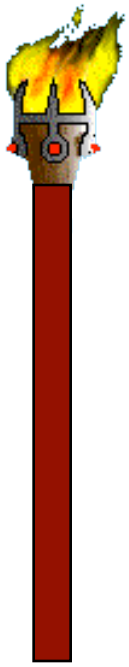
CCGSC 2010 Presents

Igniting Exascale Computing



Bill Gropp
Pete Beckman
Frank Cappello

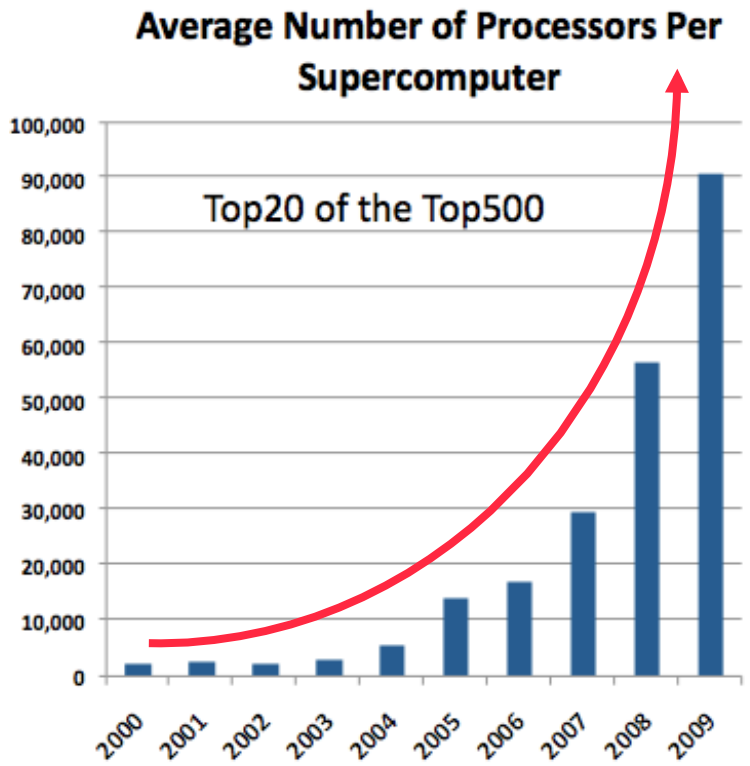
Al Geist
Satoshi Matsuoka
Thomas Sterling

Fear of the Exponential Growth in Parallelism

Sequoia 1.5M cores 2011

- Fundamental assumptions of today's system software architecture did not anticipate exponential growth in parallelism
- Number of system components is increasing faster than component reliability, which is set by COTS needs
- Number of components and MTBF leading to a paradigm shift – **Faults will be the norm rather than rare events. SW adaptation to frequent failures or**

"be crazy and die like a dog"



Fear of the Unknown

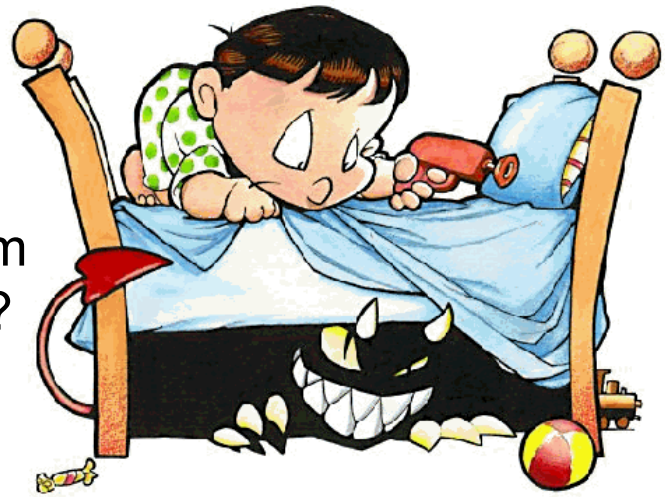
Hard errors – permanent component failure either HW or SW (hung or crash)

Transient errors –a blip or short term failure of either HW or SW

Silent errors – undetected errors either hard or soft, due to lack of detectors for a component or inability to detect (transient effect too short). Real danger is that answer may be incorrect but the user wouldn't know.

Statistically, silent error rates are increasing.
Are they really? Its fear of the unknown

Are silent errors really a problem or just monsters under our bed?



Fear of a Paradigm Shift
"Failure is the Norm"

Factors Driving up the Fault Rate

Number of components both memory and processors will increase by an order of magnitude which will increase hard and soft errors

Smaller circuit sizes, running at **lower voltages** to reduce power consumption, increases the probability of switches flipping spontaneously due to thermal and voltage variations as well as radiation, increasing soft errors

Power management cycling significantly decreases components lifetimes due to The thermal and mechanical stresses

Heterogeneous systems make error detection and recovery even harder, for example, detecting and recovering from an error in a GPU (hundreds of cycles to drain the pipes)

It's Already Here

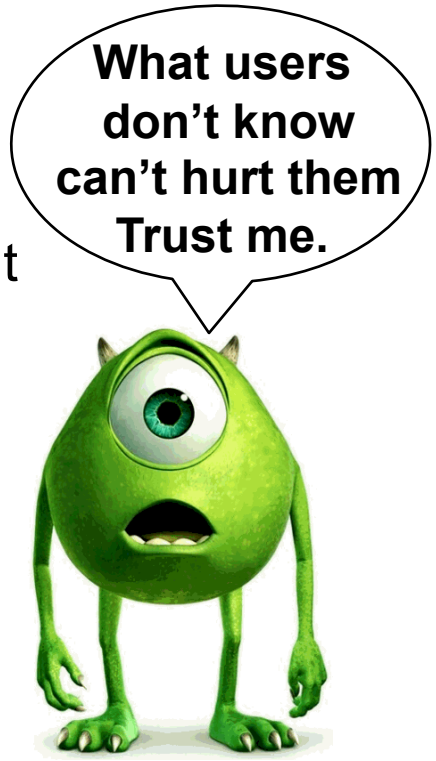
Eg. in 3 days

error msgs

614	inactive link
861	machine check exception
24,215	deadlock timeouts

Yet system stays up for a week at a time

How can we overcome the fear?



Systematic study of the error types and rates in today's Petascale systems, including a study of silent error rates.

What are the 10 **most common failure types** an application may see?

Define a Fault Model API that defines a standard notification protocol and recovery options to SW components

How do we move past qualitative statements to quantitative predictions?

The above studies and definitions are a first step, but Extrapolation of an exponentially growing function…
Good luck with that!