Resource Characterization

Rich Wolski, Dan Nurmi, John Brevik, Graziano Obertelli

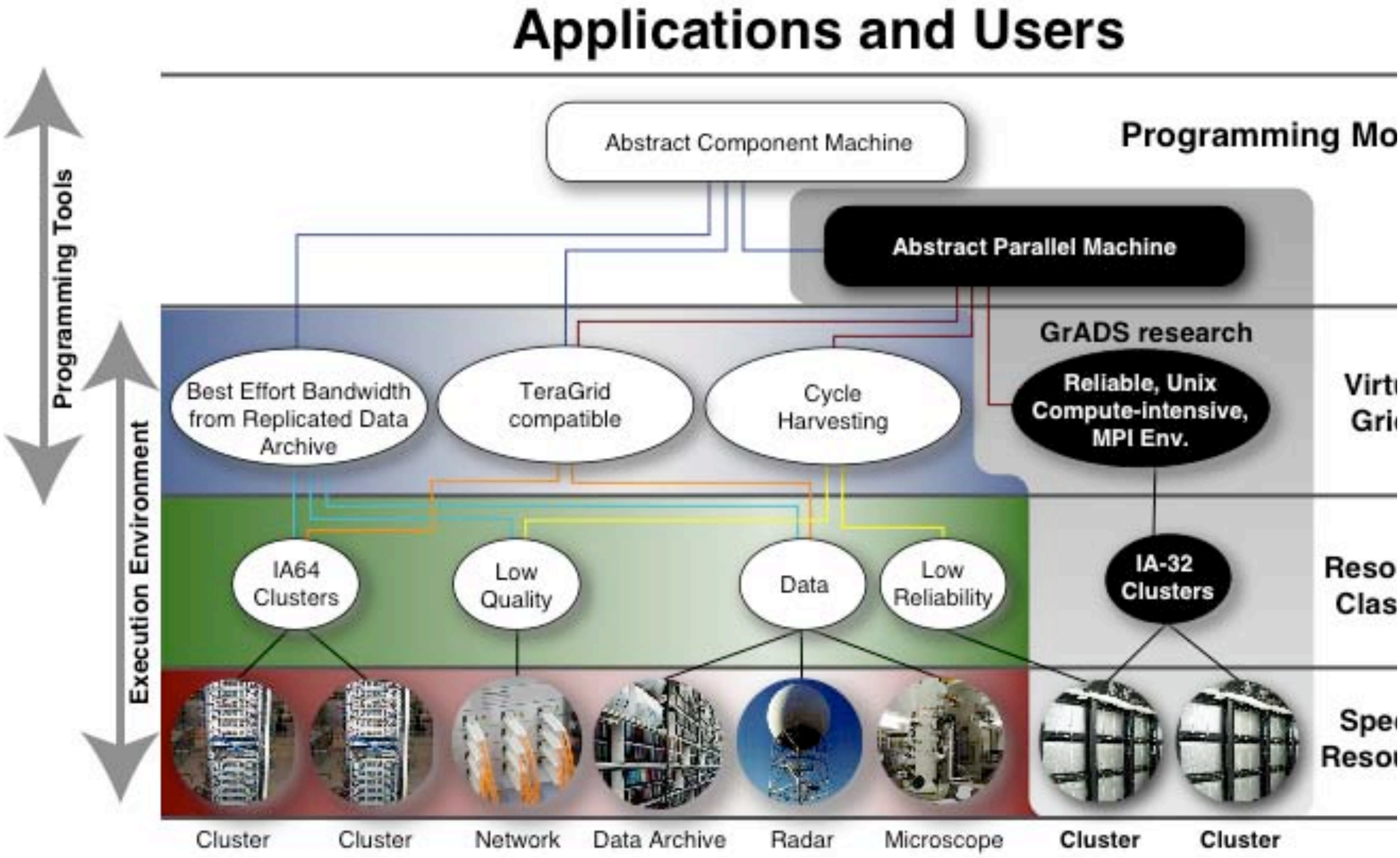
Computer Science Department

University of California, Santa Barbara

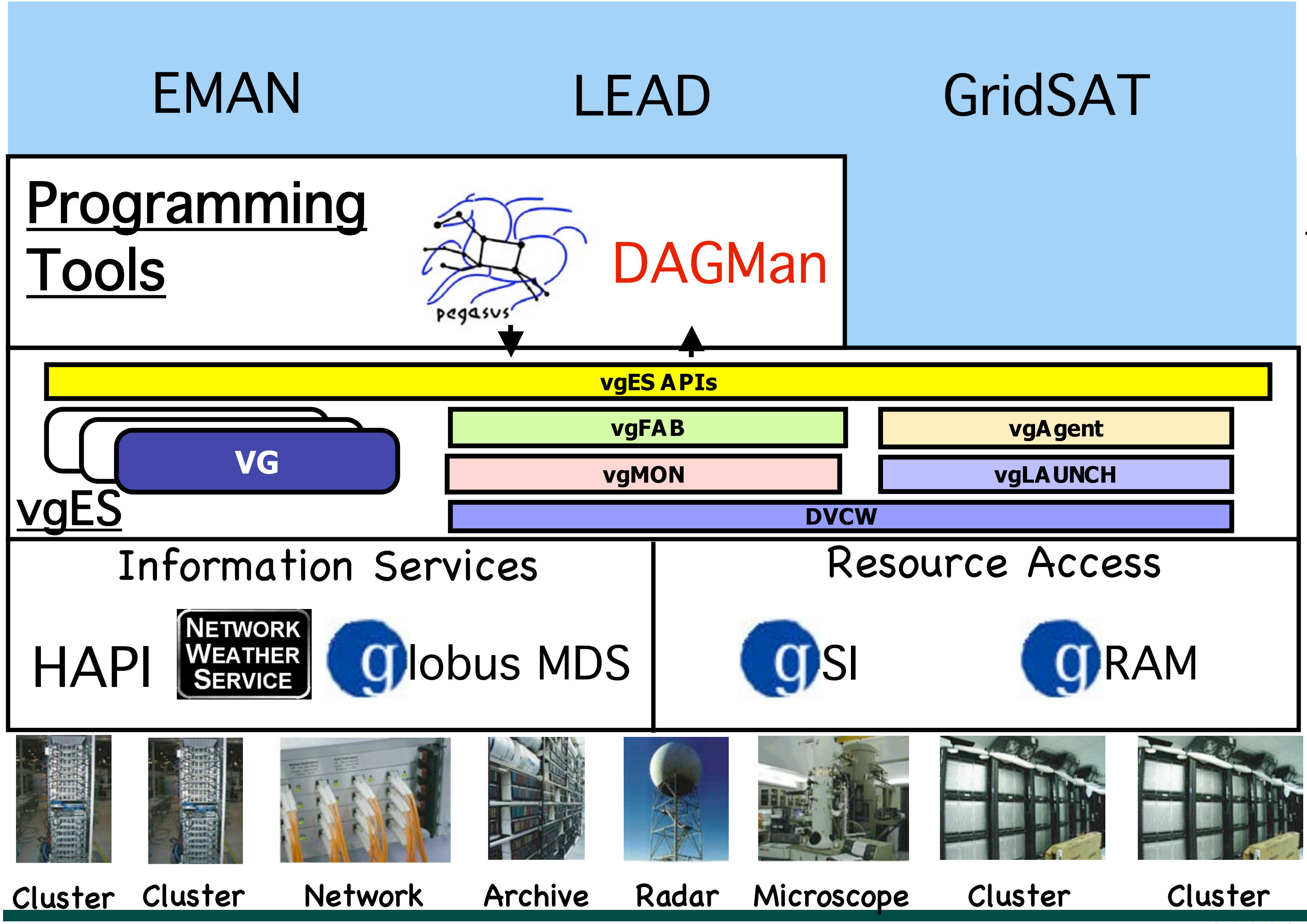
VGrADS Site Visit

April 28, 2005

VGrADS Vision



VGrADS Functional Decomposition (so far)



EMAN

LEAD

GridSAT

Programming Tools

pegasus

DAGMan

vgES APIs

VG

vgFAB

vgAgent

vgMON

vgLAUNCH

DVCW

vgES

Information Services

HAPI

NETWORK WEATHER SERVICE

globus MDS

Resource Access

gSI

gRAM

Cluster

Cluster

Network

Archive

Radar

Microscope

Cluster

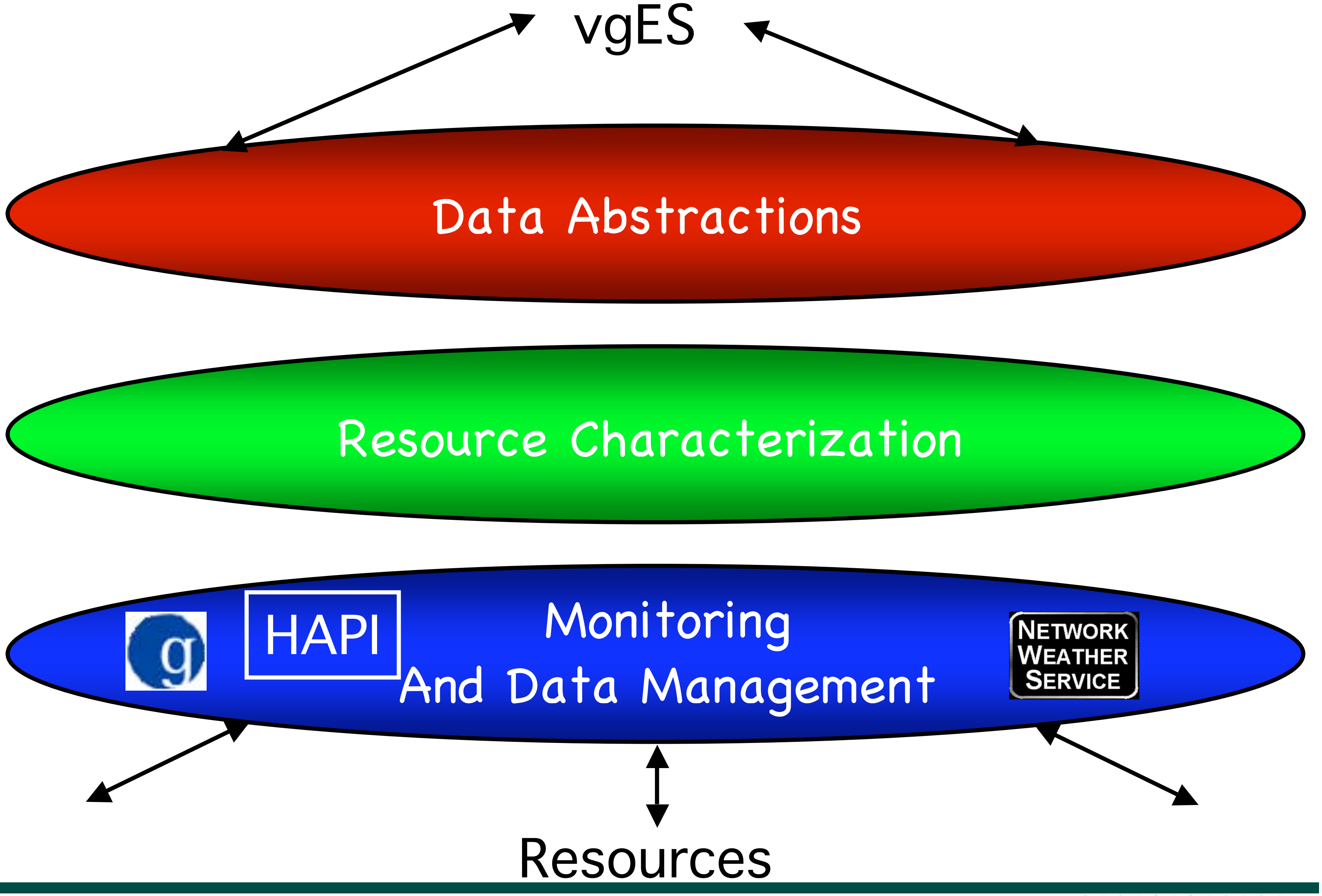
Cluster

Progr
Prepara

Virtu
Grid

Grid
Servic

VGrADS Information Services



vgES

Data Abstractions

Resource Characterization

HAPI

Monitoring
And Data Management

Network
Weather
Service

Resources

Resource Characterization

- **Abstract description of resources in terms of program accessible attributes**
- **Quantitative Approach: use automatic statistical methods to capture the dynamics of changing resource characteristics**
  - Monitor data is plentiful and noisy
  - Must summarize the quantitative behavior of each resource
    - Reduce the complexity associated with using quantitative performance and reliability readings
    - Summaries must be statistically "reliable" to enable effective program-based reasoning and debugging (confidence measures
- **Key Questions:**
  - *Can we make effective quantitative characterizations?*
  - *Can we deliver the characterizations scalably and fast enough?*

Characterization Research

- **Leverage previous work:**
  - NWS makes time series predictions for characteristics that are well-modeled as continuously changing levels
    - Network BW and Latency (end-to-end)
    - CPU load
    - Available memory
- **New Research: focus on characteristics that do not fit time-series models well**
  - Resource availability and failure prediction
    - Predicted duration-until-next-failure as a quantitative characterization (Reed+Wolski)
    - Used to schedule checkpoints (Poster by Dan Nurmi)
  - Batch Queue Wait time prediction
    - New approach to an old problem

Batch Queue Wait Time

- Problem: The vgES and Programming Tools need to know how long individual jobs will wait before they will acquire the resources then need
  - Perceived execution time is really affected by wait times
  - Choose the "provisioning" method that best serves the application needs => VGrADS scheduling
- Goal: Rigorous confidence bounds on the amount of time a specific job will wait in a batch queue before it is scheduled a cluster or parallel machine.
  - Statistical nature implies that a quantifiable confidence range is necessary
  - Need an answer that applies to an individual job Previous work: fi statistical model then predict
  - Smith, Taylor, Foster (IPDPS, 1999), Downey (IPPS 1997)
  - Feitelson, http://www.cs.huji.ac.il/~feit/parsched

Modeling and Prediction are Different

- **Model fitting**
  - From the distribution, calculating an expectation is possible
  - Probably not what a user or scheduler needs
    - Mean and variance do not explain what is going to happen to specific job
- *"At most how long will I have to wait before my job runs?"*
  - The answer is a percentile
- *"At most how long will I have to wait before my job runs wit 95% confidence?"*
  - The answer is the 95th percentile
- Goal: estimate the percentiles without explicitly fitting a mod
- Better Goal: estimate percentiles and quantified confidence bounds
  - *Statistical certainty at specified confidence levels*

The Brevik Method

- John Brevik's invention based on Binomial distribution
  - Probability that exactly j values are below qth quantile is

$$\binom{n}{j} \cdot (1-q)^{n-j} \cdot q^j$$

- Probability that k or fewer values are less than the qth quan

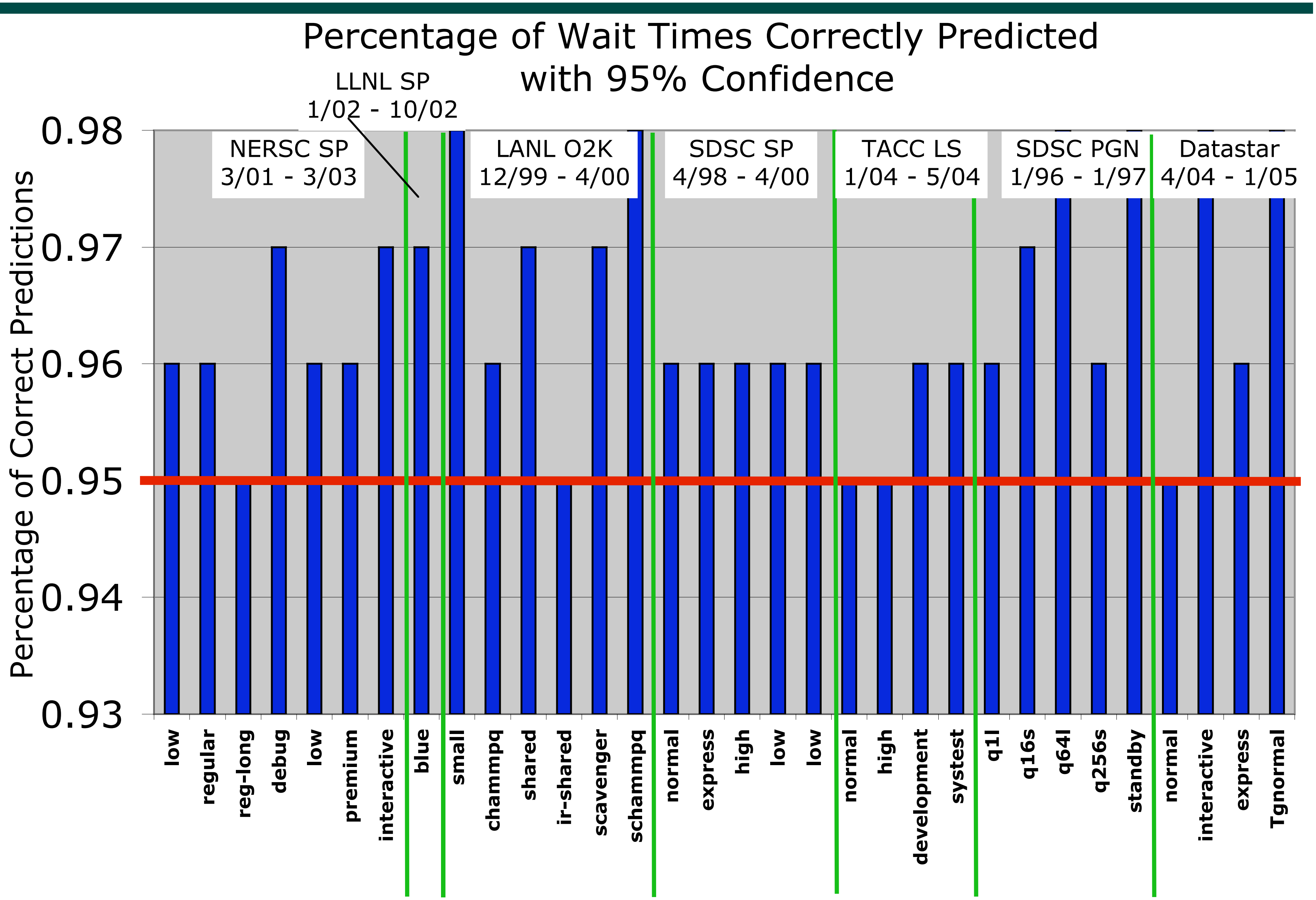
$$\sum_{j=0}^{k} \binom{n}{j} \cdot (1-q)^{n-j} \cdot q^j$$

- Very robust requiring few sample points (not understood)
- Requires multi-precision arithmetic to calculate because n and can be quite large

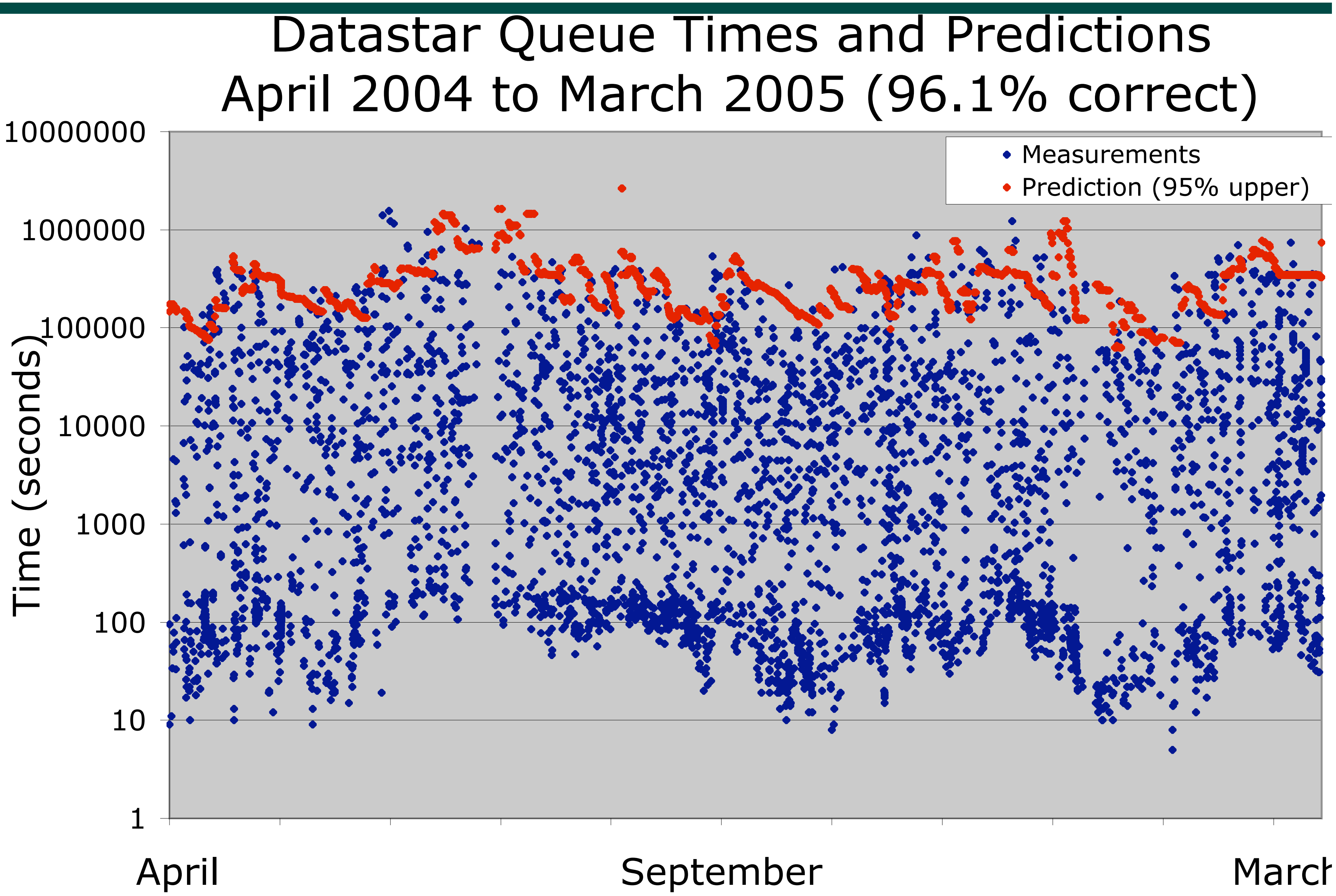
How Well Does it Work?

- Examine the batch queue logs that record wait time
- Choose a quantile and a confidence level
  - 0.95 quantile with 95% confidence
- For each job
  - Calculate the upper limit on the quantile
  - Observe whether job's wait time is less than that limit
- For the entire trace, record the percentage of job wait time that are less than the prediction
  - Value should be less than quantile if method is working
- 5 sites and machines (NERSC, LANL, LLNL, SDSC, TACC)
- 9 years (96 through 05)
- 1,200,000+ jobs

Quantifiable Confidence

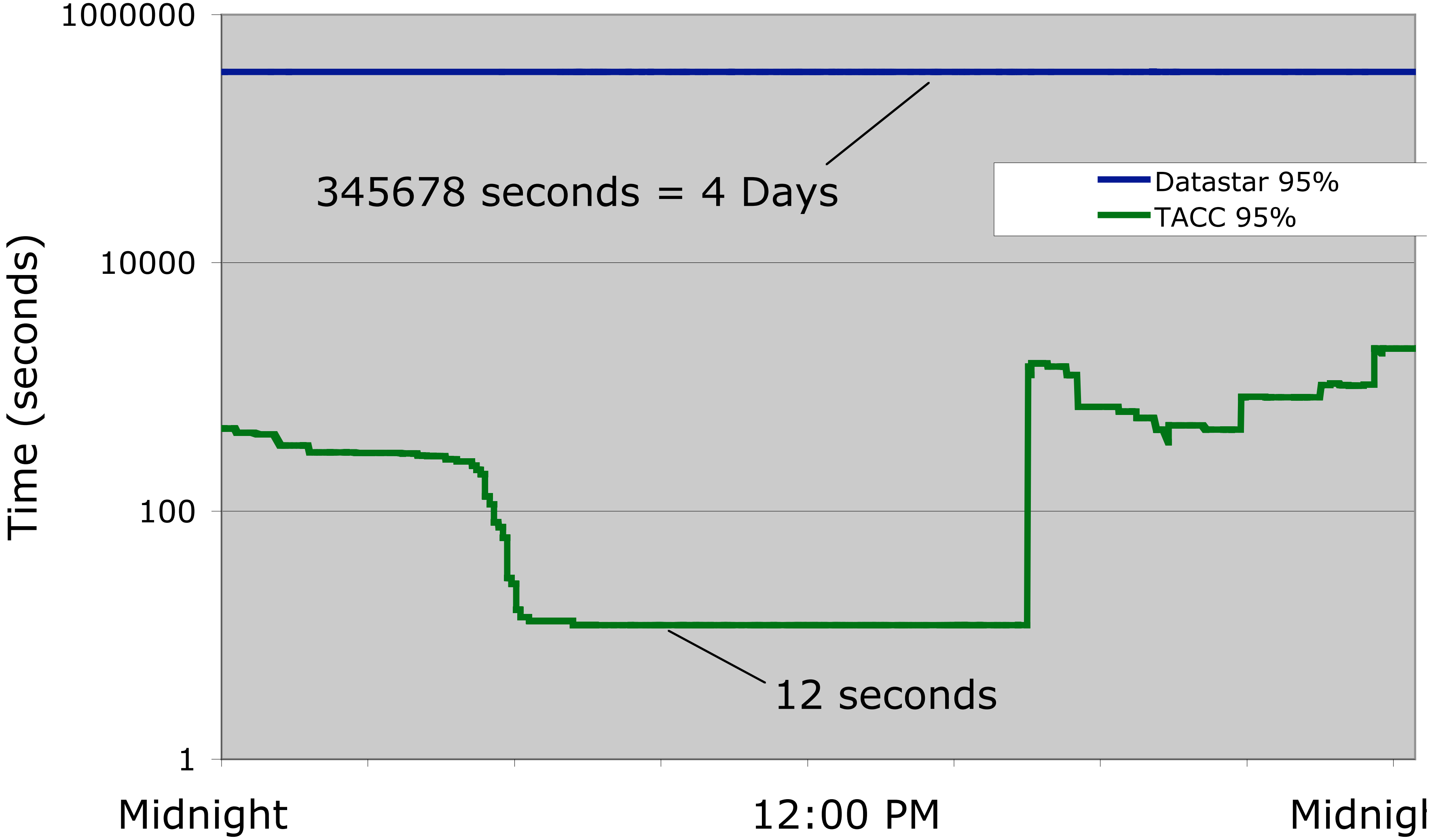


Capturing Dynamics



Choosing the Best Worst Case

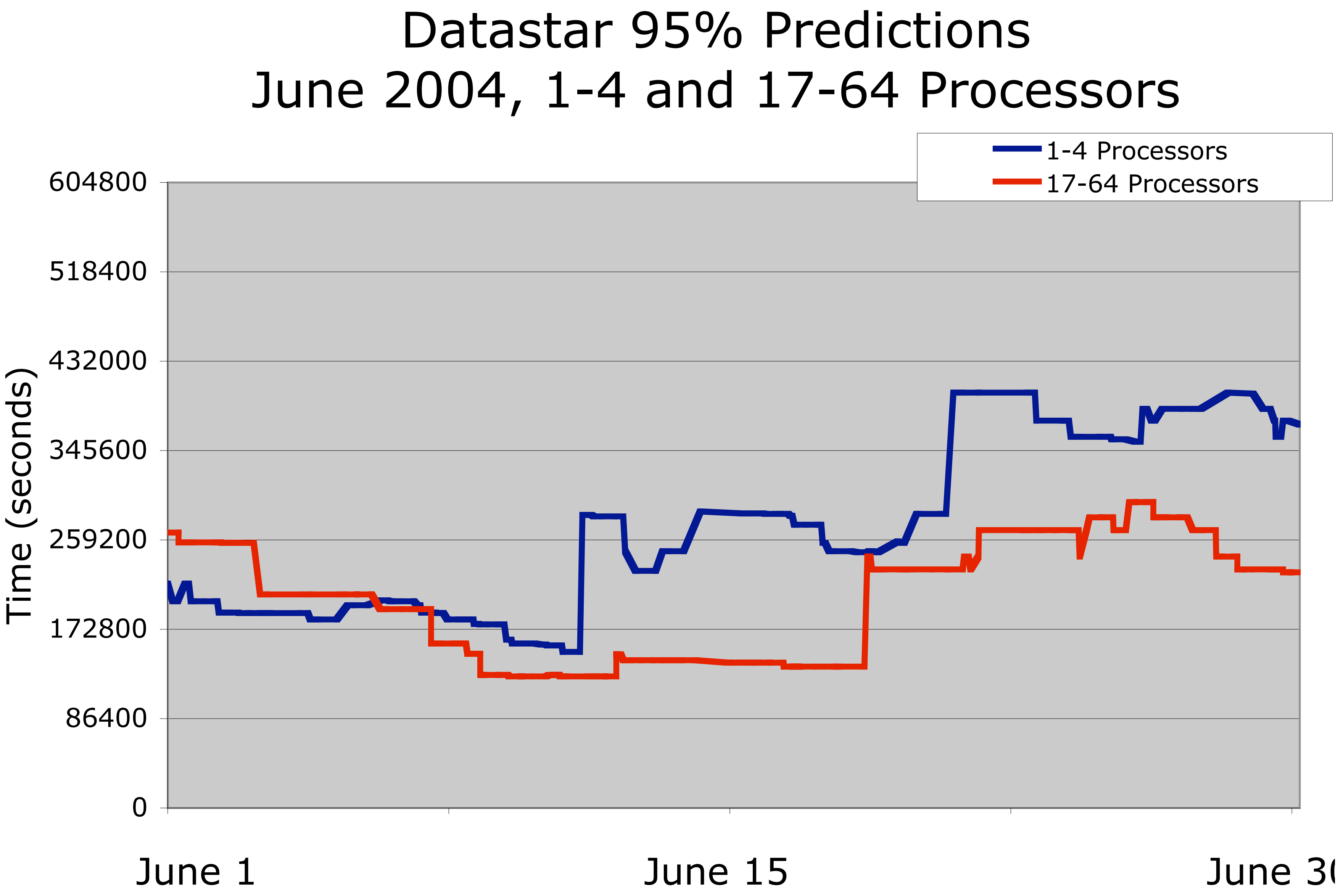
TACC and Datastar Upper 95% Predictions
Thursday February 24, 2005



345678 seconds = 4 Days

12 seconds

Choosing the Best Number of Processors



Datastar 95% Predictions
June 2004, 1-4 and 17-64 Processors

A Batch of Results

- Brevik Method can predict quantiles with specified levels of confidence
  - Must control history adaptively to handle non-stationarity
  - Robust and data frugal enough to work for processor counts too (much harder)
- Combinations of quantiles provide a qualitative way to evalua resources
  - If median and 95th percentile are lower, chances are job will start soo
- Quantiles provide a quantitative way to predict possible outcomes
  - 45% chance that a job will start between the median and the 95th percentile
- Possibly New Scheduling Research: Quantitative Contingency Scheduling
  - Build a schedule with contingencies based on quantiles
  - Adjust based on conditional predictions

Delivering the Good News

- Virtualization: construct forecast "snapshot" of resource characteristics for vgES
  - Use forecasting to cover asynchrony
  - Use statistical similarity to improve scalability
- Test: NWS Network "Doppler Radar"
- For VGrADS
  - Replicated name servers (UCSD and Rice)
  - Replicated data caches (UCSB and Rice)
  - 300 microseconds/forecast over 100mb local ethernet
- VGrADS testbed: The Movie

Conclusions

- New automatic resource characterizations
  - New approach to batch queue and machine availability
  - Lead to new scheduling techniques (c.f. Dan Nurmi Poster)
  - Quantifiable confidence levels
- Result: *We provide rigorous bounds on statistical attribute values to vgES and Programming Tools*
- New Information System data structures
  - Scalable and high performance
  - Provide an instantaneous "picture" of the resources
- Result: *Virtualization in the Information System promotes scalability and performance*