Distributed Resource Discovery on PlanetLab with SWORD

http://www.swordrd.org/

David Oppenheimer, Jeannie Albrecht
David Patterson, Amin Vahdat
UC Berkeley / UC San Diego

First Workshop on Real, Large Distributed Systems
December 5, 2004

Introduction

- **Increasing number of large-scale distributed systems that run across wide-area networks**
    - content distribution networks
    - peer-to-peer storage
    - distributed games
    - Grid applications

- **Applications have minimum resource requirements to achieve desired QoS**
    - **compute-intensive:** spare CPU, physical mem, disk space
    - **network-sensitive:** positions in network topology near potential users, good network connections among nodes, "interesting" network locations
    - **hybrid:** all of the above

Introduction (cont.)

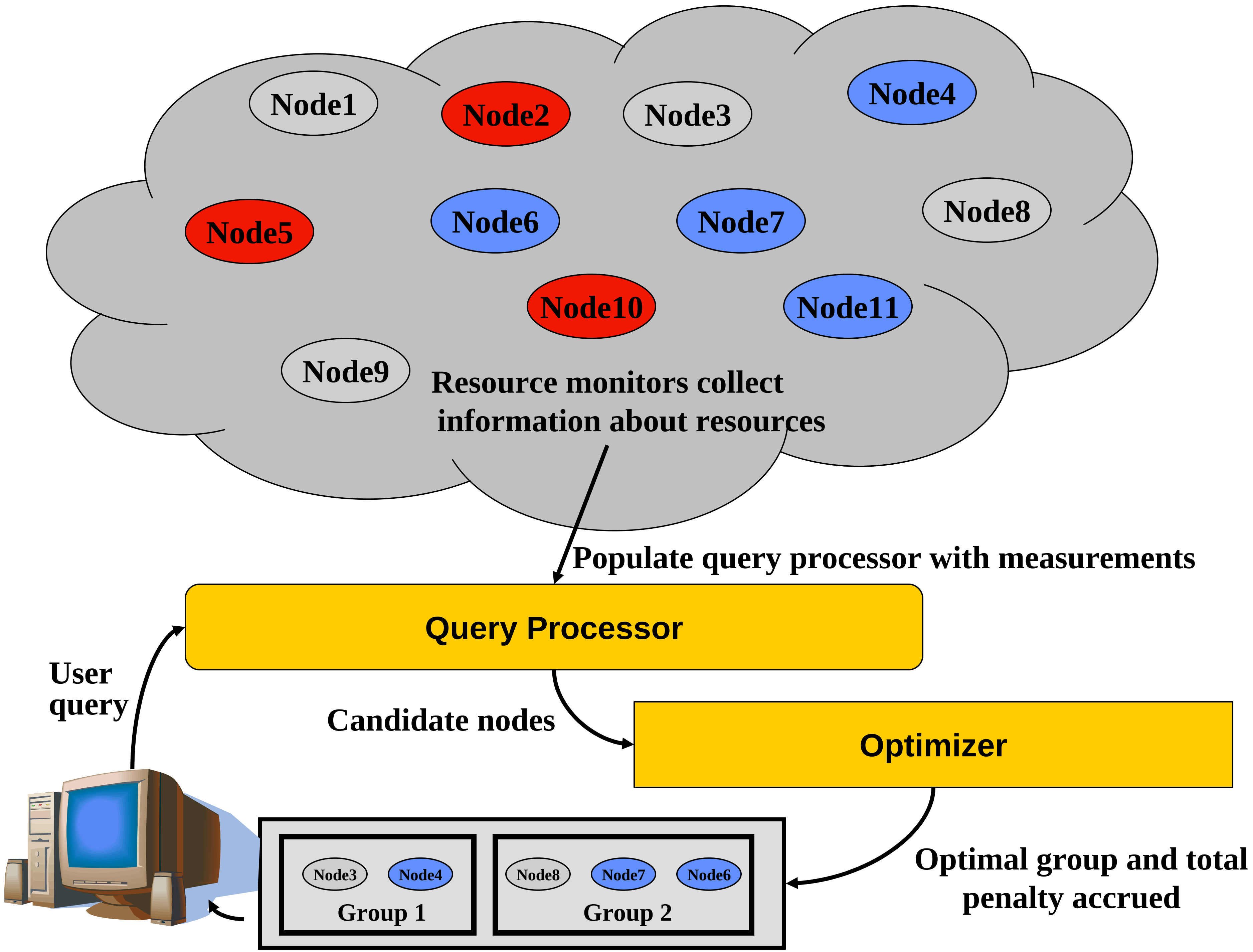
- **Deployment platforms are heterogeneous**
  - **rapidly-changing** attributes
    - **per-node** spare CPU, memory, disk space
    - **inter-node** latency, available bandwidth, loss rate
  - **slowly-changing** attributes
    - due to federation or incremental deployment
    - hardware arch., OS, software installed, admin. policies, ...
- **At deployment time, only a subset of nodes will meet the application's needs**
- **Goal: pick subset of nodes to run on that meet the application's requirements**
  - integrated *resource discovery* and *service placement*

Example query

```
Group NA
   NumMachines 16
   Required Load [0, 2]
   Preferred Load [0, 1], penalty 90
   Required FreeDisk [500, MAX]
   Preferred FreeDisk [1000, MAX], penalty 90
   Required OS [``Linux'']
   Required AllPairs Latency [0, 20]
   Preferred AllPairs Latency [0, 10], penalty 90
   Required AllPairs BW [0.5, MAX]
   Preferred AllPairs BW [1, MAX], penalty 2
   Required Location [``NorthAmerica'', 0, 50]
```

```
Group Europe
   NumMachines 16
   Required Load [0, 2]
   Preferred Load [0, 1], penalty 90
   Required FreeDisk [500, MAX]
   Preferred FreeDisk [1000, MAX], penalty 90
   Required OS [``Linux'']
   Required AllPairs Latency [0, 20]
   Preferred AllPairs Latency [0, 10], penalty 90
   Required AllPairs BW [0.5, MAX]
   Preferred AllPairs BW [1, MAX], penalty 2
   Required Location [``Europe'', 0, 50]
```

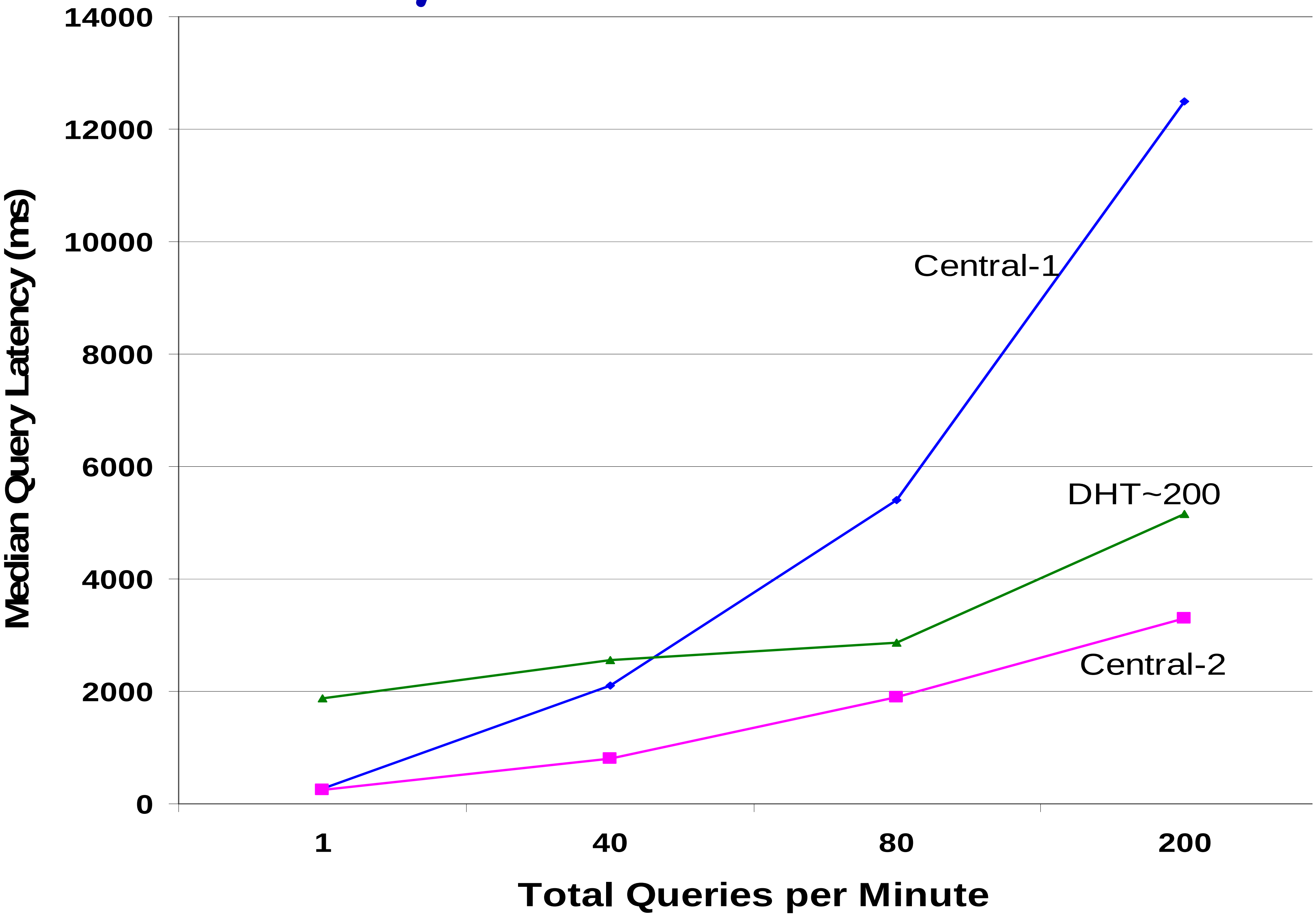
```
InterGroup
    Required OnePair BW NA Europe [3, MAX]
    Preferred OnePair BW NA Europe [5, MAX], penalty 2
```


Node1
Node2
Node3
Node4
Node5
Node6
Node7
Node8
Node9
Node10
Node11
Resource monitors collect information about resources
Populate query processor with measurements
Query Processor
User query
Candidate nodes
Optimizer
Node3
Node4
Group 1
Node8
Node7
Node6
Group 2
Optimal group and total penalty accrued

PlanetLab deployment

- Has been running continuously on 200+ PlanetLab nodes for about six months
- Extensible set of measurements sent every two minutes
  - **Ganglia** host measurements
  - **Trumpet** end-to-end host tests
  - **slicestat** information via CoTop
  - **Vivaldi** network coordinates
- Query processor implemented on top of Bamboo
- Two ways to issue queries
  - web page
  - point command-line client at any SWORD node

Latency vs. workload rate



1. Centralized vs. P2P

- "Infrastructure" distributed testbeds (like PlanetLab) tend to be "small"
  - 100s-1000s; not 10,000s-100,000s
- As a result, centralized solutions may provide sufficient performance (and lower implementation complexity)
- Design suggestion: evaluate centralized solution before embarking on P2P implementation
  - performance for expected workload
  - availability and disaster tolerance requirements
  - bandwidth requirements
  - implementation effort, given desired features
  - debugging effort

2. Simulation *vs.* emulation *vs.* PlanetLab

- **I have an idea for a new distributed architecture for**
  - Google
  - Akamai
  - Kazaa
  - a vigilante anti-spam screensaver network
  - ...
- **How do I evaluate it?**
  - how integrate PlanetLab into evaluation strategy?

2. Simulation vs. emulation vs. PlanetLab

	Property	Fast network simulator	Emulated nodes & net	PlanetLab
system	Scale			
	Network topo. and link char.			
	Node effects			
stimulus	Workload			
	Operator actions			
	Faults			
meas.	Reproducibility			
	Experiment management			

2. Simulation vs. emulation vs. PlanetLab

	Property	Fast network simulator	Emulated nodes & net	PlanetLab
system	Scale	1000s	~1000	~500
	Network topo. and link char.	Flexible, latency only	Flexible, all effects	Hard-wired, all effects
	Node effects	No	Yes	Yes
stimulus	Workload			
	Operator actions			
	Faults			
meas.	Reproducibility			
	Experiment management			

2. Simulation vs. emulation vs. PlanetLab

	Property	Fast network simulator	Emulated nodes & net	PlanetLab
system	Scale	1000s	~1000	~500
	Network topo. and link char.	Flexible, latency only	Flexible, all effects	Hard-wired, all effects
	Node effects	No	Yes	Yes
stimulus	Workload	Flexible	Flexible	Flexible & Realistic
	Operator actions	No	Flexible	Realistic
	Faults	Net only	Flexible	Realistic
meas.	Reproducibility			
	Experiment management			

2. Simulation vs. emulation vs. PlanetLab

	Property	Fast network simulator	Emulated nodes & net	PlanetLab
system	Scale	1000s	~1000	~500
	Network topo. and link char.	Flexible, latency only	Flexible, all effects	Hard-wired, all effects
	Node effects	No	Yes	Yes
stimulus	Workload	Flexible	Flexible	Flexible & Realistic
	Operator actions	No	Flexible	Realistic
	Faults	Net only	Flexible	Realistic
meas.	Reproducibility	High	Medium	Low
	Experiment management	Easy	Medium	Hard

2. Simulation *vs.* emulation *vs.* PlanetLab

- PlanetLab deployment *complements* rather than *replaces* traditional evaluation approaches

- Design suggestion
  - deploy your system on PlanetLab
  - use traces of workload, contention, and failures from PlanetLab to drive simulation or emulation
    - best of both worlds

Conclusion

- Integrated resource discovery and placement for services, computations, and experiments
  - pick subset of machines that meet your app's requirements
- Query semantics specialized for resource discovery
  - topology of interconnected groups
  - penalty functions
- Distributed (DHT) and centralized implementations
- Small centralized cluster superior to DHT-based
  - but DHT-based provided reasonable performance and high availability
- PlanetLab's *realism* complements flexibility and reproducibility of traditional evaluation apprchs.

Please use SWORD!

Distributed Resource Discovery on PlanetLab with SWORD

http://www.swordrd.org/

David Oppenheimer, Jeannie Albrecht
David Patterson, Amin Vahdat
UC Berkeley / UC San Diego

First Workshop on Real, Large Distributed Systems
December 5, 2004