* not the fastest.

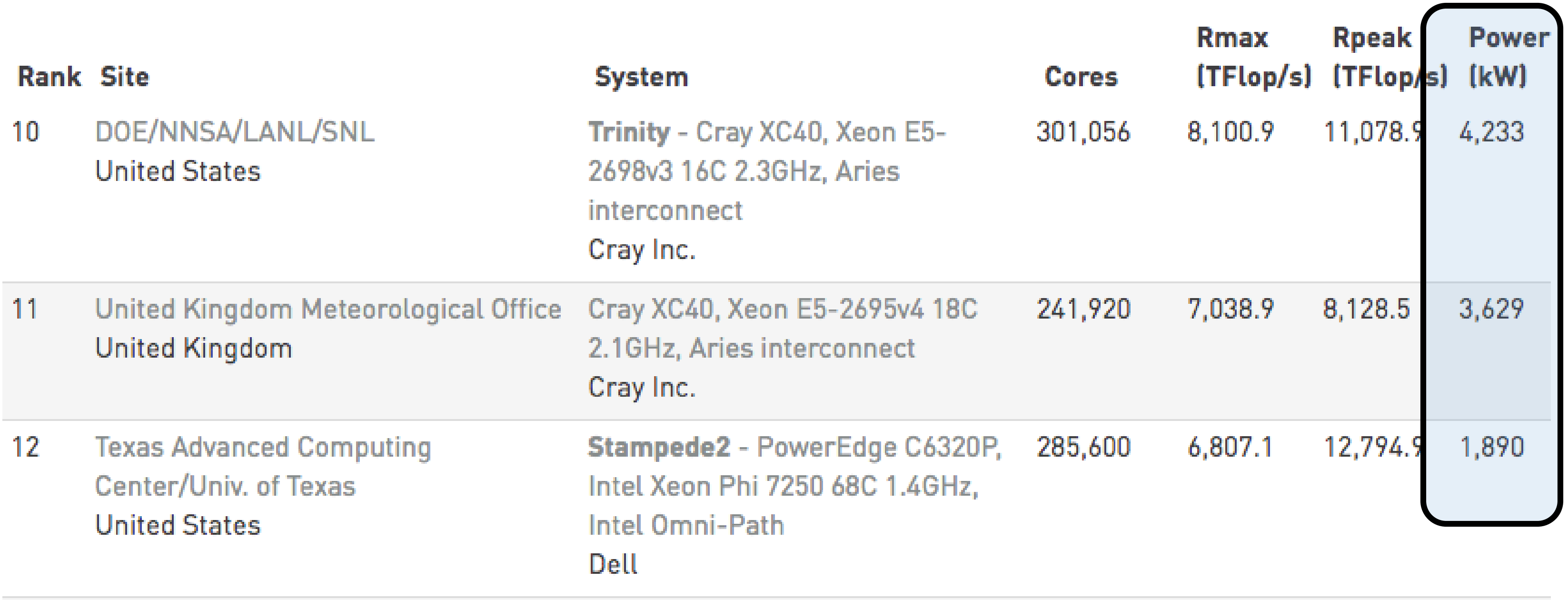
But what is under-discussed:

- As far as I can tell, a Dual-socket Xeon node costs 1.6x what a KNL node costs, even after discounts.
  - A dual-socket, dual GPU nodes is probably >3x a Xeon Phi node.
- A KNL node uses about 100 less watts per node than a dual-socket Xeon node.

Power from Top 500

List unveiled in June at ISC17 in Frankfurt

Stampede-2 uses half the power of a roughly equivalent performance system (see 11 vs 12)



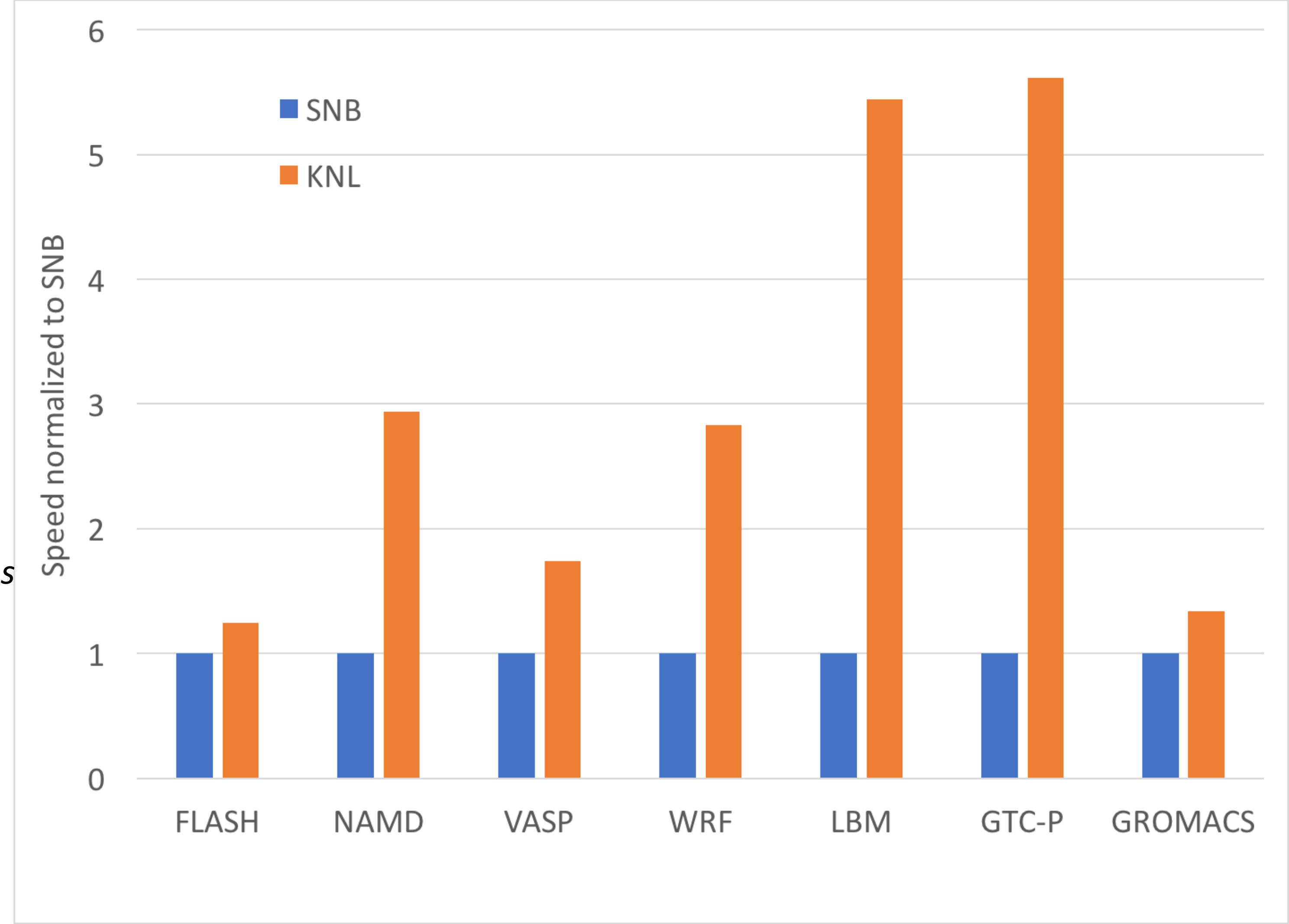
Rank	Site	System	Cores	Rmax (TFlop/s)	Rpeak (TFlop/s)	Power (kW)
10	DOE/NNSA/LANL/SNL United States	**Trinity** - Cray XC40, Xeon E5-2698v3 16C 2.3GHz, Aries interconnect Cray Inc.	301,056	8,100.9	11,078.9	4,233
11	United Kingdom Meteorological Office United Kingdom	Cray XC40, Xeon E5-2695v4 18C 2.1GHz, Aries interconnect Cray Inc.	241,920	7,038.9	8,128.5	3,629
12	Texas Advanced Computing Center/Univ. of Texas United States	**Stampede2** - PowerEdge C6320P, Intel Xeon Phi 7250 68C 1.4GHz, Intel Omni-Path Dell	285,600	6,807.1	12,794.9	1,890

Reference Application Graph

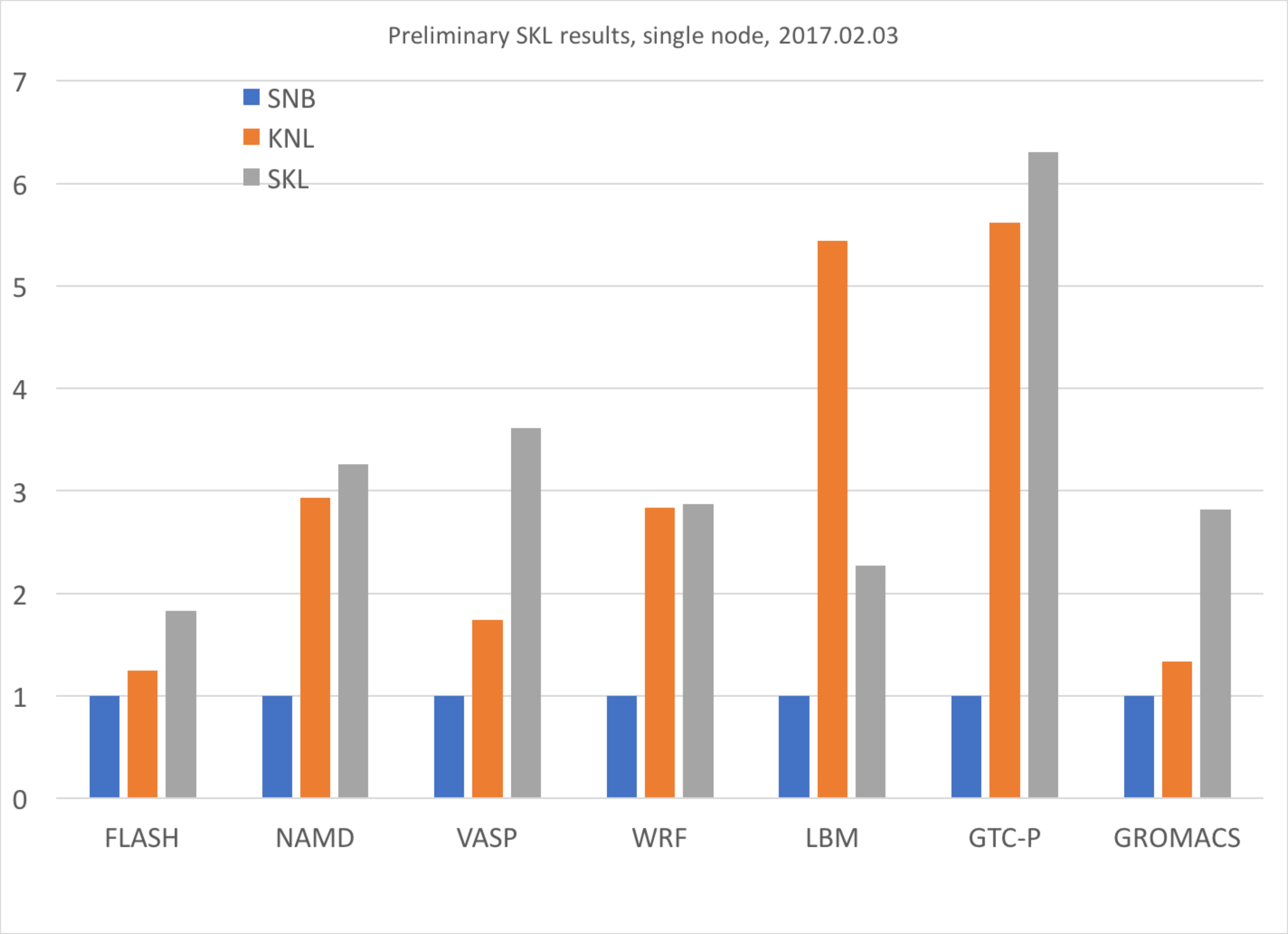
Measured Performance between Stampede and Stampede-2 nodes.

Codes are unmodified versions, as checked out from the source repository, to accurately reflect the "average user" experience.

(There is nothing less fair than comparing optimized codes on one platform to un-optimized codes on a different platform)



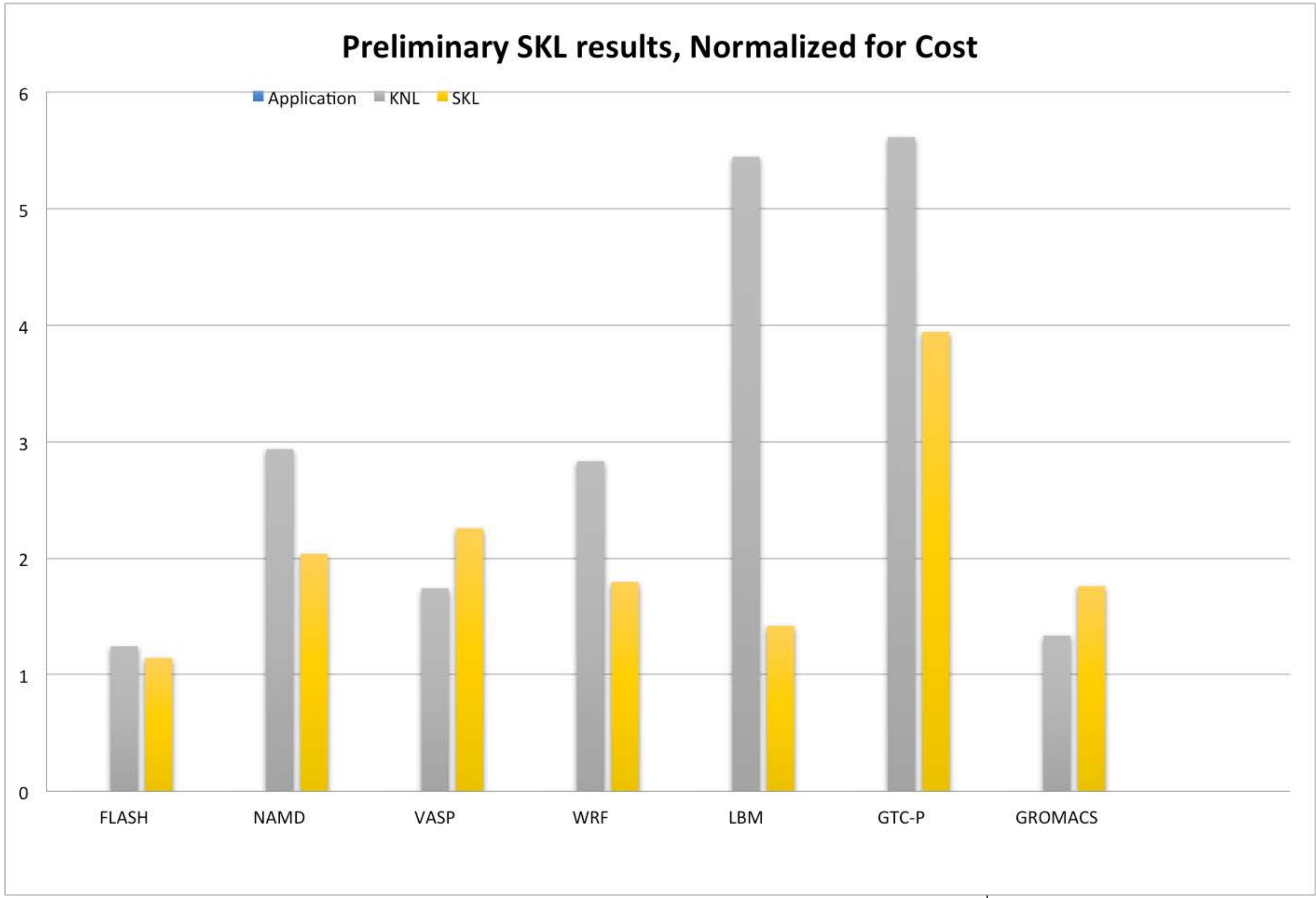
Reference Application Graph



Reference Application Graph –Normalized for Cost

This compares Xeon Phi Performance on major parallel applications to our projections for “Sky Lake”, the next generation Xeon, in a dual-socket configuration with top bin parts.

The KNL wins on price-performance in many applications with *no* optimization (but not always in absolute performance!).



Our Experience with Xeon Phi

Xeon Phi looks to be the most cost and power efficient way to deliver performance to highly parallel codes.

Is “fastest” our only metric? Are we maximizing for an individual user, or most Science &Engineering computing output for the amount of investment?

In straight Xeon, we would have had at least a thousand less nodes.

TEAM

Literally, everyone at TACC!!! – and a lot of folks at Dell, Intel, and Seagate – and our academic partners.

Co-Pis : Tommy Minyard, Bill Barth, Niall Gaffney, Kelly Gaither.

Deployment, Ops, Security:

Laura Branch, Dennis Byrne, Sean Hempel, Nathaniel Mendoza, Patrick Storm, Freddy Rojas, David Carver, Nick Thorne, Dave Cooper, Frank Duomo, Je’aime Powell, Peter Lubbs, Sergio Leal, Jacob Getz, Lucas Nopoulos, Garland Whiteside, Matthew Edeker, Dave Littrell, Remy Scott

HPC, Data, Vis, and Life Sciences App support

John Cazes, Cyrus Proctor, Robert McLay, Ritu Arora, Todd Evans, Si Liu, Hang Liu, Lars Koersterke, Victor Eikhout, Lei Huang, Kevin Chen, Doug James, Antonio Gomes, Kent Milfeld, Jerome Vienne, Virginia Tueheart, Antia Limas-Lanares, John McCalpinWeijia Xu, Chris Jordan, David Walling, Siva Kula, Amit Gupta, Maria Esteva, John Gentle, Suzanne Pierce, Ruizhu Huang, Tomislav Urban, Zhao Zhang, Paul, Navratil, Anne Bowen, Greg Foss, Greg Abram, Jaoa Barbosa, Luis Revilla, Craig Jansen, Brian McCann, Ayat Mohammed, Dave Semararo, Andrew Solis, Jo Wozniak, Joe Allen, Erik Ferlanti, Brian Beck, James Carson, John Fonner, Ari Kahn, Jawon Song, Matt Vaughn, Greg Zynda,

Education, Outreach, Training

Rosie Gomez, Joon-Yee Chuah, Dawn Hunter, Luke Wilson, Jason Allison, Charlie Dey

Web Services

Maytal Dahan, Steve Mock, Rion Dooley, Josue Coronel, Alex Rocha, Carrie Arnold, David Montoya, Mike Packard, Cody Hammock, Mike Keller, Joe Stubbs, Tracy Brown, Rich Cardone, Steve Terry, Andrew Magill, Joe Meiring, Marjo Poindexter, Juan Ramirez, Harika Gurram

User Support, PM, Admin

Chris Hempel, Natalie Henriques, Tim Cockerill, Bryan Snead, Janet Mccord, Dean Nobles, Susan Lindsay, Akhil Seth, Bob Garza, Marques Bland, Karla Gendler, Valori Archuleta, Suzanne Bailey, Paula Baker, Janie Bushn, Katie Cohen, Sean Cunningham, Aaron Dubrow, Dawn Hunter, Shein Kim, Hedda Prochska, Jorge Salazar, Matt Stemalszak, Faith Singer, Arleen Umbay, Valerie Wise, Ashley Bucholz, Melyssa Fratkin, Manu John, John West

Demands on HPC Centers are Changing

- Ten or 15 years ago, we needed to buy and run a machine, and give users command line access to it.
  - Then occasionally help them parallelize their code, or fix their I/O
  - Maybe work with them on visualization of data

- For the most part, every user was expected to more or less know their science, and their code, and what to do with the results.

Research Infrastructure as a Profession

- It has more or less dawned on the academic community that computational science and data science (and perhaps cognitive computing) are disciplines in their own right.
  - Organizing fundamental principles
  - Underlying mathematics
  - Algorithms.
- Scientific/Research/Technical computing infrastructure is now as a profession that reflects the embodiment of these disciplines, and other things.
  - Much Larger Capture of the Scientific Workflow

Research Computing as a Profession

- 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).

- What people expect from Research Computing now (or should), is not just a facility, but a partner in the computational aspects of the scientific workflow.
  - In the face of a bewildering landscape of systems, storage, applications techniques, programming languages, etc.

Business Models for Modern HPC Centers

The great quandary of running an HPC Shop

- You can’t charge for your services.
- You must charge for your services.

Do BOTH those things at the same time, and your success is assured.

The TACC model

- Probably not widely replicable.

  - Much that is specific to academic
  - Much that works only if you are willing to compete as one of a few ”national” centers, which carries significant initial investment.

- So, we will talk about that, but also about other models that I’ve had some success with.

The TACC Model

- Who you really are, and what your priorities really are, can pretty much be summarized by how you spend your money.
- Averaging over a few years:
  - TACC spends ~$30M/year
  - Payroll + Fringe == ~$15M/year
  - Average capital expenditure == ~$10-12M/year (far from linear).
- So, we have to bring in $30M a year. . .
  - UT-Austin Base funding: ~8%
  - IDC Return: ~3%
  - Philanthropy: ~2%
  - Industry sponsors+ projects: ~3%
  - Service center revenue: ~3%
  - Contract Project funding : ~80%

The TACC Model (cont.)

- Most of our project funding is federal
  - Exception: we receive about $3M/year from ”other state sources”, e.g. UT System funding to support the medical centers, etc.
- Project funding was formerly almost entirely “HPC”
  - Track 2 (Ranger, Stampede, Stampede2)
  - XSEDE (HPC shared services)
  - Other NSF systems, e.g. Wrangler
- Growth funding is in what I would call “Comprehensive Cyberinfrastructure”
  - iPlant/Cyverse
  - DesignSafe
  - DARPA Synthetic Design and Discovery
  - Neuronex, etc.

The TACC Model (cont.)

- We have more or less approached saturation on NSF ”HPC” money.
- So, we grow through diversifying, which takes a few forms:
  - Widen our range of services; capture more of the scientific workflow.
    - Web gateways, data management and curations, software, etc.
  - Approach other agencies
    - More mission oriented; broader appeal helps.
    - E.g., NIH wants site that perform genomic pipelines, not hardware.
    - DARPA wants ML platforms, etc.
  - Attack the problem from the other end.
    - Partner with PI’s going after the largest grants on campus – provide the support they need.
    - You may have to change to do this.

Not a National Center?

No problem, there is still a great case to make.

- Most research is computational today, one way or another.
- Find your top researchers, work with them (top means the ones who actually produce funding).
- Find the units they work in, approach their bosses.
  - This perhaps is more administrative/political work than you want.
- At ASU, my center was 50% internally funded, 50% externally/cost recovery.
  - Base budget came from 3 Deans, CIO, and VPR in five equal shares.
  - I had 5 bosses, and they were another group to “manage”.

Cost Recovery

- Cost Recovery is the greatest tool you have to make sure your services are valued,
  - …And the easiest way to screw yourself over.
- IT is typically viewed as a *cost*. Costs are to be minimized (increasingly at universities, CIOs report to CFOs – check your org chart. CFOs have basically one job).
  - Don’t be a cost. Be an investment, or even a revenue source.
- Full cost recovery will get you killed. Don’t do it.
  - Argue that subsidized services promote other efficiencies.
    - People won’t waste time trying to buy and manage their own stuff (instead of doing research).
    - They won’t cripple buildings all over campus trying to install it.
    - Non-experts going to external providers creates security and data risks.
    - Aren’t they more productive with a little help?
  - At the same time, ”free” is not always the best model to make people value you.

A Few Business Model Recommendations

- Unlimited and Free is a bad idea.
  - Try and think of yourself as a drug dealer… you want to provide enough free to get them addicted, then raise the price.
  - In other words, startups are free, large users should have to either pay, or at least justify the investment.
  - If you generate *some* cost recovery, you are leveraging the dollars they give you.
- (For Academics) Find some external funding
  - There are lots of places that need CI help, lots of grant programs to do something innovative, or to help places getting started
    - MRI, EPSCOR, CC*NIE, etc. at the federal level.
  - Units that win some grants are fundamentally treated differently than service units. And also, that helps with. . .

A Few Business Model Recommendations (2)

- Your staff expertise is your primary asset. The machine is just nice for show (whether you are academic or not).
  - You want to be treated as a research *partner* to those on campus, not as a support service. Research computing is hard, and a discipline unto itself. Often, your unit is a better partner than the CS department.
  - Hardware money is somewhat easy to get, win or attract donors for. It’s (sort of) non-recurring, and flashy.
  - Spend your base budget on staff – staff money is hard. This is what you want your senior management to fund; tell them if they give you that, you can come up with the hardware resources.
  - In some cases, you don’t even have to *have* hardware resources – you can outsource this and survive, if your *staff* are valuable to your *users*.
    - Amazon, Google, or TACC can never put expert help down the hall from your people.

Thanks!
Discussion?

dan@tacc.utexas.edu