

# Parallel Research at Illinois

Parallel Everywhere

PARALLEL@ILLINOIS

[www.parallel.illinois.edu](http://www.parallel.illinois.edu)



Iliac IV



Cloud Computing Testbed



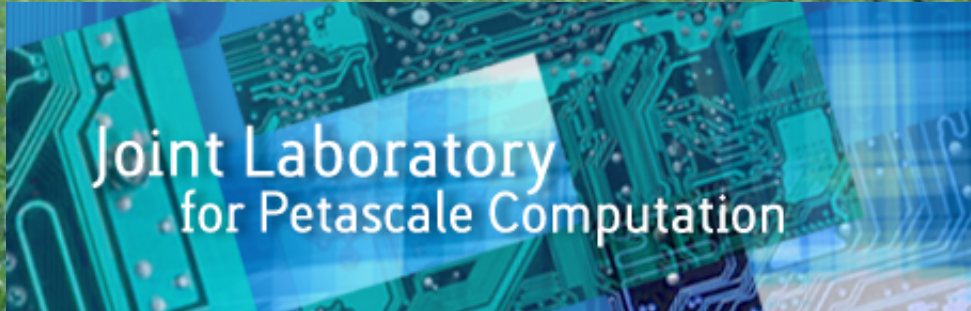
Extreme Scale Computing



OpenSparc Center of Excellence



CUDA Center of Excellence



## Why is Parallel so Cool?

# Moore's Law Pre 2004

- Number of transistors per chip doubles every 18 months
- Performance of single thread increases with every new hardware generation
  - Faster clock cycle
  - More instructions per cycle
- Better performance on existing applications, and new applications enabled with each new hardware generation

# Moore's Law Post 2004

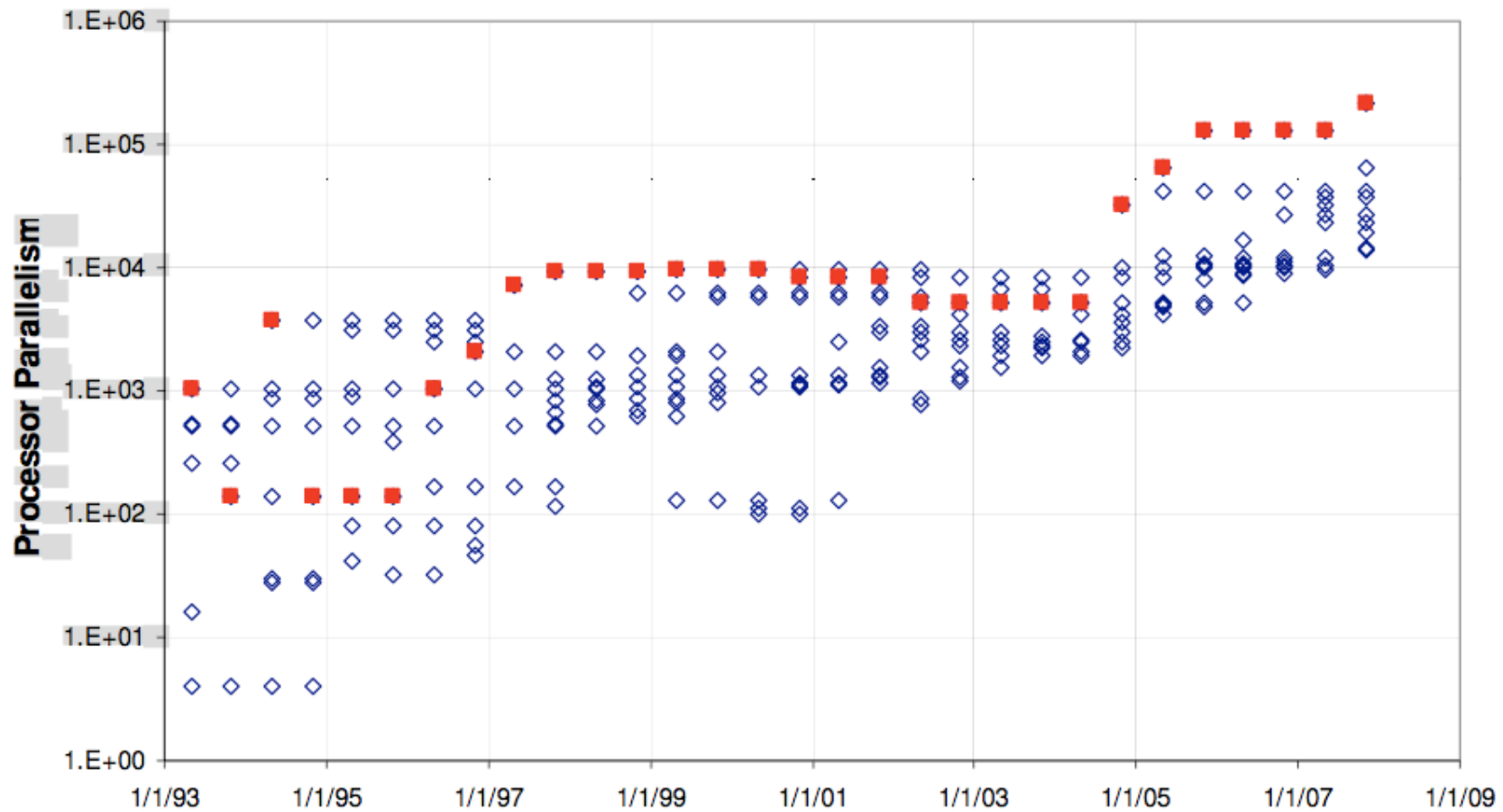
- Number of transistors per chip doubles every 18 months
- Thread performance does not improve; number of cores per chip doubles
  - clock speed does not increase, due to power constraints
  - diminishing returns on new microprocessor features
- New generation hardware provides better performance on existing application or support new applications that cannot run on old hardware – *only if these applications run in parallel & scale automatically*

# Ubiquitous Parallelism

- Every computer is (or will be soon) parallel
  - including cell phones, embedded processors...
- Every application will run in parallel
- The number of threads in top supercomputers more than doubles every 18 months
- Increasingly, computations migrate to cloud
  - lower cost of ownership
  - lower power cost

# Growth in Number of Processors

- **Top** & **top 10**



# A Google-Sized Cloud



**> 100 MW**



# Illinois-Size Petascale Computing Facility



# Parallel @ Illinois – The Human Side



# Extreme Scale Computing

- What applications can use  $\gg 1$  M concurrent threads?
- How does one program such applications?
- How does one deal with increased failure frequency?



INSTITUTE FOR ADVANCED COMPUTING  
APPLICATIONS AND TECHNOLOGIES

PARALLEL@ILLINOIS

# Cloud Computing

- How does one handle applications that are both compute intensive and data intensive?
- How does one support efficient on-demand computing?



# UPCRC

- What client applications can use 100's of cores?
- How does one make parallel programming easy?
- How does one build chips with 100's of cores?



# Synergy

- Parallel computing is a common concern from mobile client to exascale computers
- We expect significant technology sharing across the range
  - components (multicore processors, GPUs)
  - parallel programming languages
  - on demand supercomputing
  - ...
- **It is happening at Illinois**