HPC Downtime Budgets: Moving SRE Practice to the Rest of the World

SREcon Europe 2016

Cory Lueninghoener

July 12, 2016

NNSA National Nuclear Security Administration
Operated by Los Alamos National Security, LLC for the U.S. Department of Energy's NNSA

whoami

- Cory Lueninghoener
- HPC Design Group Leader, Los Alamos National Laboratory
- System administrator, config management junkie, scalable system builder
- Co-chair of LISA 2015
- @cluening | linkedin.com/in/cluening | github.com/cluening | cluening@lanl.gov

Where I Come From

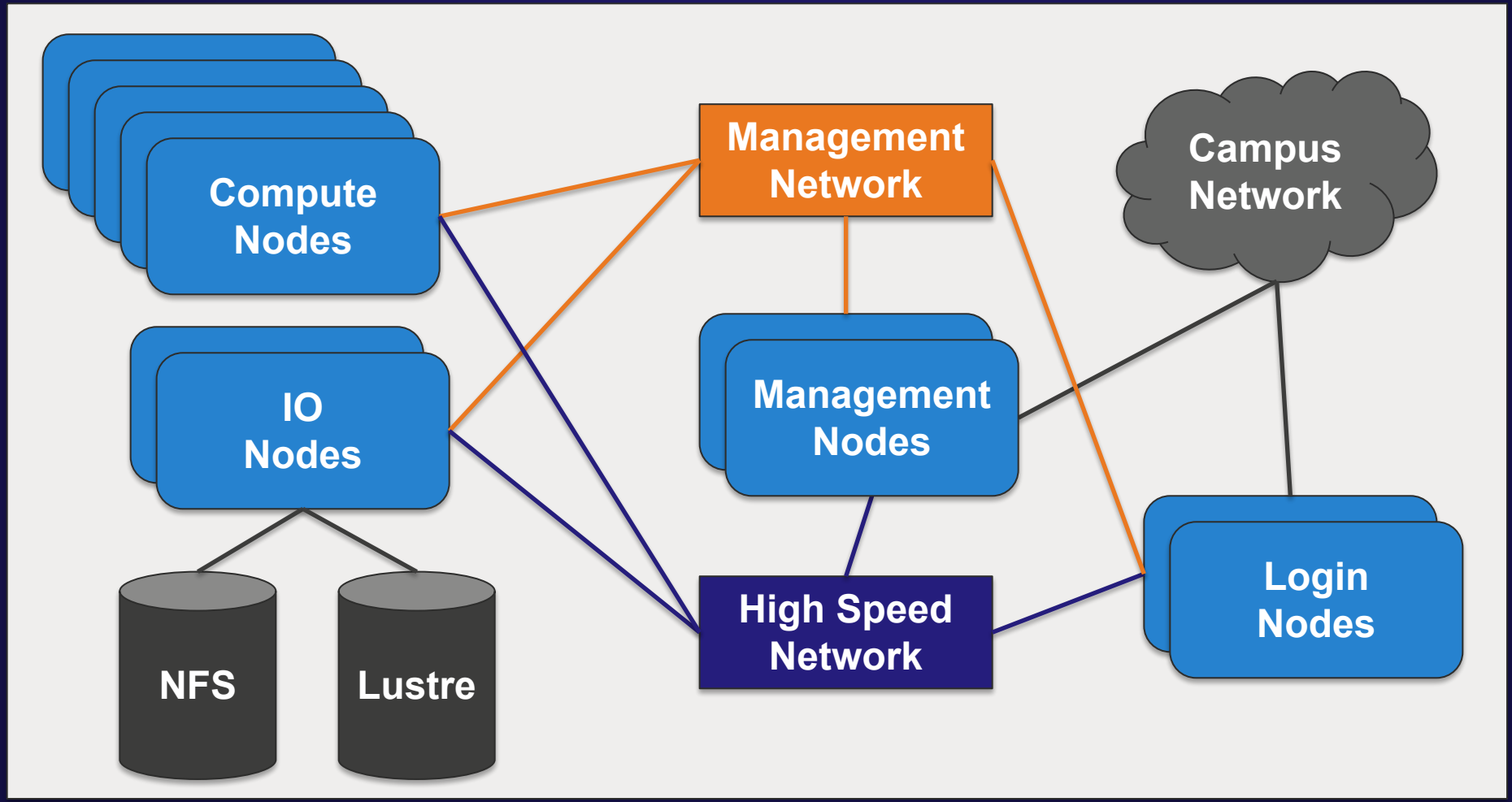
High Performance Computing

HPC Systems

HPC Applications

```
//
int init_pre(MarFS_XattrPre*        pre,
             MarFS_ObjType          obj_type, /* see NOTE */
             const MarFS_Namespace* ns,
             const MarFS_Repo*      repo,
             const struct stat*     st) {

   time_t now = time(NULL);     /* for obj_ctime */
   if (now == (time_t)-1)
      return errno;

   // --- initialize fields in info.pre
   pre->repo         = repo;
   pre->ns           = ns;

   // captures the version of the software, not what's in the config-fi
   pre->config_vers_maj = MARFS_CONFIG_MAJOR; // marfs_config->version_
   pre->config_vers_min = MARFS_CONFIG_MINOR; // marfs_config->version_

   pre->obj_type     = obj_type;

   pre->compression  = repo->comp_type;
   pre->correction   = repo->correct_type;
   pre->encryption   = repo->enc_type;
```

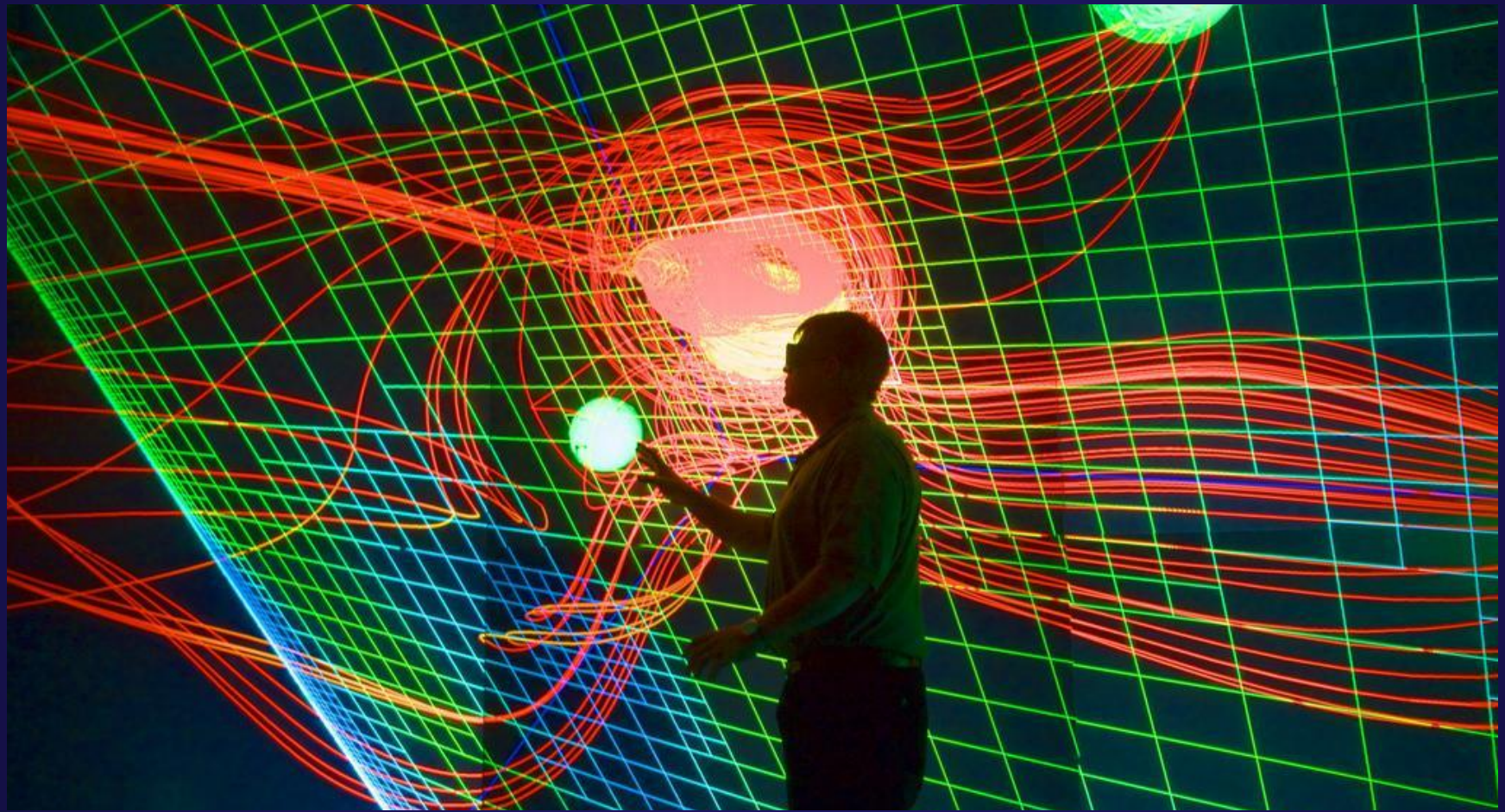
A Typical HPC Cluster Design



At the Lab, We Manage…

- ~20 clusters - ranging from 100s to 10,000s of nodes
- ~36,000 compute nodes
- ~1.2 million compute cores
- ~110 PB of Lustre across 10 filesystems
- ~5,000 square meters of computer room floor

A Typical HPC Application Design



Our Systems Support…

- Job sizes ranging between 64 cores to 64,000 cores
- Job sizes ranging between 4 and 4,000 nodes
- Job times ranging between 1 and 24 hours
- ~3,000 users
- ~6,000 jobs in a typical day

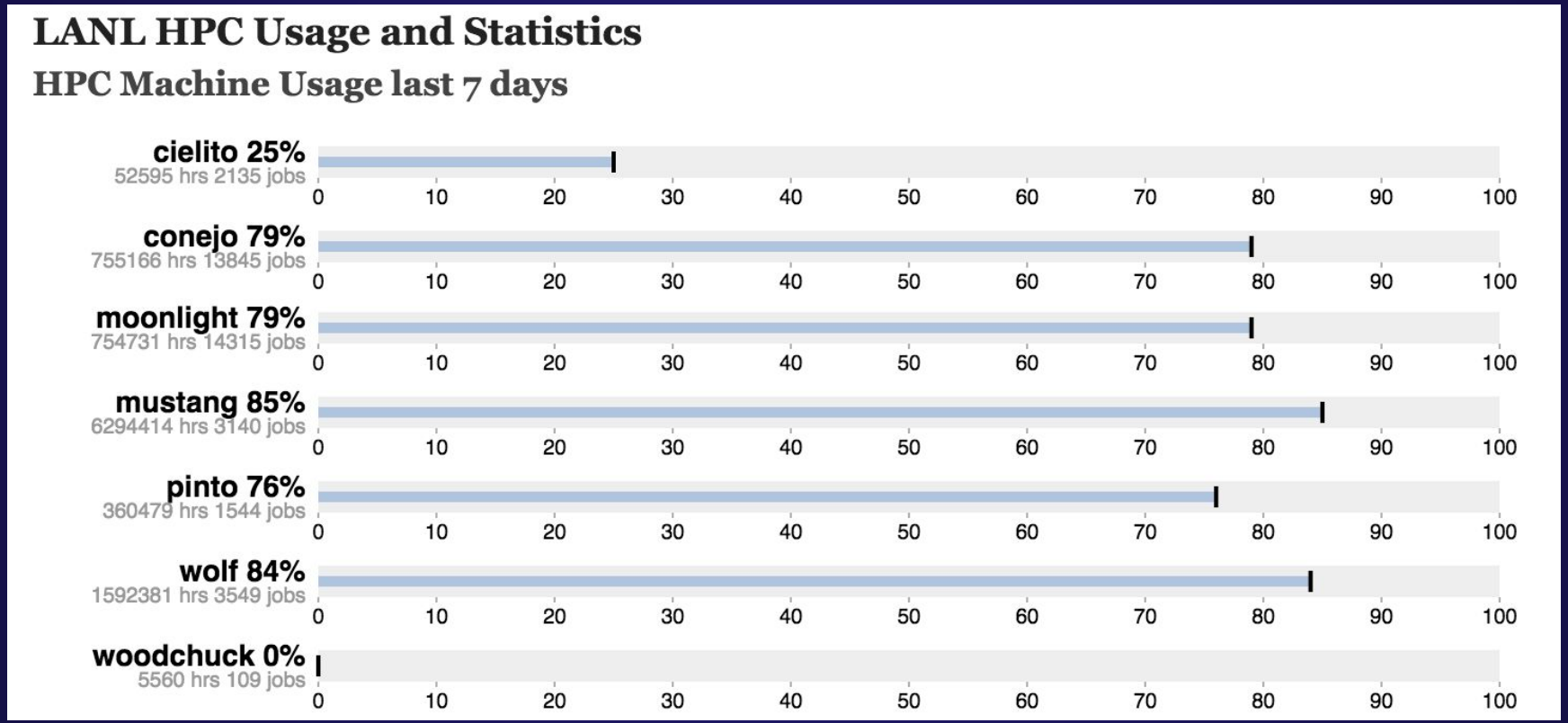
Comparing HPC to Web Engineering

- **Notable differences**
  - Instead of lots of small, redundant, resilient, or independent jobs, a smaller number of really large tightly coupled jobs
  - Instead of millions of transactions per unit time, tens or hundreds
  - Instead of millions of customers, a smaller set of domain experts
- **But a lot of similarities**
  - Very large scale systems
  - Very small scale operations/admin/SRE/whatever teams
  - Lots of monitoring and metrics gathering to do

Metrics We Gather

- **Typical systems monitoring stuff**
  - Node health – temperature, load, error log, …
  - Outages – both planned and unplanned
- **HPC-specific stuff**
  - High speed network health
  - Parallel filesystem health – speed, free space, …
  - Job queue depth – number of jobs in queue, size, owner, length of time
  - Utilization – job start time, end time, size, owner

Utilization Is An Important Metric



The Problem

Monthly Cluster Maintenance

Calendar for Jun 2016

<<Previous Month * denotes an entry is referred to by another ticket(s) Next Month>>

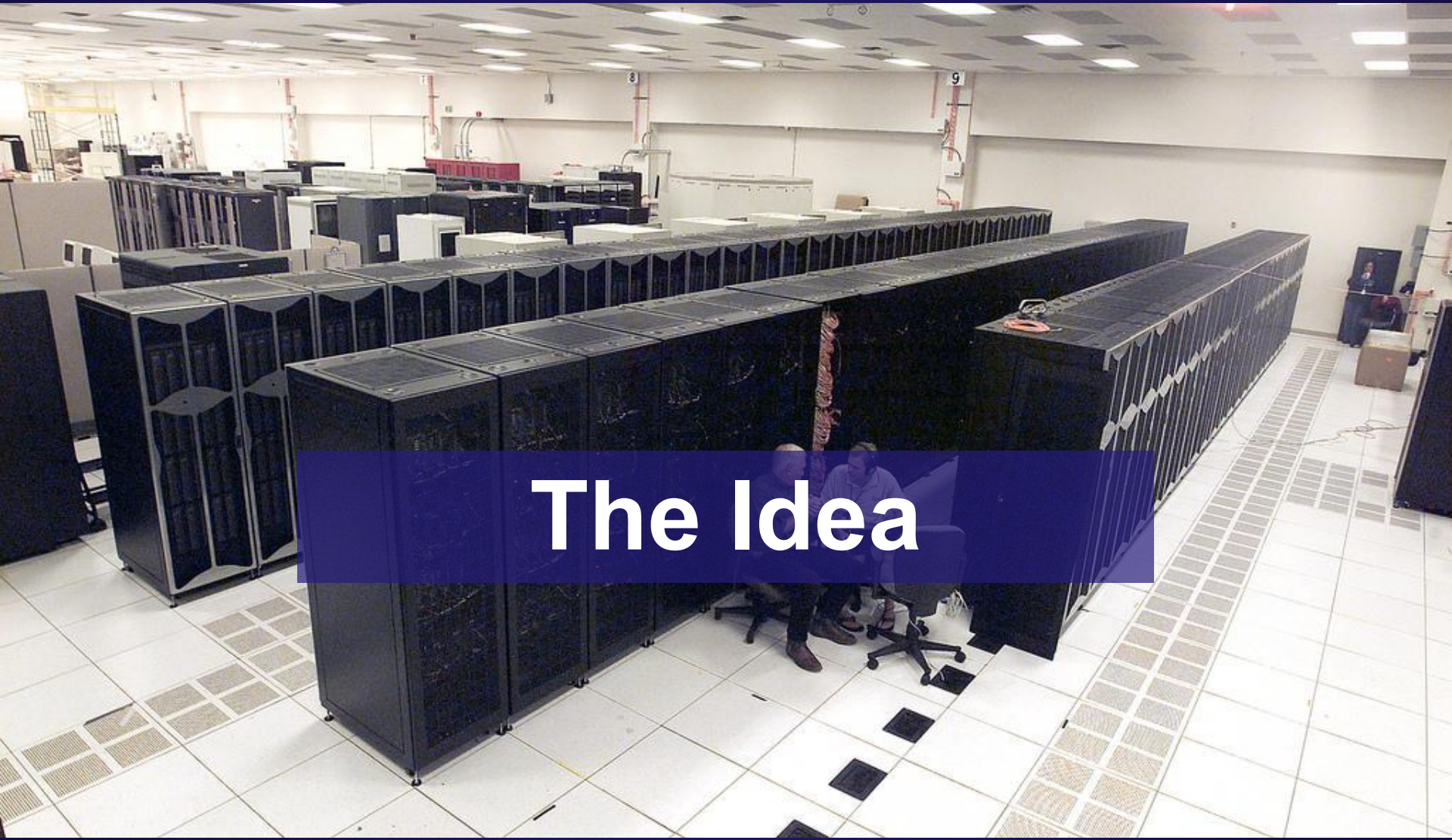
Sun	Mon	Tue	Wed	Thu	Fri	Sat
29	30 Platforms Change Time Memorial Day Holiday	31 Turquoise DTN DST * Trinity PM/DST	1 Conejo PM/DST * Cielito PM/DST * Trinitite PM/DST Hobo Decommissioned	2 Programming Env. Maintenance PM/DST *	3	4
5	6 Platforms Change Time	7 Wolf PM/DST * Lightshow PM/DST * CFTA PM/DST Turquoise DTN DST Open User Gateways DST * TFTA PM/DST *CANCELED* * Trinity PM/DST	8 Trinitite PM/DST Cielo PM/DST * Hobo PM/DST *CANCELED* Mustang PM/DST * Woodchuck PM/DST * Cortez DST – 5 to 8 pm	9 Open Support Services DST * Open Monitoring Services DST * Open License Services DST *	10	11
12	13 Platforms Change Time	14 Turquoise DTN DST Pinto PM/DST * GPFS PM/DST Trinity PM/DST * Redcap PM/DST *	15 Trinitite PM/DST Viewmaster II PM/DST	16 Programming Env. Maintenance PM/DST * Secure Support Services DST *	17	18
19	20 Platforms Change Time	21 Turquoise DTN DST Moonlight PM/DST * Trinity PM/DST	22 Trinitite PM/DST Cielo PM/DST Luna PM/DST * Network PM * Cortez DST – 5 to 8 pm	23 Secure License Services DST *	24	25
26	27	28 RFTA PM/DST * Secure User Gateways DST *	29	30	1	2

Monthly Cluster Maintenance

- **Maintenance often affects an entire cluster at a time**
  - This is a design tradeoff
  - Individual node work: no problem
  - OS image, high speed network, filesystems: safest with idle cluster
- **User impact mitigation**
  - Schedule similar clusters separately
  - Do pre-work when possible
  - Only idle cluster when needed
  - Leave login nodes up as much as possible

Minimize Impact on Users

- **Balance: system work vs. user work**
- **Maintenance impact on users**
  - How do we measure it?
  - How do we use that info to make stronger decisions?
- **Error budget concept provides some inspiration**



The Idea

SRE Error Budgets

- **Hack your SLA to your advantage**
- **The basic concept:**
  - SLA says 99.99% uptime? That's actually .01% downtime
  - Use every bit of that time to innovate or fix
- **Exact implementation doesn't quite fit**
  - .01% of 10,000,000 web requests is 1,000 requests
  - .01% of 1,000 HPC jobs is 1/10 of a job

But the idea of tracking downtimes and using them to your advantage does fit

HPC Downtime Budgets

- Each cluster has one 10-hour Dedicated System Time (DST) scheduled each month
- In a quarter, 3 DSTs x 10 hours = 30 hours of scheduled downtime
- So, the budget per quarter per cluster is 30 hours
- Or, about 1.4% of the cluster's available time

This is something we can track and use to make decisions

Goals

- Track maintenance downtime
- Make downtime risk analysis easier
- Inspire team members to play the game
- Report up the chain how we are doing

Not Goals

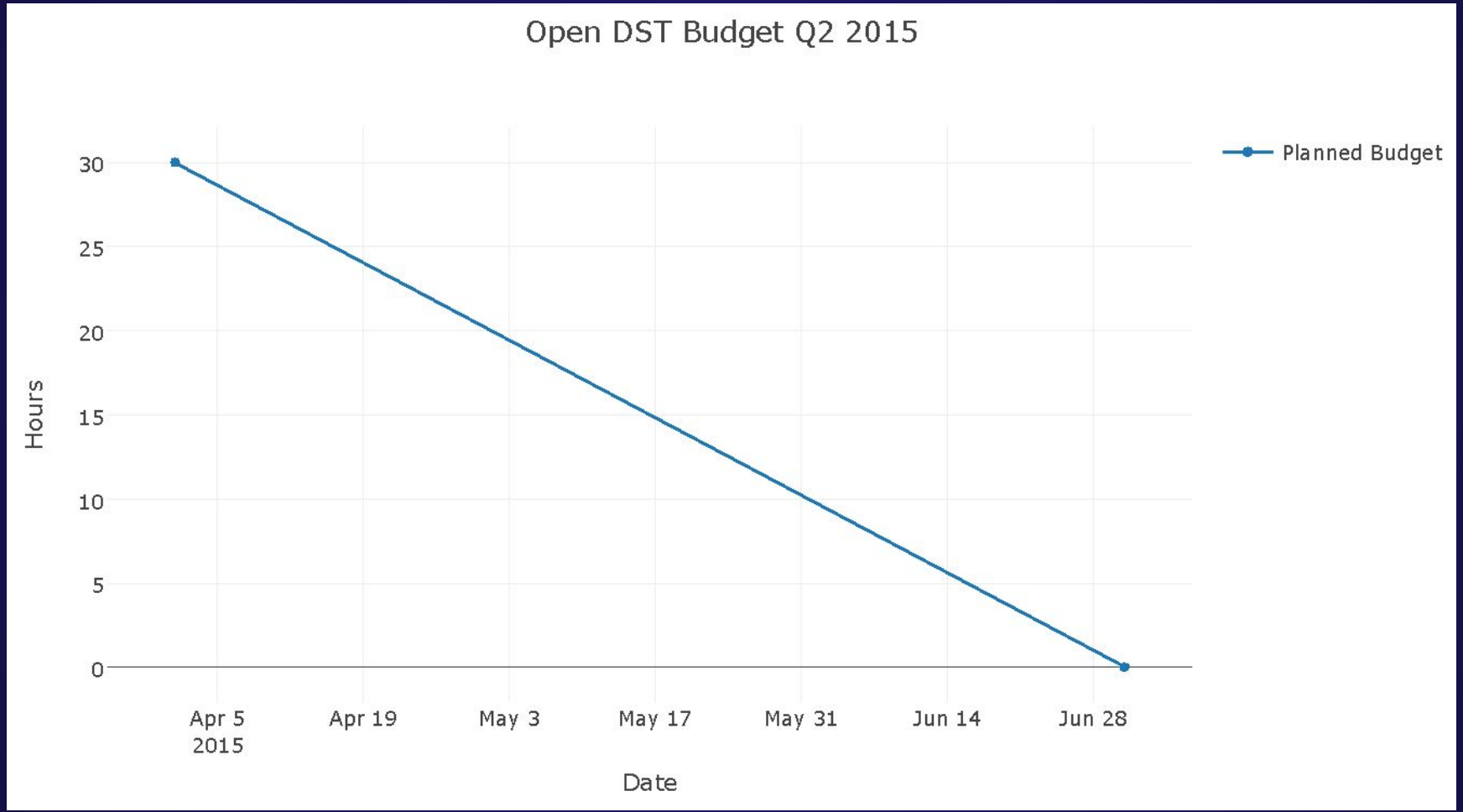
- Make maintenance days more stressful
- Get into a minute-pinching mode
- Figure out how to beat the system
- Become more worried about the metric than the outcome

Initial Implementation

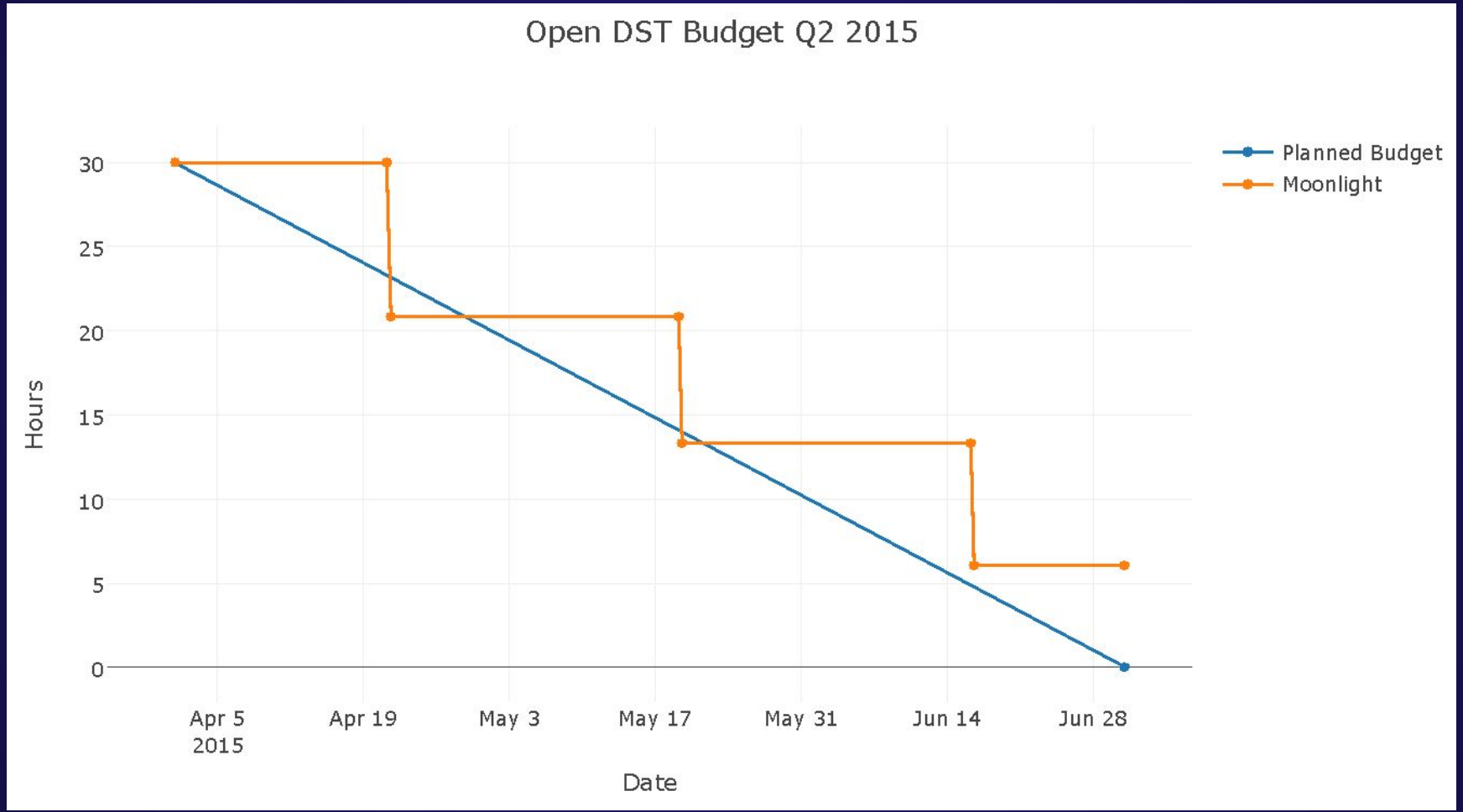
- Python script
- … that connects to a mysql stats database
- … to generate json output
- … that is graphed using plotly

Status: still proof of concept

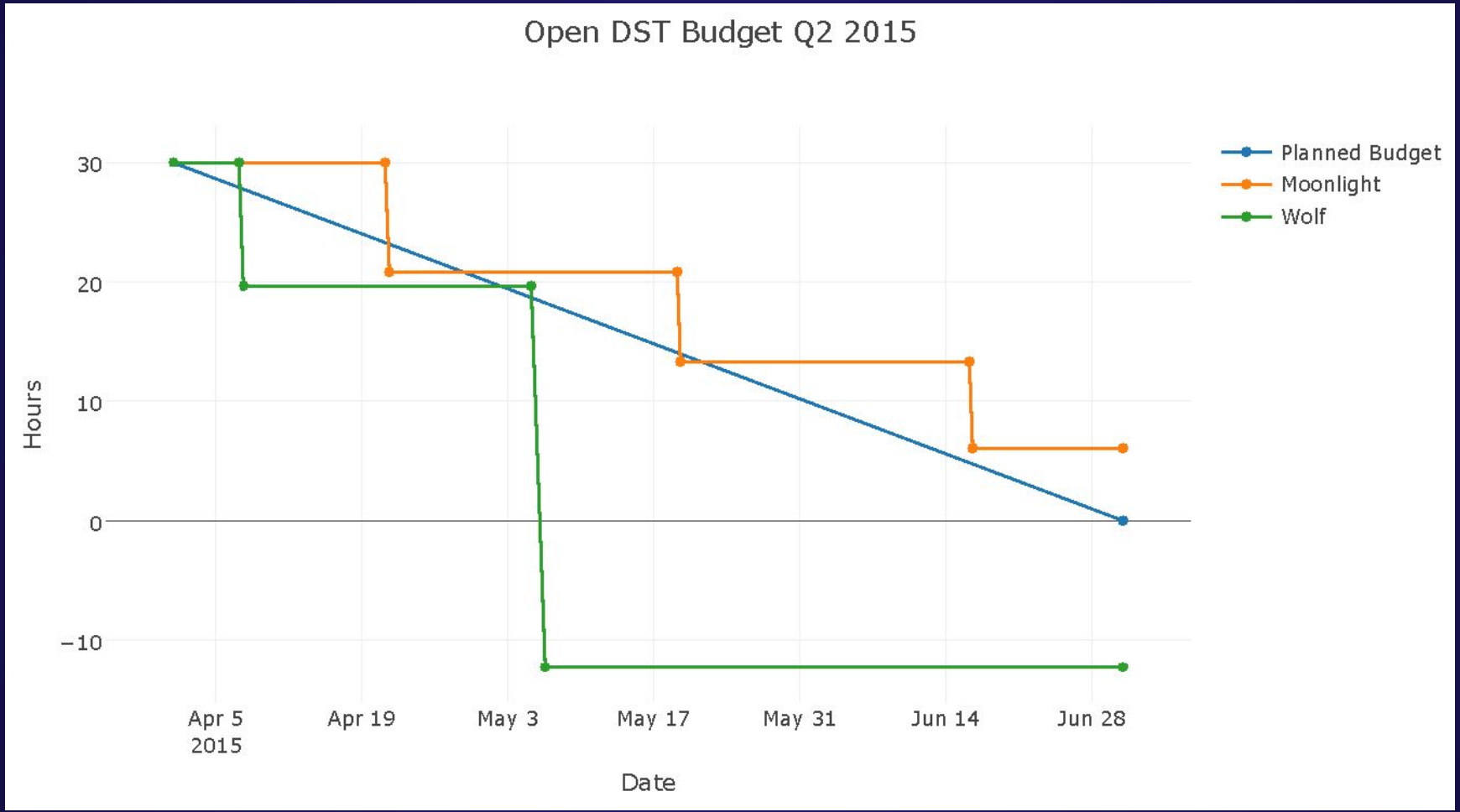
Initial Implementation



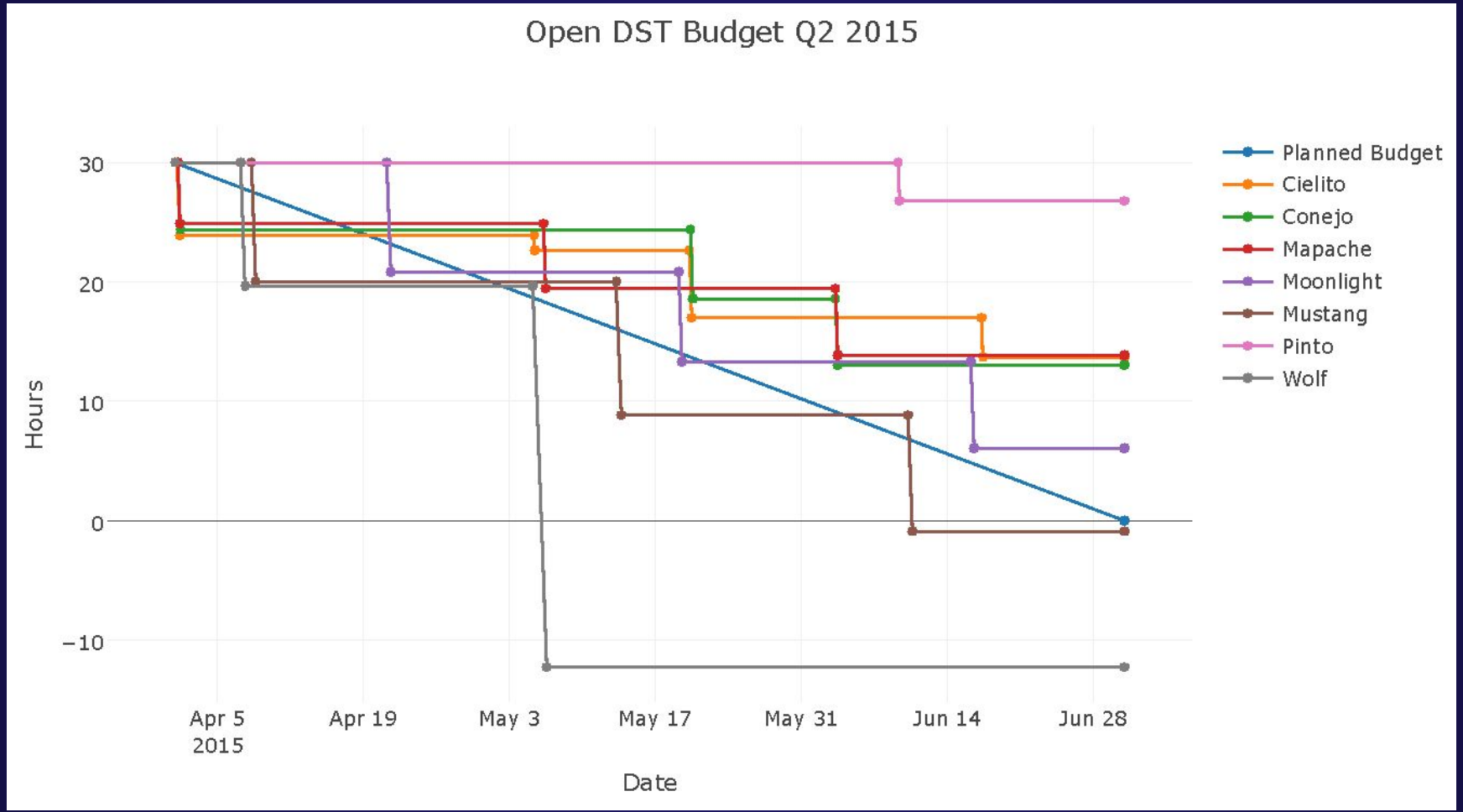
Initial Implementation



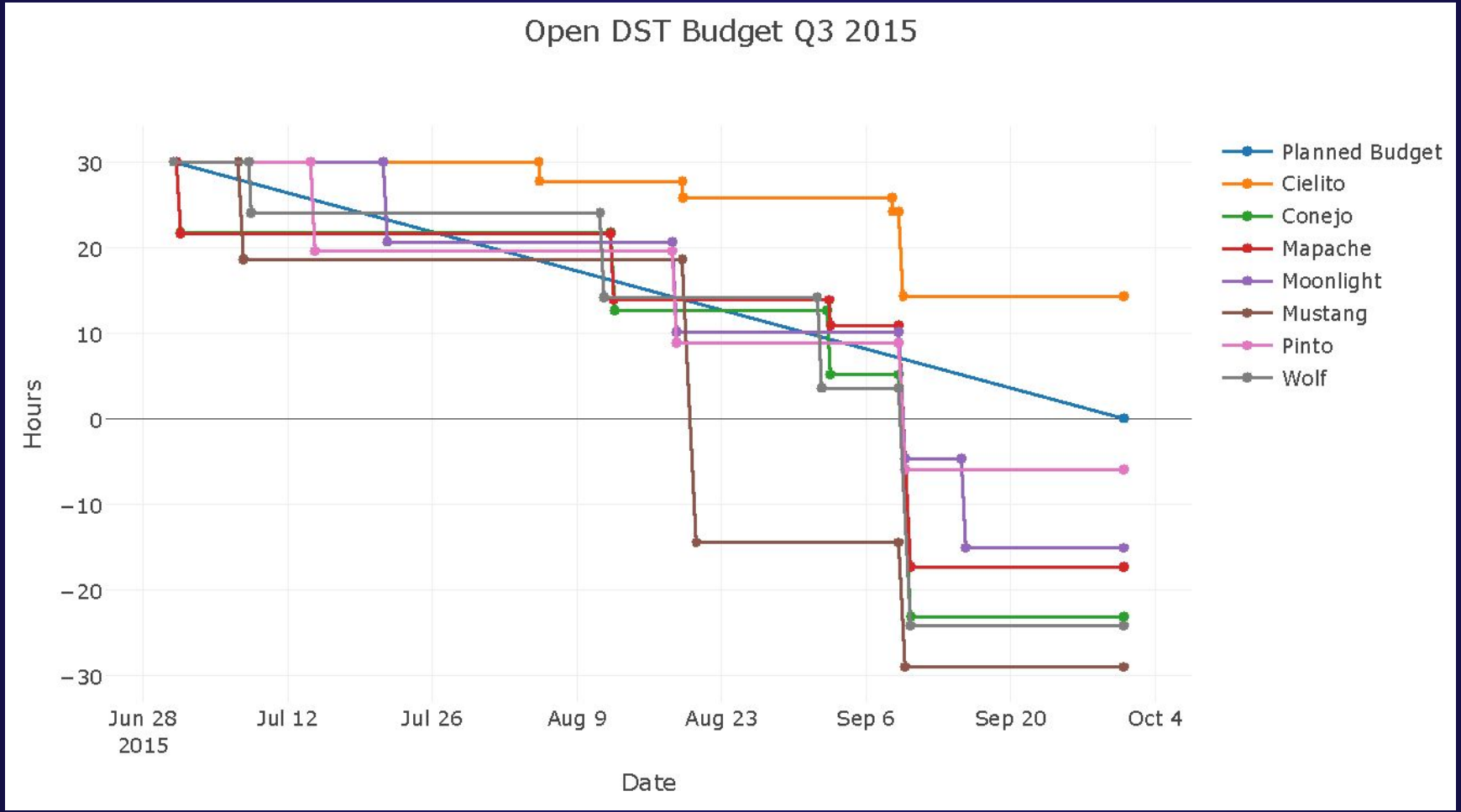
Initial Implementation

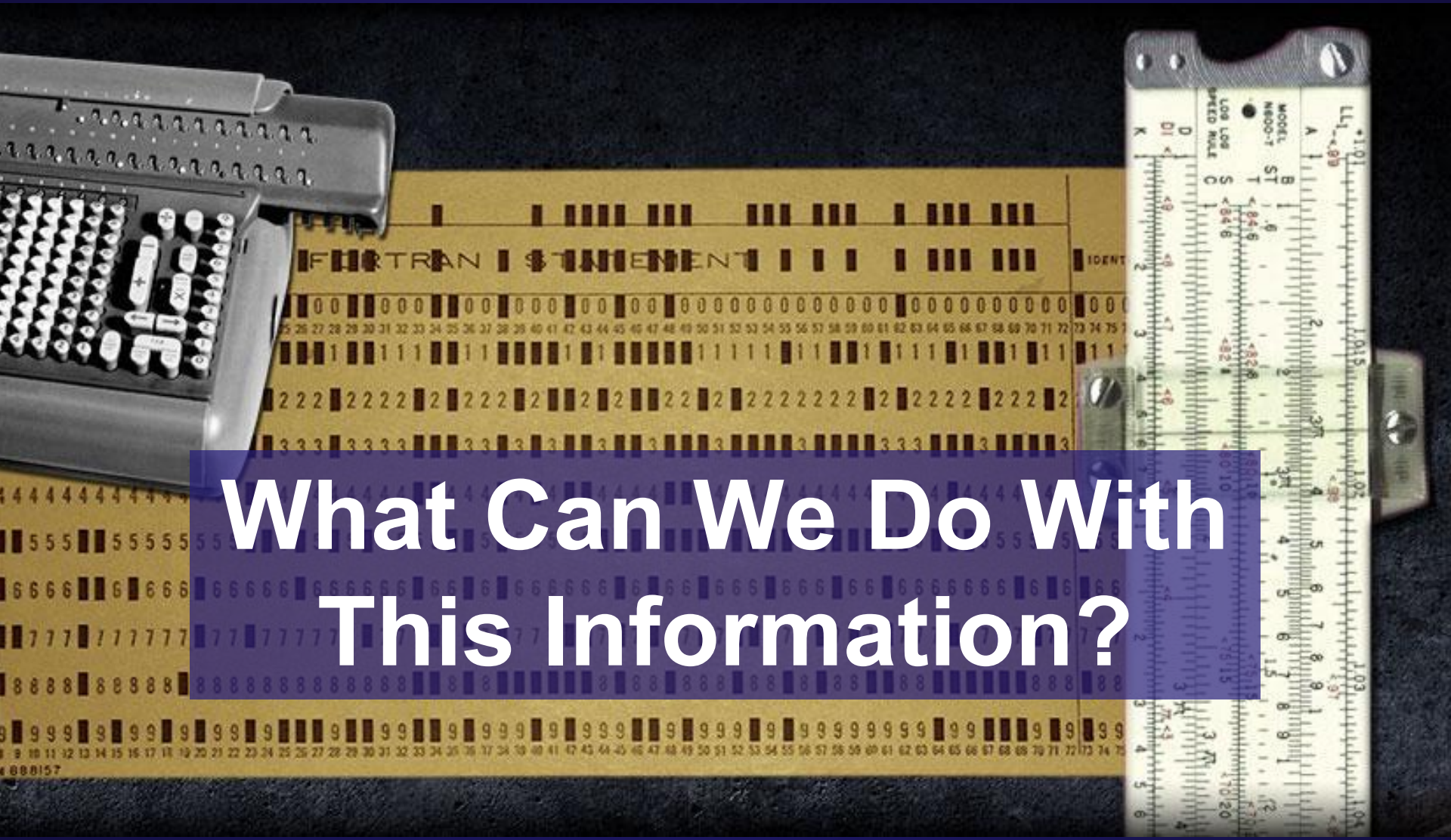


Initial Implementation



Initial Implementation

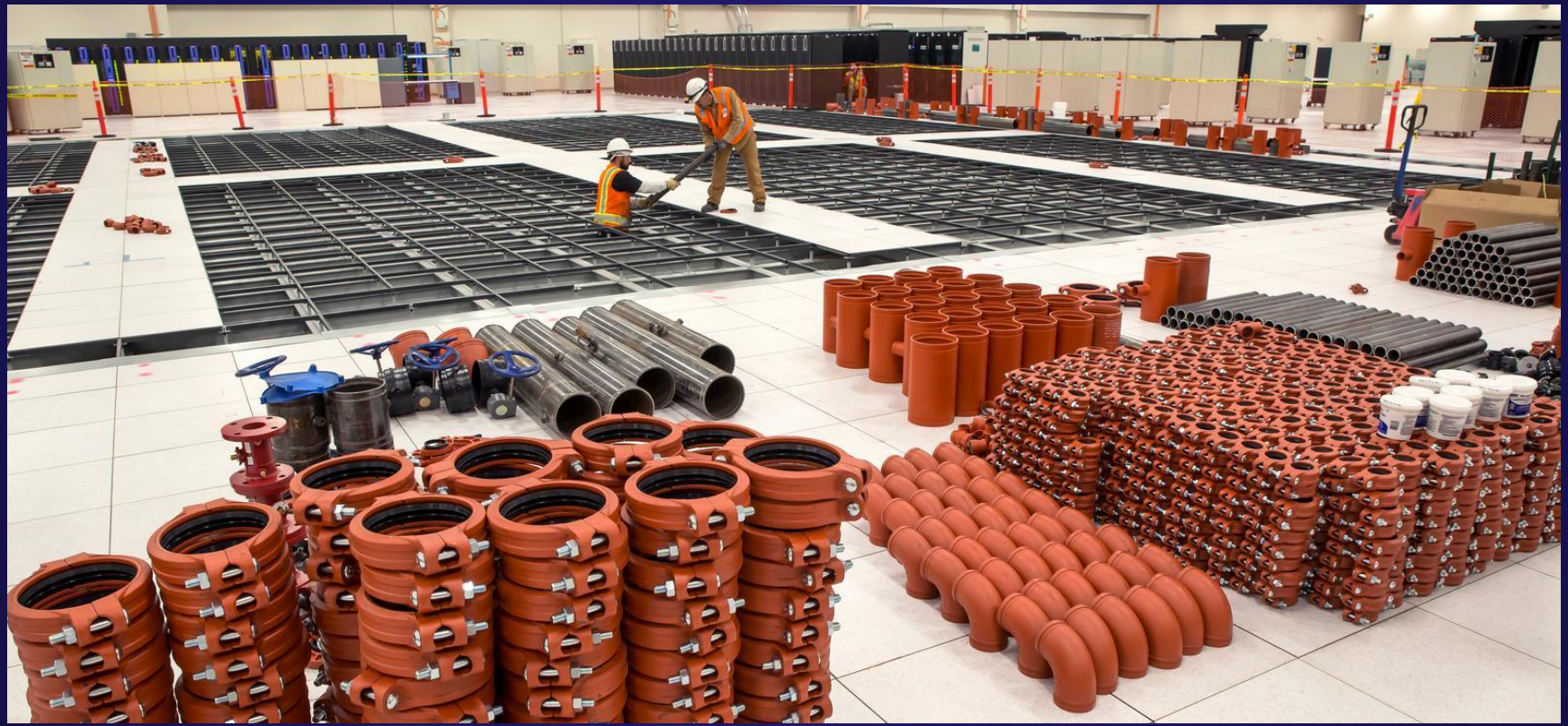




What Can We Do With This Information?

Downtime Budgeting and Reporting

Facilities Work Budgeting



Lightning Strike Budgeting

Dedicated User Time Requests

Admin Time to Fix Technical Debt

The Challenge

- **How can this talk be generalized?**
- **Many SRE ideals easily map to traditional environments**
  - Monitoring and metrics
  - Incident planning and analysis
  - Automation and configuration management
- **Many don’t**
  - Scalability
  - SRE hiring
  - Continuous deployment

The Challenge

- What makes SRE environments unique?
- How can I teach somebody in a different environment to use what I have learned?

The Challenge

- This is not a conversion or evangelization challenge
- This is a community-building challenge

Cory Lueninghoener
cluening@lanl.gov
@cluening