Argonne NATIONAL LABORATORY

# Cognitive Dissonance in HPC

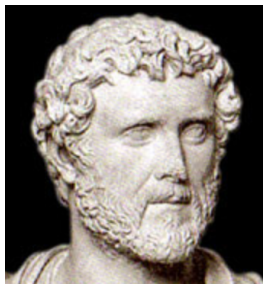
**Pete Beckman**

Argonne National Laboratory

Northwestern University

U.S. DEPARTMENT OF ENERGY

# Aesop's Fables: The Fox and the Grapes



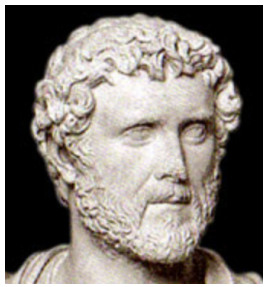


- Fox wants the grapes
- Fox can't have the grapes

Resolve…

- I don't really want them (sour grapes)
- Understand what you should want, seek it!

# Aesop's Fables: The Fox and the Grapes



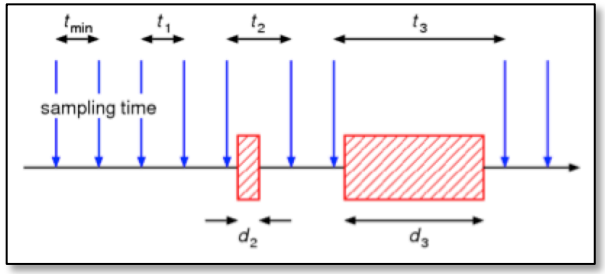


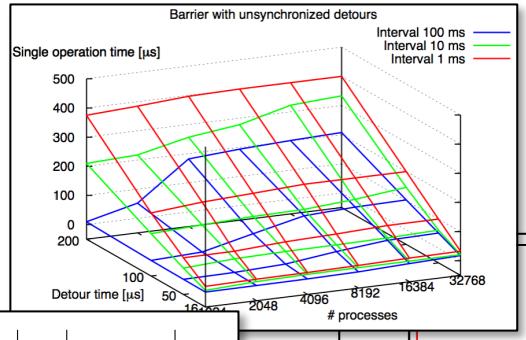
## Cognitive Dissonance

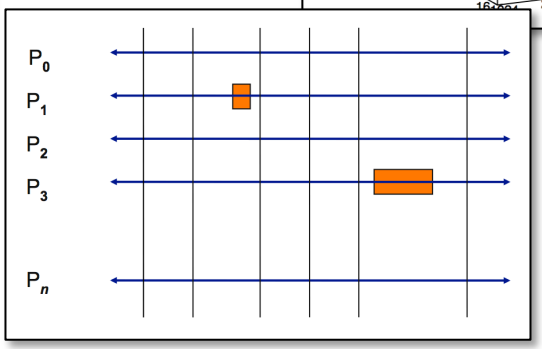
[…] the mental stress or discomfort experienced by an individual who holds two or more contradictory beliefs, ideas, or values at the same time, or is confronted by new information that conflicts with existing beliefs, ideas, or values. (wikipedia)

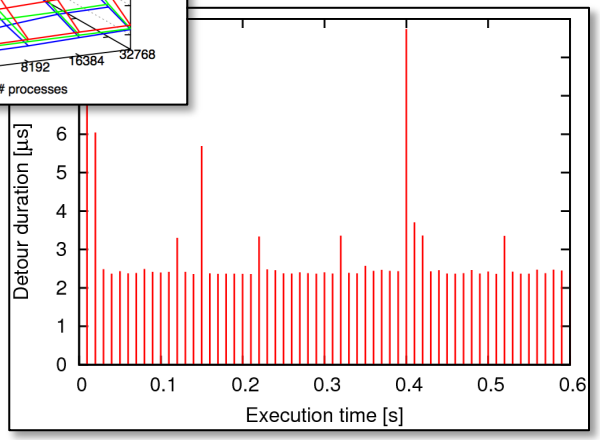
# The Fox Wants a Low-Noise (Jitter Free) OS

- **Research Challenges for Exascale**
  - "[…] Very low noise kernels and operating systems need to be developed"
- **Procurement**
  - "[…] with minimal necessary functionality, extremely low noise that runs on a subset of CN processors"
- etc. etc. etc.



# What Prevents Scalability?

- **Insufficient parallelism**
- **Insufficient latency hiding**
- **Insufficient resources** (Memory, BW, Flops)

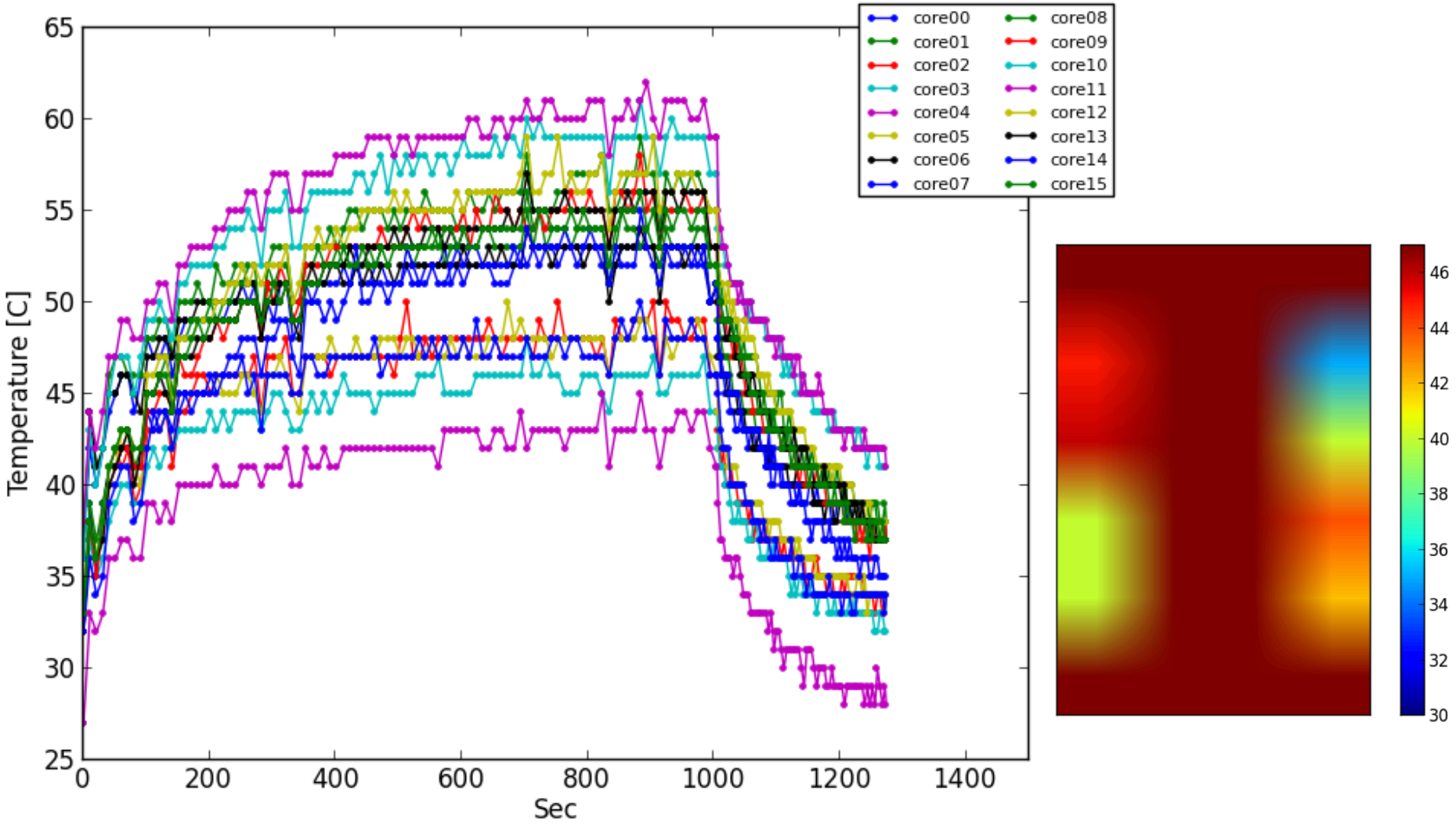
# What Prevents Scalability?

- **Insufficient parallelism**
  - As the problem scales, more parallelism must be found
- **Insufficient latency hiding**
  - As the problem scales, more latency must be hidden
- **Insufficient resources** (Memory, BW, Flops)
  - As the problem scales, so must the resources needed

# What Prevents Scalability?

- **Insufficient parallelism**
  - As the problem scales, more parallelism must be found
- **Insufficient latency hiding**
  - As the problem scales, more latency must be hidden
  - *The Fox also (correctly) wants dynamic behavior:*
    - *Power management, resilience, PGAS, data-driven execution, multi-resolution, …*
- **Insufficient resources** (Memory, BW, Flops)
  - As the problem scales, so must the resources needed

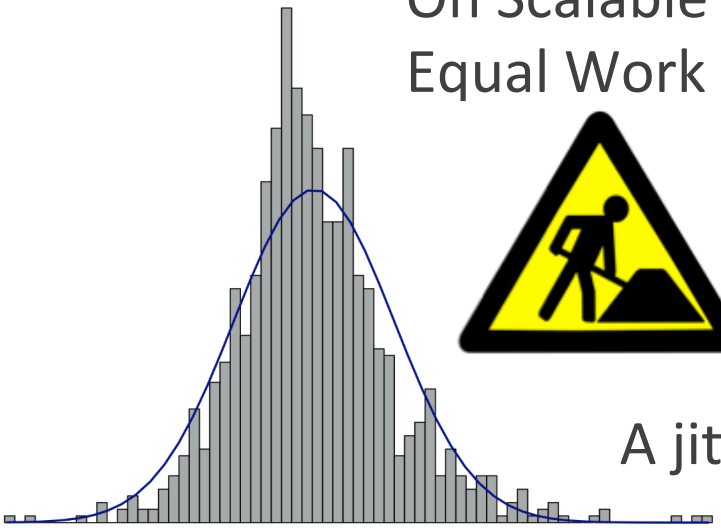
# Following the Power…



# Fox, You Should Want Dynamic and Adaptive

**Accept the Truth:**

On Scalable Systems
Equal Work is not Equal Time

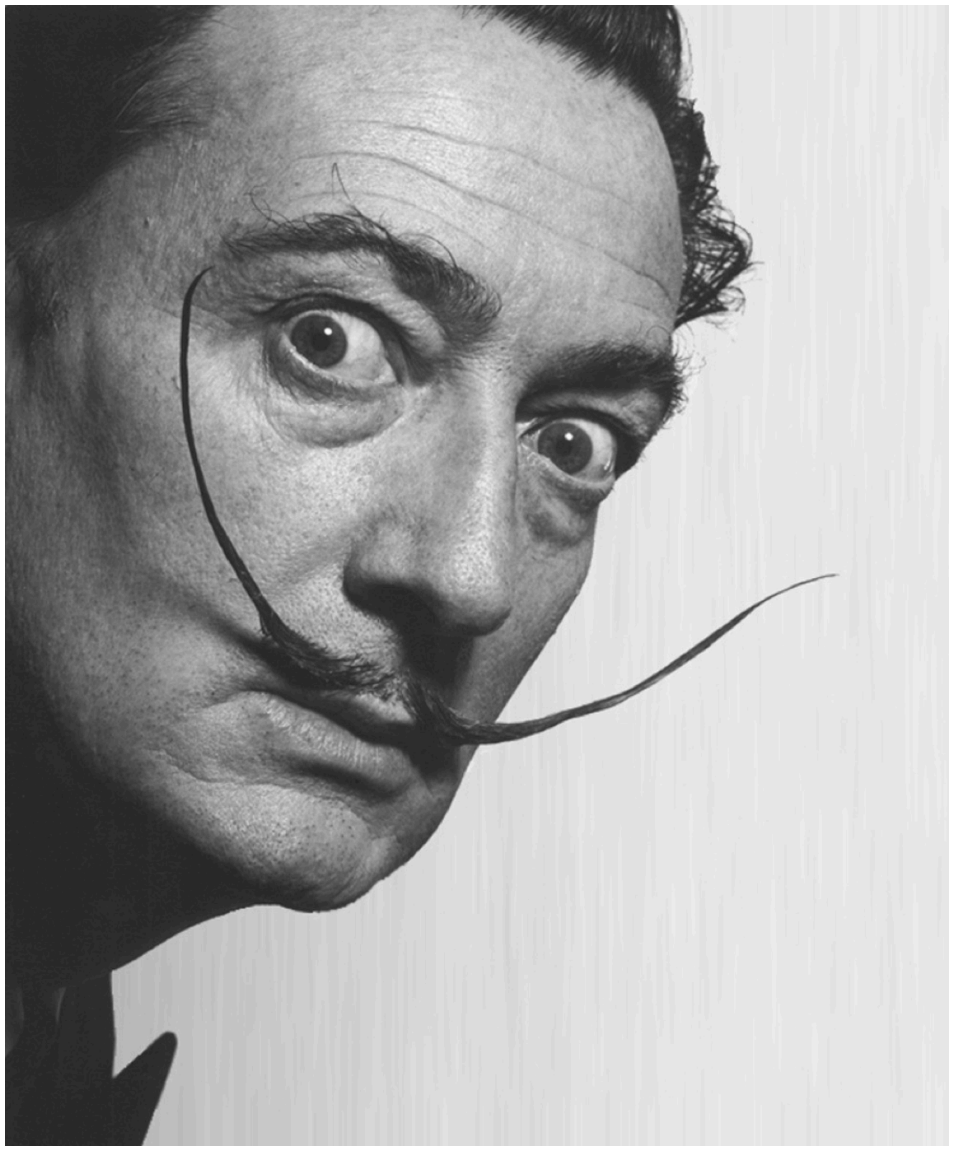




≠


A jitter-free system is physically impossible

- Seek new latency tolerant algorithms and methods.
- Create new tools that measure and predict latency tolerance and execution distribution

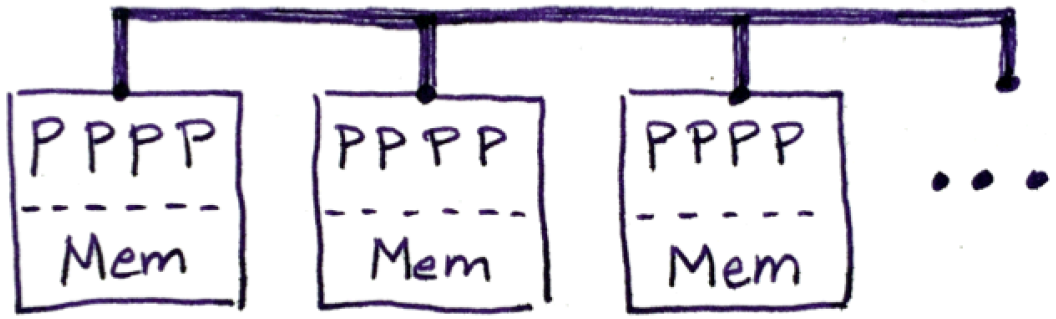


Salvador Dali

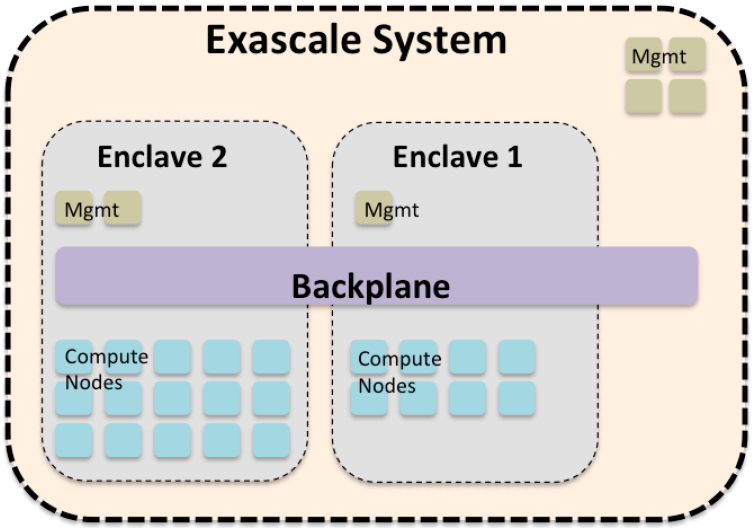
# Protect Me
# From What I Want

# The Fox Wants *Highly Productive* Computing
## (Simple Abstractions That Run Blindingly Fast)

- Example: **Research Challenges for Exascale**
  - "[…] the data-centric model, data lives in persistent memory, with many CPUs surrounding it, and the data moves as little as possible through a shallow/flat storage hierarchy. "
  - Fox wants "Flat or Shallow" memory
  - "Oh for the flat memory of the Cray vector machines"
- The Fox can sometimes be distracted with PGAS



# But First…..




New abstractions & implementations

**ANL**: Pete Beckman, Marc Snir, Pavan Balaji, Rinku Gupta, Kamil Iskra, Franck Cappello, Rajeev Thakur, Kazutomo Yoshii
**LLNL**: Maya Gokhale, Edgar Leon, Barry Rountree, Martin Schulz, Brian Van Essen
**PNNL**: Sriram Krishnamoorthy, Roberto Gioiosa
**UC**: Henry Hoffmann
**UIUC**: Laxmikant Kale, Eric Bohm, Ramprasad Venkataraman
**UO**: Allen Malony, Sameer Shende, Kevin Huck
**UTK**: Jack Dongarra, George Bosilca, Thomas Herault

# Back to the Fox wanting

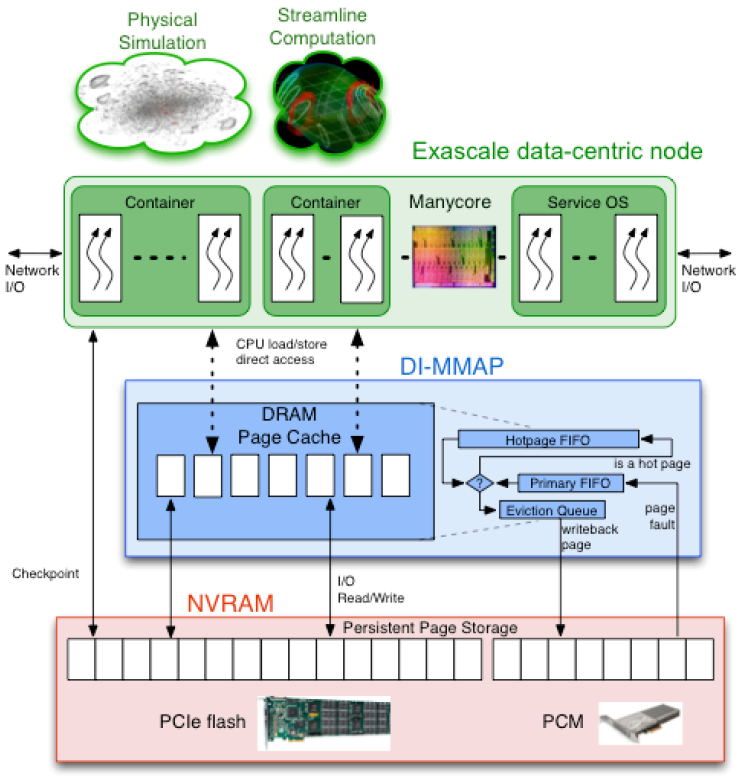
## *Highly Productive* Computing that's Blindingly Fast

- (…wanting flat/shallow memory)
- What we know: many types of storage technologies with difference characteristics
  - 3D in package memory, fast scratchpad, NVRAM
  - Many locations (die, package, bus, interconnect)
- Two options,
  - Hide everything in the runtime, look flat, and succeed occasionally
  - Fix our programming abstractions and provide hierarchy
    - double a[1000];  // Where will this go?  How will it be moved?

# Example: Transparently making NVRAM look like RAM

## Extending the memory hierarchy with Flash NVRAM

- Problem
  - Transparently incorporate NVRAM into memory hierarchy for simulation and analysis applications
- Solution
  - Linux OS module DI-MMAP manages fixed size, statically allocated page buffer of memory mapped file pages
  - Application data structures reside in mapped pages and accessed as if in memory
  - Available at http://bitbucket.org/vanessen/di-mmap
- Recent results
  - Integrated DI-MMAP into Labs' standard TOSS distribution and slurm job manager
  - Improved DI-MMAP performance for "extended memory" mode with optimized write-back
  - Added support for MPI communications from memory-mapped data structures in pinned and locked memory pages
  - Verified DI-MMAP functionality in Linux container
- Impact
  - Applications can exploit NVRAM in container without re-write





Contact:
Brian Van Essen (vanessen1@llnl.gov)
Maya Gokhale (maya@llnl.gov)

# Fox, You Should Want New Memory Abstractions

- To increase performance and reduce power, multiple technologies and hierarchies are needed.

- Our flat memory abstraction is decades old
- Don't be distracted by PGAS
- We need new views of memory and placement
- We need runtime systems to move memory objects/blocks (not pages)
- We need new views on consistency, coherency, and sharing

# The Fox Wants Low Run-Time Variance

- **Procurement:**

"The system shall provide correct and consistent runtimes. An application's runtime (i.e. wall clock time) shall not change by more than 3% from run-to-run in dedicated mode and 5% in production mode."

Why? What is the impact on the domain science?

Real reason: Application tools for performance analysis are poor, so folks simply default to wall clock, and want it to be repeatable

**Fox, you should want helpful performance tools that work well while computer responds dynamically to power loads or fault.**

# The Fox Wants Cloud Computing, Not HPC, Because On-Demand is Better Than a Queue

- There are many valid reasons to like cloud models
  - Software complexity (VMs or Docker can provide relief)

- **Fox, you should want HPC systems to be more flexibly configured:**
  - On-Demand computing should be supported
    - (modify sharing and cost model)
  - Embrace variable run-times
  - Complex strategies for queuing

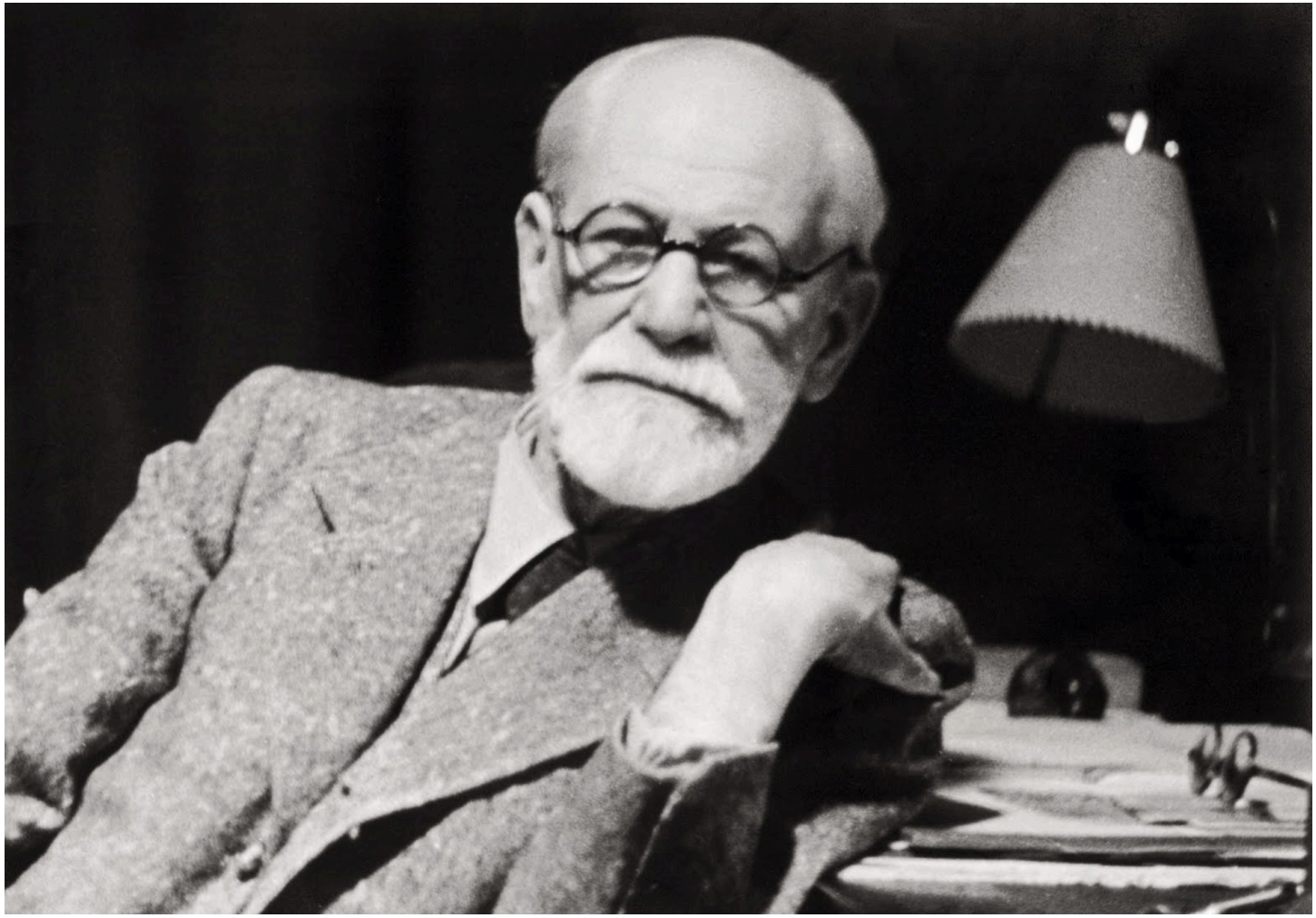
# If Only We Had Time, Foxes Need More Therapy

- The Fox Wants a Revolutionary New Execution Model
- The Fox Wants Reliable Computing Platforms
- The Fox Wants……

# This is Not Hopeless… The Fox Has (Sometimes) Learned

- The Fox wanted to define application success as high percentage of peak flops
  - (or maybe not)… […] "Looking forward, it is likely that some revolutionary technologies will be required, particularly if one considers the low fraction of peak performance achieved by many HPC applications today"
- The Fox wanted to benchmark supercomputers with Linpack
- …

# The Rest of the Week
# Reflect and Seek Consonance



# Questions ?