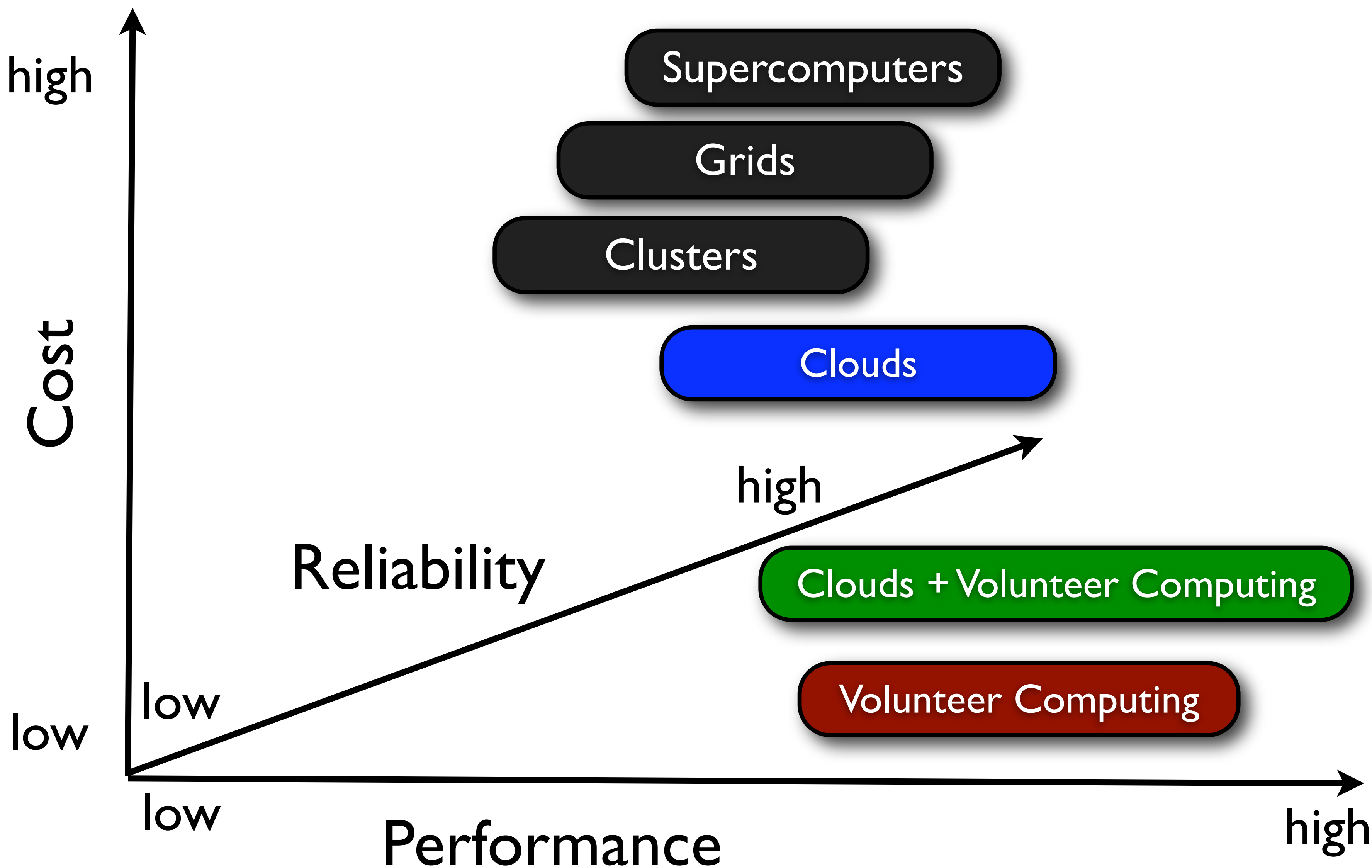
# Volunteer Computing and Cloud Computing: Opportunities for Synergy

Derrick Kondo

INRIA, France

# Performance vs. Reliability vs. Costs



# Volunteer Computing

- Use free resources in Internet and Intranet environments
  - Consists of volunteers PC's mostly over residential network
- About 65 applications from millions of Internet resources

SETI@HOME Folding@home distributed computing Rosetta@home Protein Folding, Design, and Docking

- Projects concentrated in US, Europe, and soon Asia

# Performance Today and Tomorrow

- Today
  - Over 1M active hosts, 13.5 PetaFLOPS [1]
    - 60% of Folding@home FLOPS from GPUs
  - Hosts with multiple CPU's, or [1 TeraFLOP] GPU's

## Distribution of Top CPUs

| Pos. | CPU | # CPU |
|---|---|---|
| 1 | Intel(R) Core(tm)2 Quad CPU Q6600 @ 2.40GHz | 79,839 |
| 2 | Intel(R) Core(tm) i7 CPU 920 @ 2.67GHz | 18,738 |
| 3 | Intel(R) Pentium(R) 4 CPU 3.00GHz | 229,775 |
| 4 | Intel(R) Core(tm)2 Quad CPU Q9550 @ 2.83GHz | 14,013 |
| 5 | Intel(R) Core(tm)2 CPU 6600 @ 2.40GHz | 63,214 |
| 6 | Intel(R) Pentium(R) 4 CPU 2.80GHz | 164,154 |
| 7 | Intel(R) Core(tm)2 Quad CPU @ 2.40GHz | 14,350 |
| 8 | Intel(R) Core(tm)2 Duo CPU E8400 @ 3.00GHz | 46,039 |
| 9 | Intel(R) Core(tm)2 Quad CPU Q9450 @ 2.66GHz | 7,598 |
| 10 | Intel(R) Pentium(R) 4 CPU 3.20GHz | 99,546 |

## Top Partcipant of SETI@Home:

8 virtual processors

(Intel Quad CPU Q9450 @ 2.66GHz),

5 GPUs

(NVIDIA GeForce GTX295, 895MB)

[1] [BOINC+Folding@home, March 2010]

# Performance Today and Tomorrow

- Tomorrow
  - 1 billion PCs today, 2 billion in 2015
    - Tesla 4 TeraFLOPS Personal Supercomputer at affordable prices
  - How to get 1 ExaFLOPS: 4M 1 TFLOPS GPUs * 0.25 availability
  - How to get 1 Exabyte: 1M PC disks * 1 TB

# History of volunteer computing

1995 2000 2005 now

*Applications*

distributed.net, GIMPS

SETI@home, Folding@home

Climateprediction.net
Predictor@home
IBM World Community Grid
Einstein@home
Rosetta@home
...

*Middleware*

Academic: Bayanihan, Javelin, ...

Commercial: Entropia, United Devices, ...

BOINC

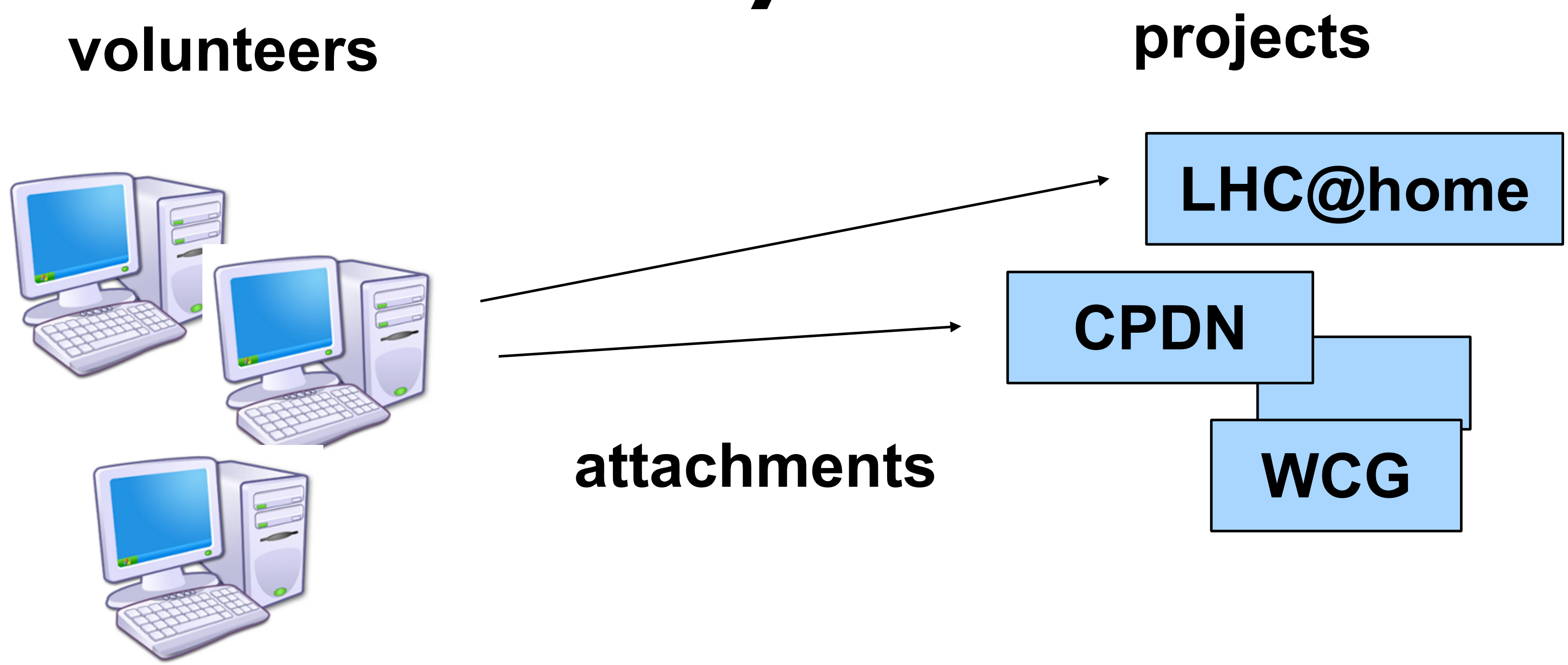
# What is BOINC?

- Berkeley Open Infrastructure for Network Computing
  - Founded and directed by David P. Anderson
- Middleware for volunteer computing
- NSF supported (2002-present)
- Open source (LGPL)
  - http://boinc.berkeley.edu

# Middleware Design Challenges

- Volatility
  - Availability
- Heterogeneity
- Accessibility
- Scalability
- Security

# The BOINC computing ecosystem




- Projects compete for volunteers
- Volunteers make their contributions count
- Optimal equilibrium

# What apps work well?

- Bags of tasks
  - parameter sweeps
    - simulations with perturbed initial conditions
  - compute-intensive data analysis
- Job granularity: minutes to months
- Native, legacy, Java, GPU
  - soon: VM-based

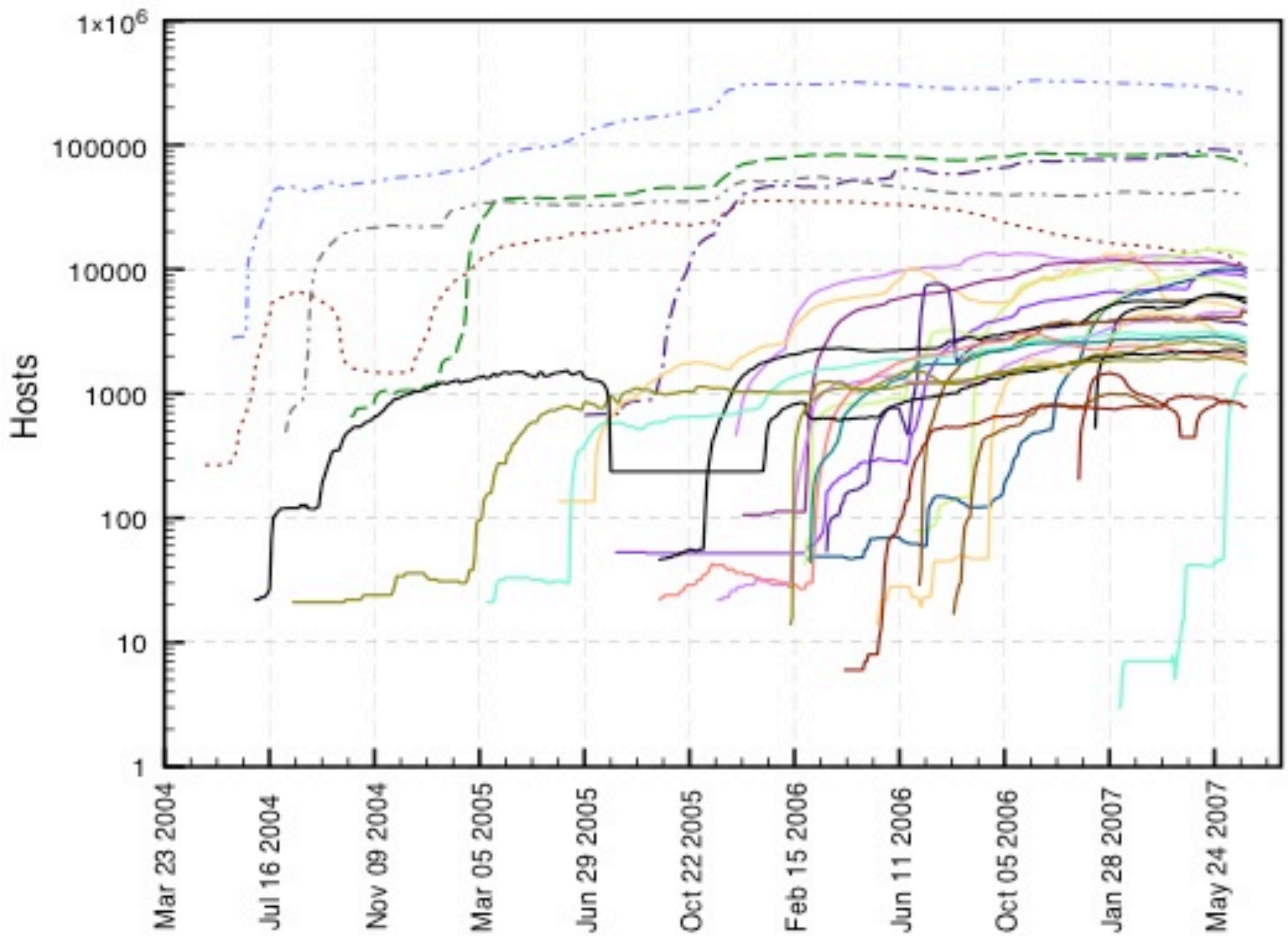
# Example projects

- Climateprediction.net
- Rosetta@home
- IBM World Community Grid
- Primegrid

# Creating a volunteer computing project

- Server
  - Set up a server
  - Develop software for job submission and result handling
  - System, DB admin (Linux, MySQL)
- Client
  - Port applications, develop graphics
- Marketing
  - Develop web site
  - Publicity, volunteer communication

# How many CPUs will you get?




| Project | Hosts upper bound |
|---|---|
| SETI@Home | 350000 |
| Einstein@Home | 90000 |
| Rosetta@home | 90000 |
| Climatepredication.net | 55000 |
| Predictor@home | 35000 |
| SIMAP | 14000 |
| Spinhenge@home | 14000 |
| SZTAKI Desktop Grid | 14000 |
| QMC@home | 12000 |
| Tanpaku | 10000 |
| Proteins@home | 10000 |
| Malaria Control | 9000 |
| Chess960@home | 8000 |
| ABC@home | 7000 |
| μFluids | 6000 |
| Leiden Classical | 5000 |
| Nano-Hive@home | 4500 |
| Rectilinear Crossing Number | 4500 |
| SETI@home Beta | 3500 |
| Seasonal Attribution | 3000 |
| XtremLab | 3000 |
| BURP | 2600 |
| Pirates@Home | 2500 |
| RALPH@home | 2000 |
| ABC@home beta | 1500 |
| The Lattice Project | 1400 |
| RenderFarm@home | 1000 |
| VTU@home | 1000 |

- 3 projects >= 90,000 hosts
- 11 projects >= 10,000 hosts
- Long-tailed [Grey]

[HCW ’09]

# Maximizing Participation

- Possible motivations [Oded Nov et al., 2010]:
- Enjoyment (socially) from participation
- Reputation (i.e., credits)
- Value of project's goals
- Raising of ego with project results

- Give ways to interact socially with (national) teams, forums, contests
- Show progress and contribution of user (automatically and real-time)
- Publicize project and major scientific results

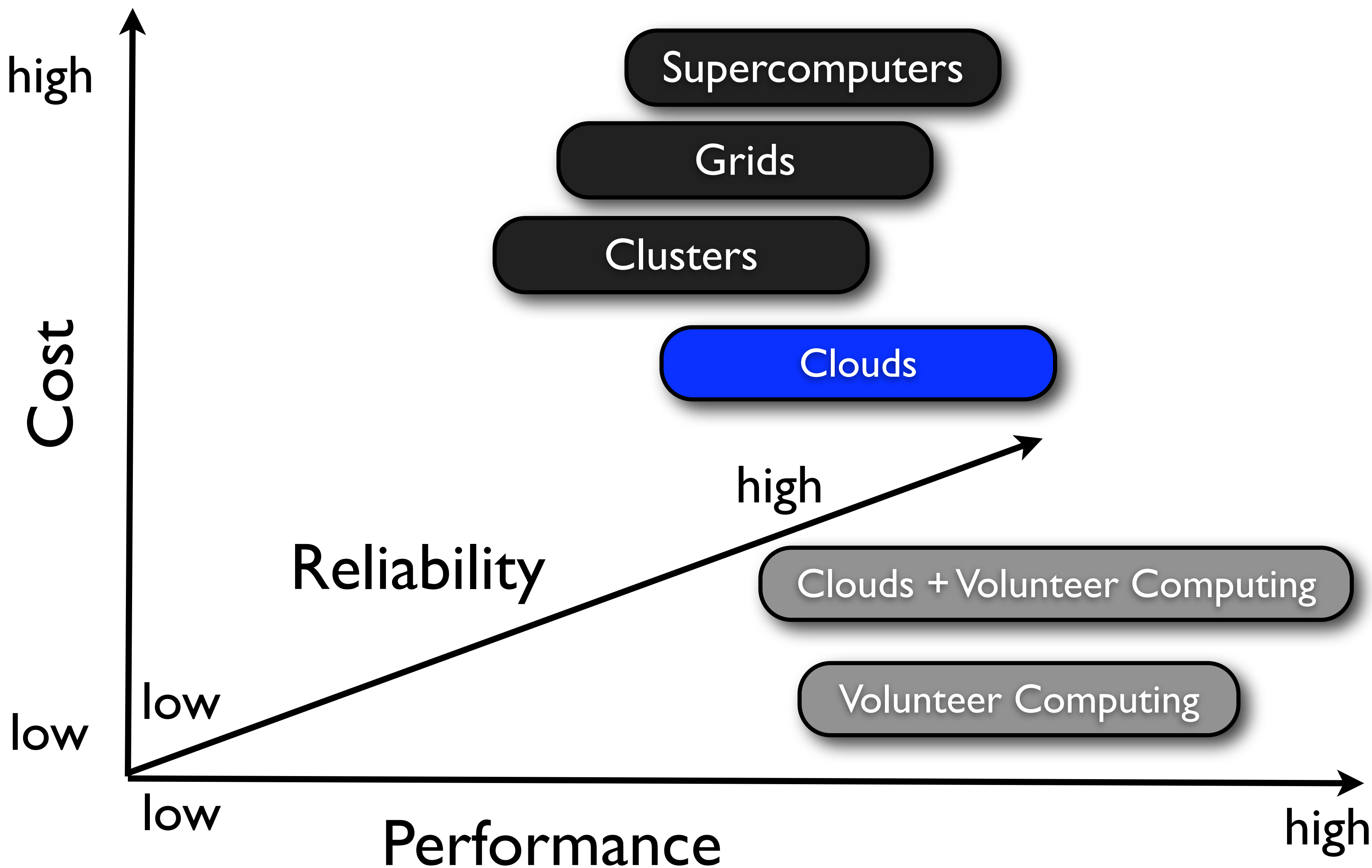
# Organizational issues

- Creating a volunteer computing project has startup costs and requires diverse skills
- This limits its use by individual scientists and research groups
- Better model: umbrella projects
  - Institutional
    - Lattice
  - Corporate
    - IBM World Community Grid
  - Community
    - AlmereGrid

# Summary

- Volunteer computing is an important paradigm for high-throughput computing
  - price/performance and performance potential
- Low technical barriers to entry (due to BOINC)
- Stimulating user motivation, and project organizational structure is critical

# Performance vs. Costs vs. Reliability



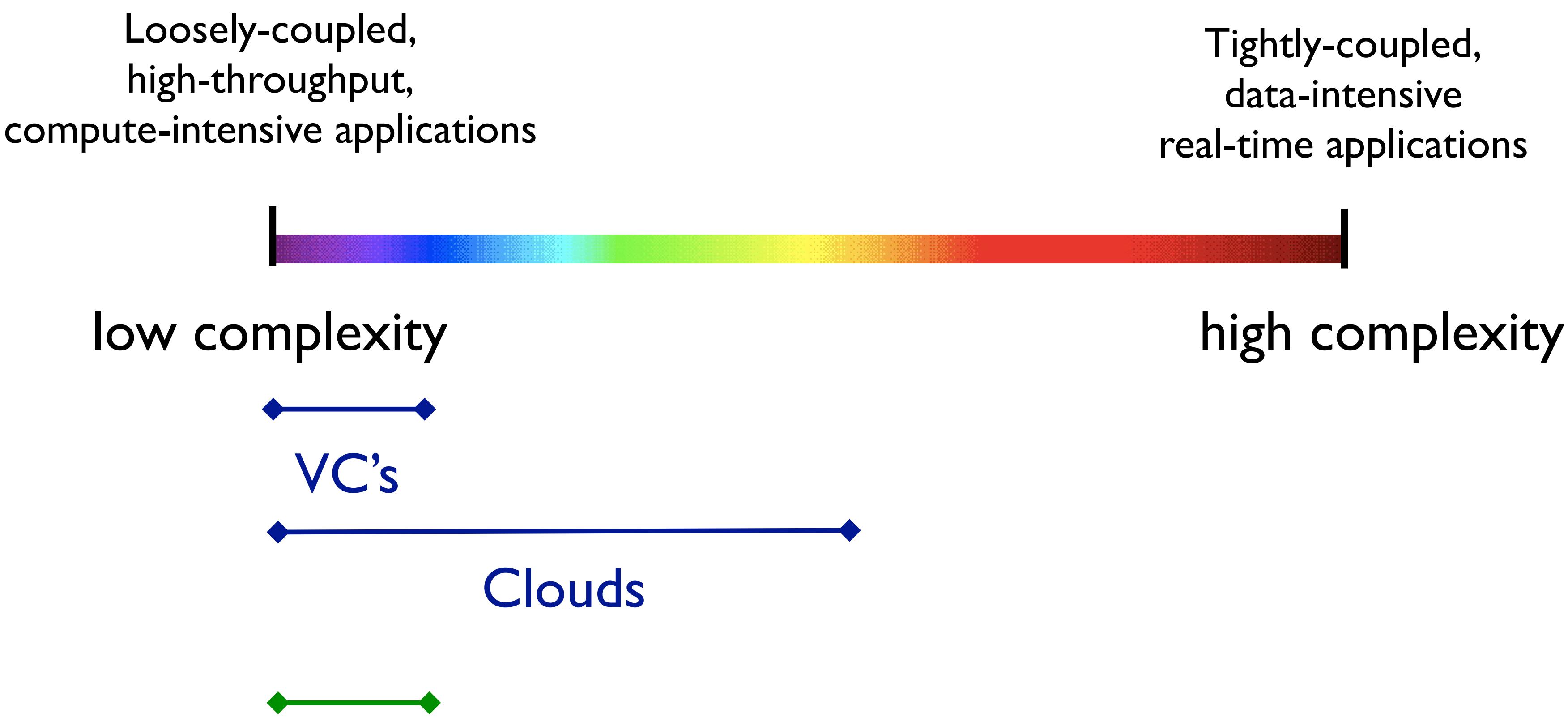
# Cloud Background

- Vision
  - Hide complexity of hardware and software management from a user by offering computing as a service
- Benefits
  - Pay as you go
  - Scale up or down dynamically
  - No hardware management, less software management

# Outline

- Performance tradeoffs
- Monetary tradeoffs
  - Client hosting
  - Server hosting

# Apples to Apples

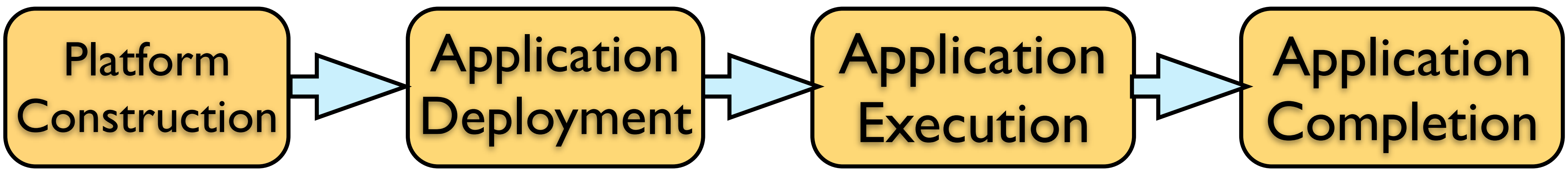



Comparison assuming
*embarrassingly parallel, compute-intensive* applications

# Method

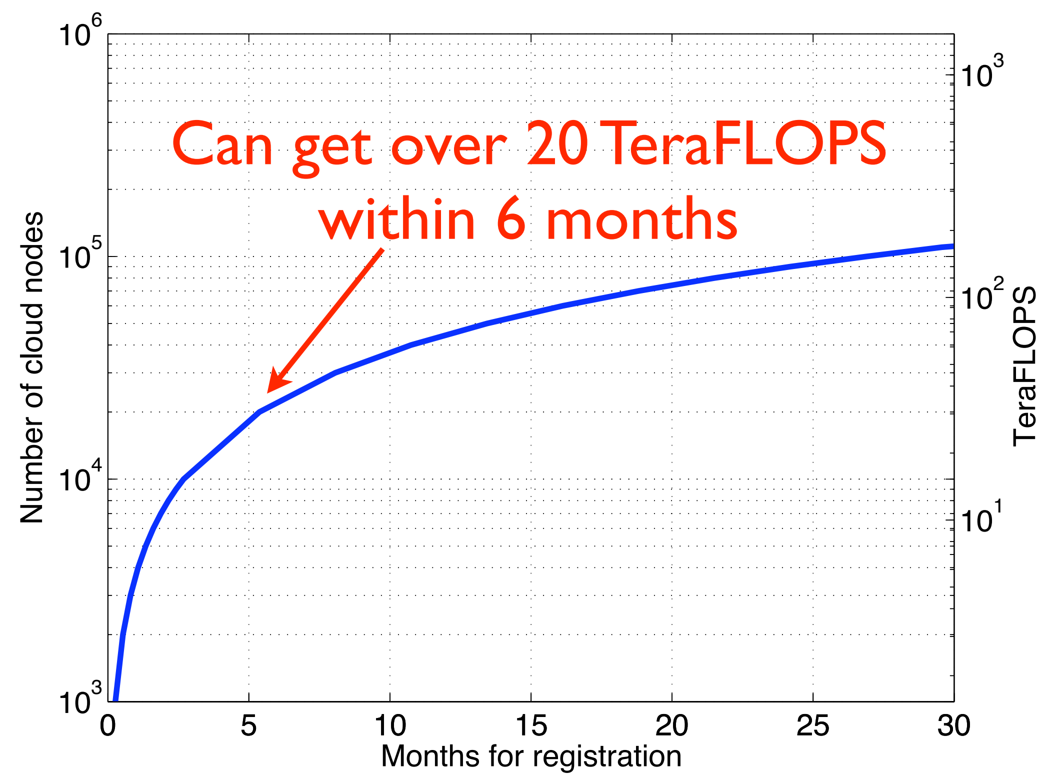
- Use real performance measurements
  - Exported BOINC project data
- Use real costs
  - Large/small BOINC projects (SETI@home / XtremLab)
  - Amazon Elastic Computing Cloud (EC2)

# Stages of Project & Application

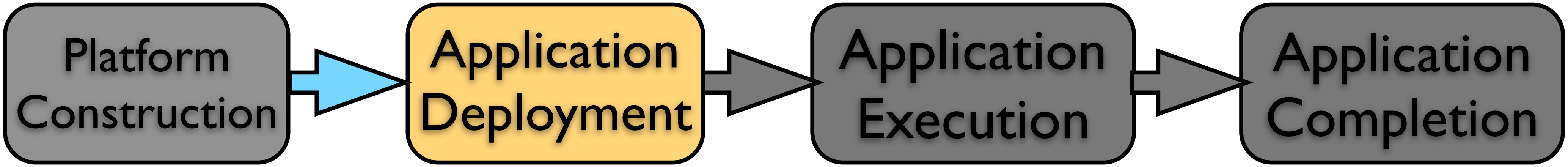


Platform Construction → Application Deployment → Application Execution → Application Completion

How long before I get X TeraFLOPS?

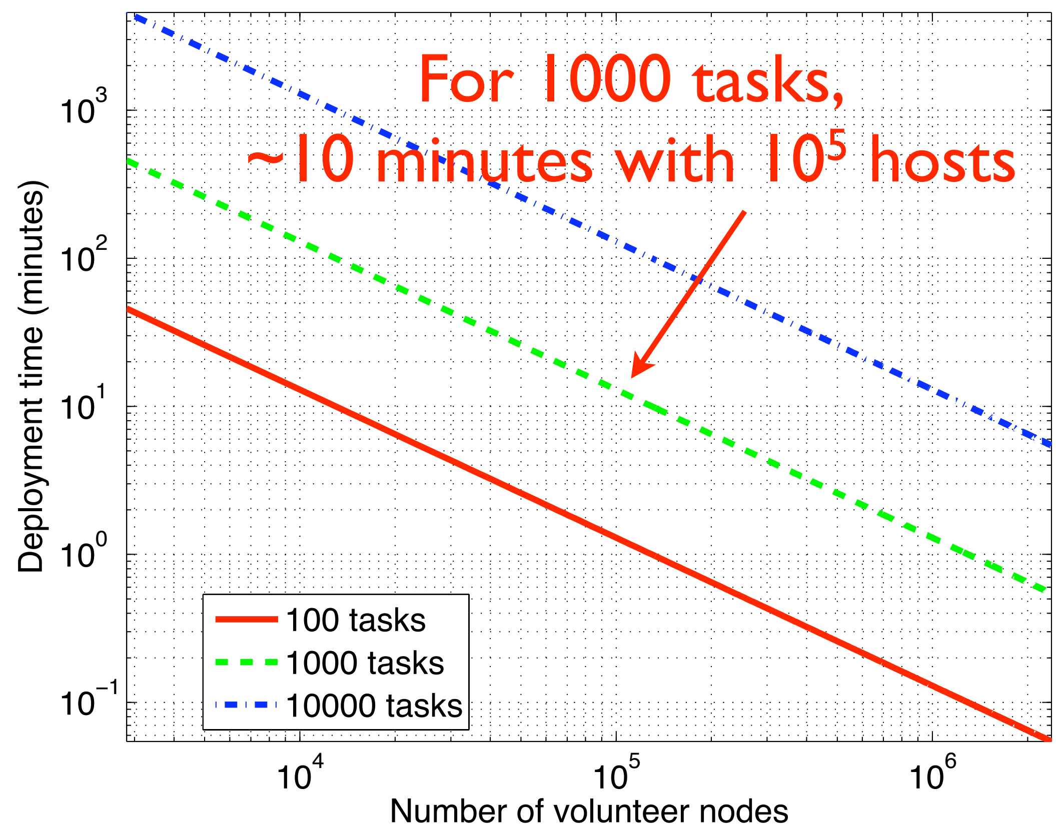



Strategy:
Add to BOINC project list
Press releases
Forum Announcements
Google Ad Sense
Respond to users (leverage volunteers)




How long to deploy my batch of tasks needing faster response time?

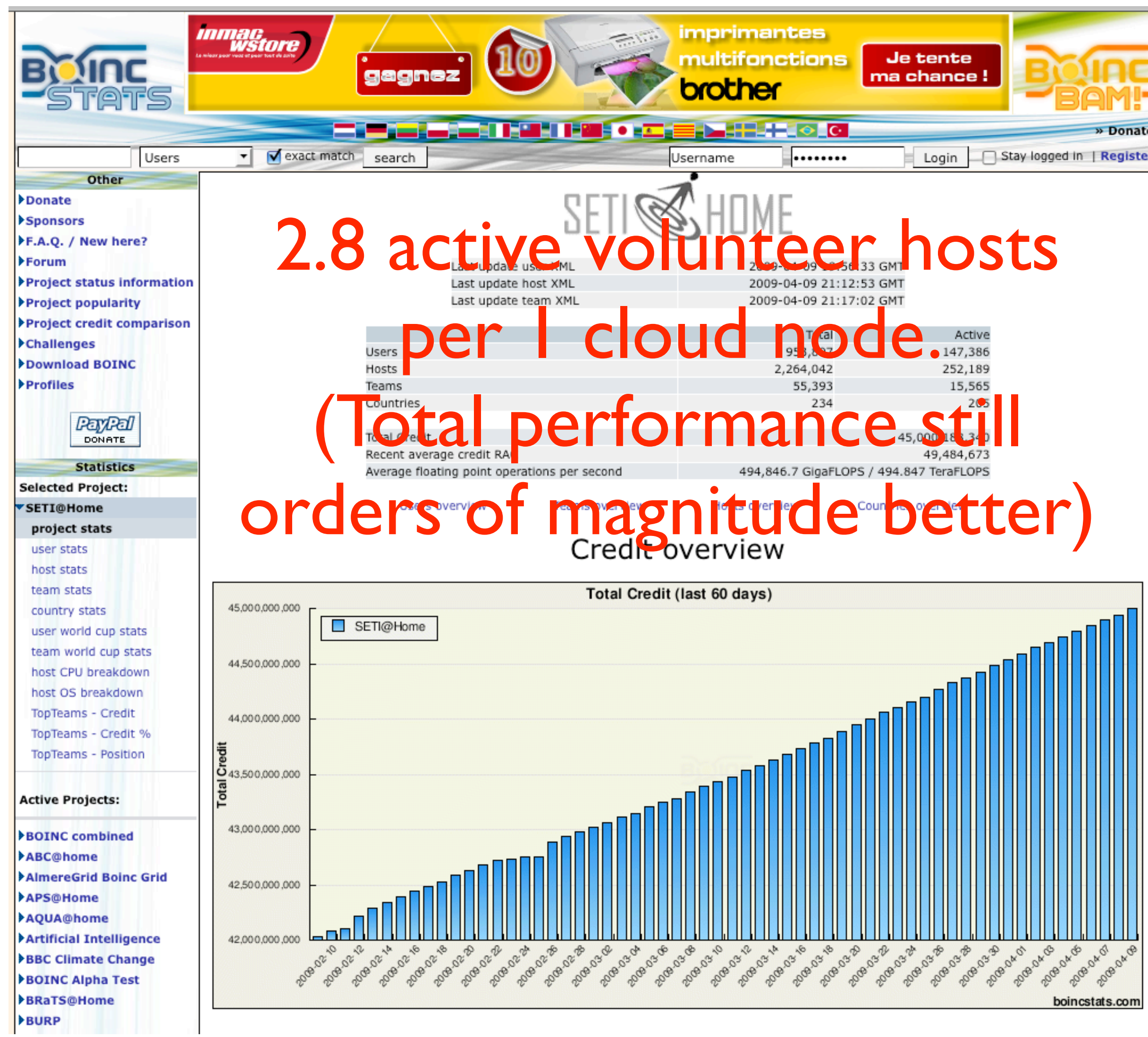
Strategy:
Specify lower latency bounds [Heien et al.]



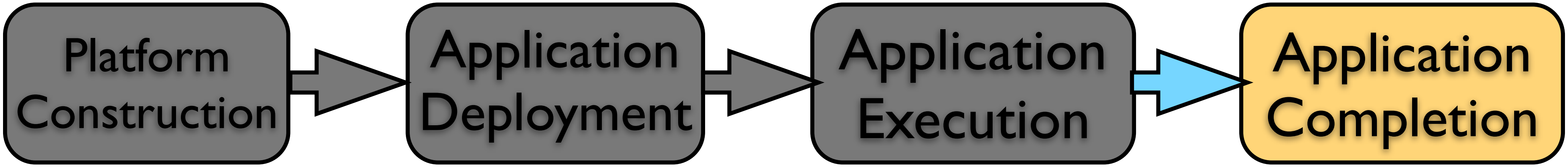
Platform Construction → Application Deployment → Application Execution → Application Completion

How many volunteer nodes are equivalent to 1 cloud node?

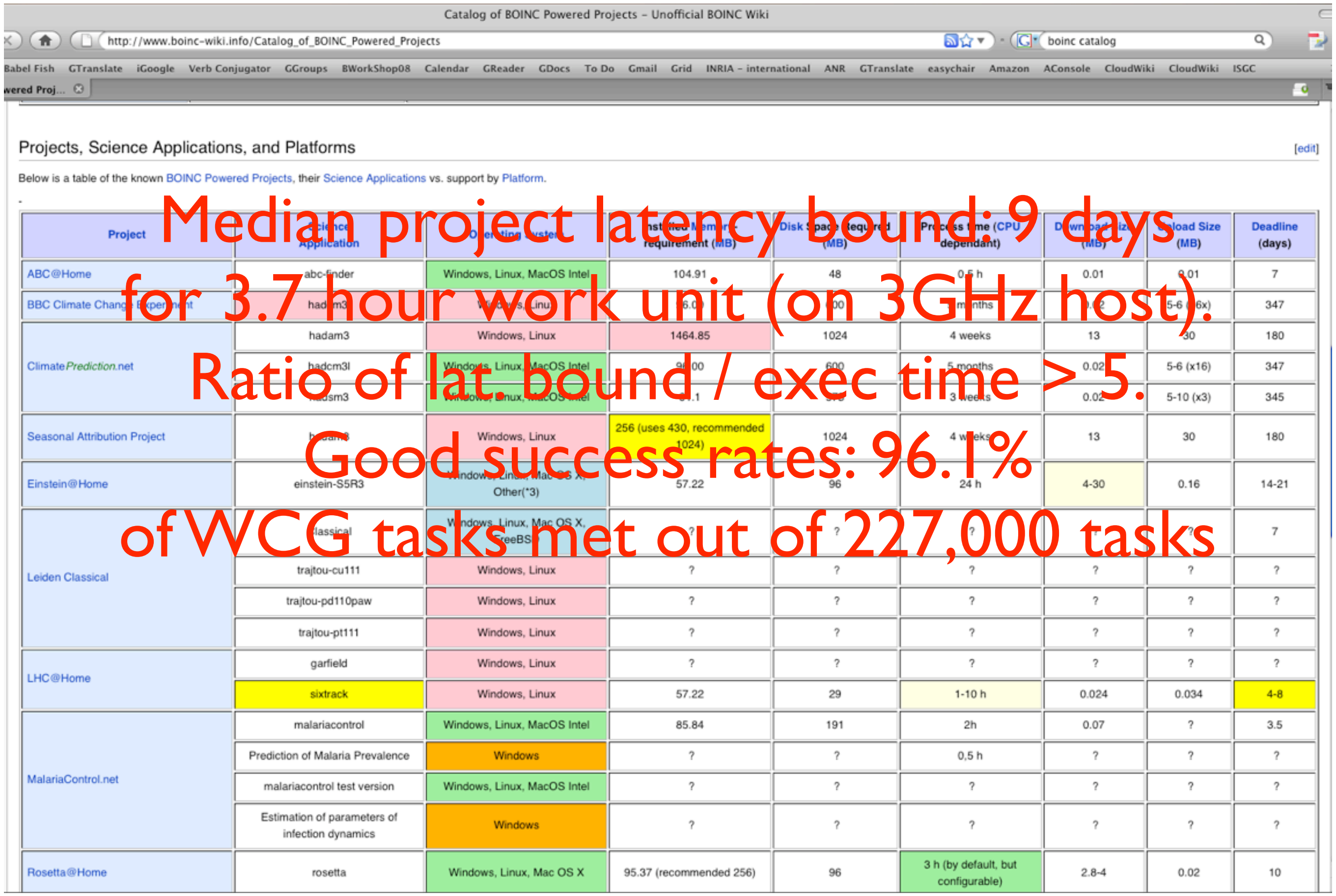
Strategy:
Use statistical prediction of availability




2.8 active volunteer hosts per 1 cloud node.
(Total performance still orders of magnitude better)




How long should I wait for task completion?




Median project latency bound: 9 days for 3.7 hour work unit (on 3GHz host). Ratio of lat. bound / exec time > 5.

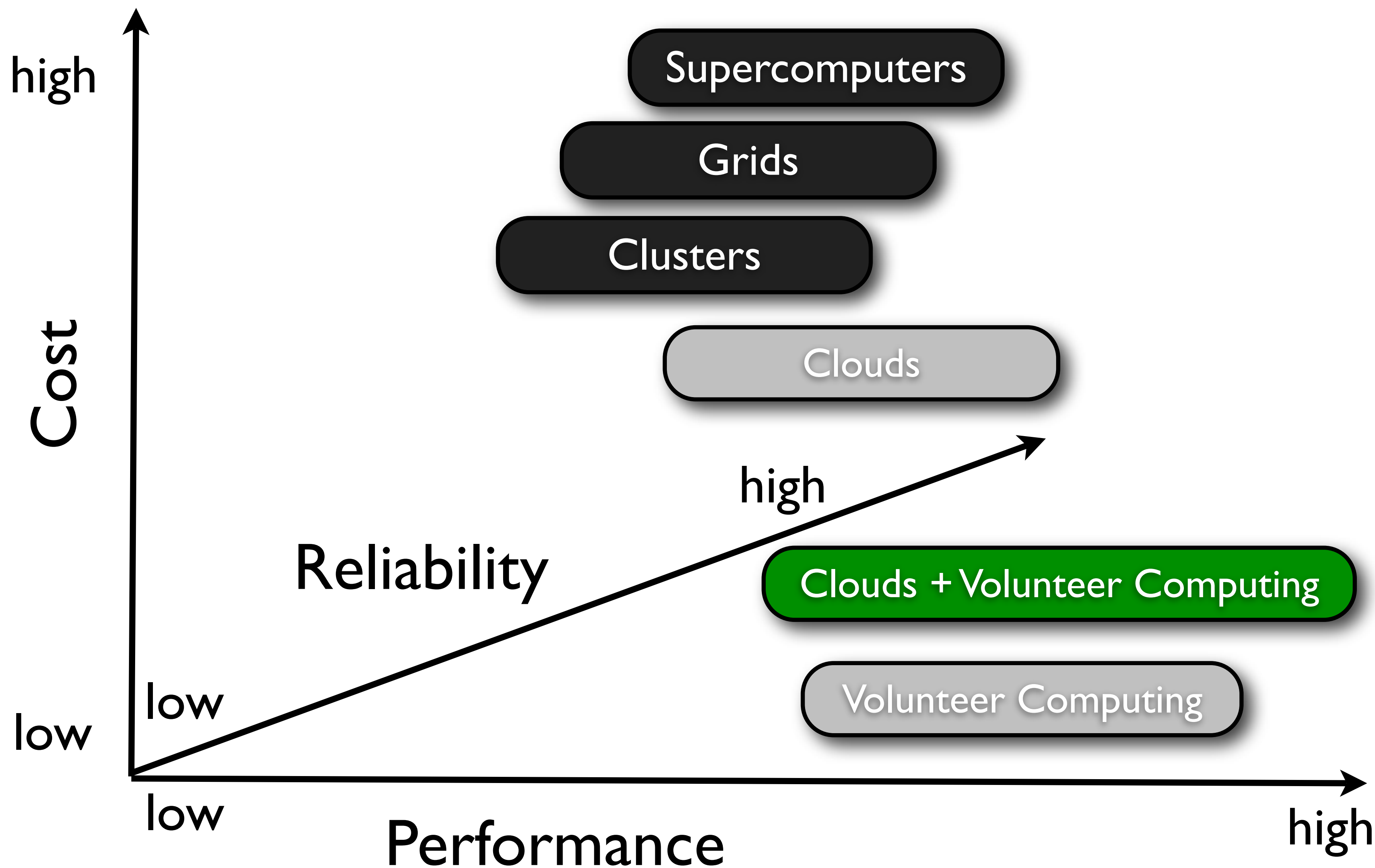
Good success rates: 96.1% of WCG tasks met out of 227,000 tasks

Strategy:
See BOINC Catalog for typical deadlines and compute/comm/mem ratios.

# Monetary Tradeoffs

- Client hosting on cloud
  - Not worth it and never will
- Server hosting on the cloud
  - Possible solution
- More details in HCW, 2009 paper...

# Performance vs. Costs vs. Reliability



# Motivation

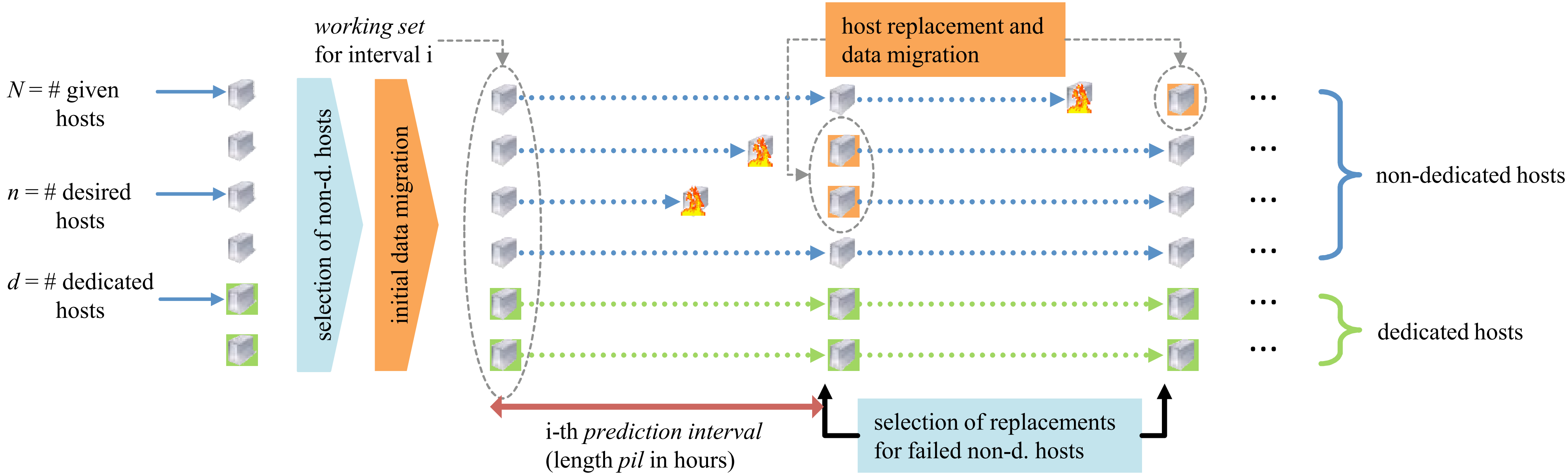
- Non-dedicated resources are cheap, but unreliable
    - VC are cheaper by at least order of magnitude
    - Amazon EC2 Spot instances at least 50% compared to dedicated ones
- Clouds are more expensive but highly reliable
    - Amazon SLA: 99.95% availability
- Applications have different levels of performance, cost, and reliability requirements

# Goal & Approach

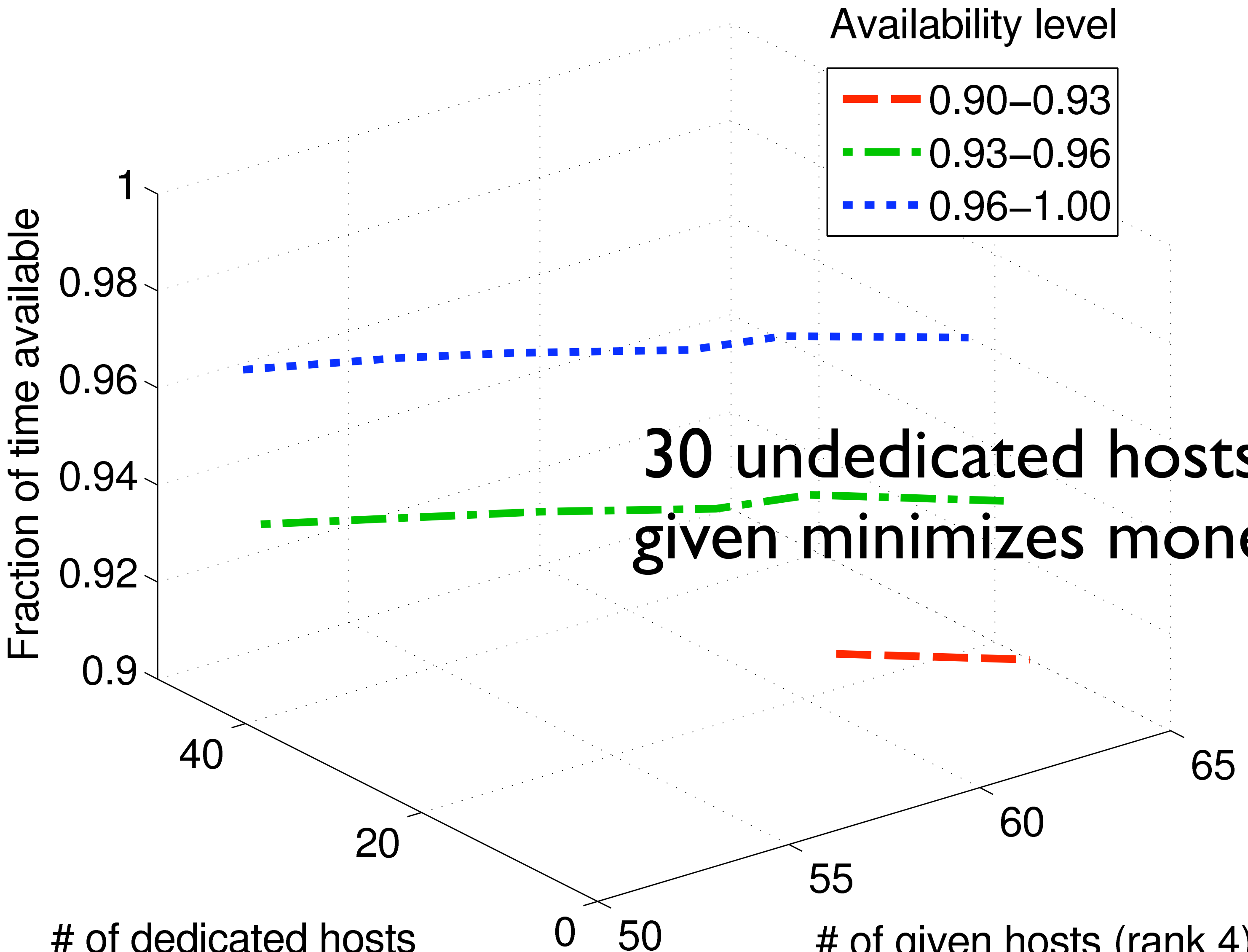
- Given reliability requirements, what is the cheapest *combination* of dedicated and undedicated resources
  - Improve reliability of VC resources with prediction and reliability ranking
  - Compute optimal mixture that satisfies reliability requirement while minimizing cost

# Resource Provisioning using Predictions

- User specifies availability guarantee (ag) and # of desired hosts (n).
- System determines # of given hosts (N) and # of dedicated hosts (d)



# Pareto Optimal
# for Availability >= .90
# for 50 requested hosts




30 undedicated hosts out of 55 given minimizes monetary costs

# Summary

- Prediction: use of ranking to predict availability of a collection of hosts
- Operational model: identify cost-optimal mixture of undedicated and dedicated hosts given reliability requirements

# Overall Summary

- Volunteer Computing
  - Good for high-throughput applications at low cost
- Cloud Computing
  - At least order of magnitude more expensive, but higher reliability
- Volunteer and Cloud Computing
  - Can support applications with diverse reliability requirements while minimizing costs

# Thank you


- Acknowledgement: David P. Anderson's material on BOINC