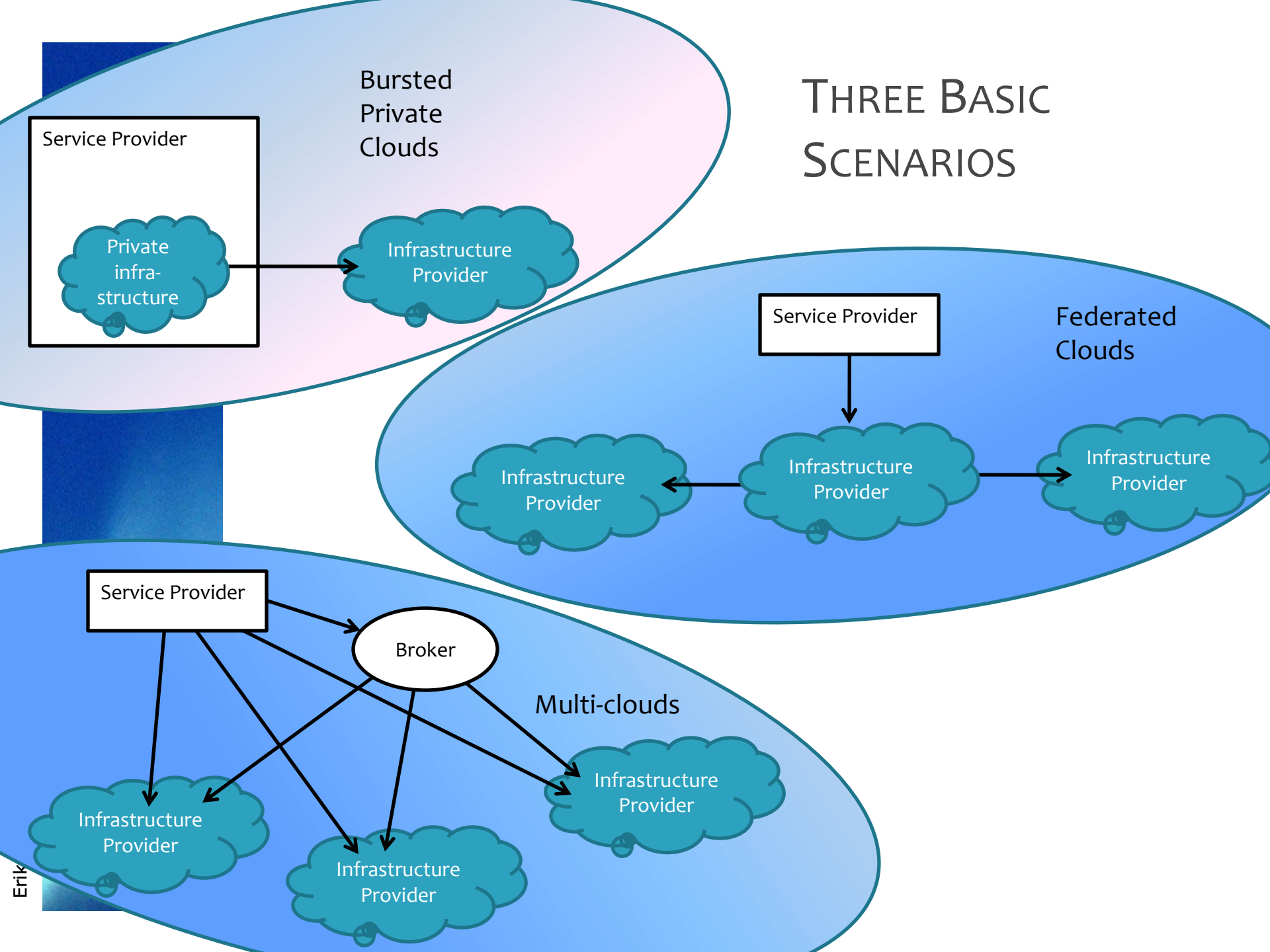
Self-Management Challenges for Multi-Cloud Architectures

Erik Elmroth
Department of Computing Science & HPC2N
Umeå University

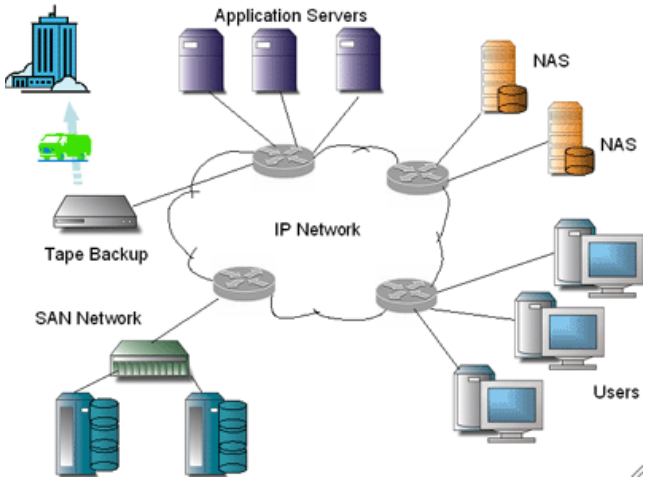
www.cloudresearch.se

Three Basic Scenarios



Cloud Resource Management

For whom?
- Service providers
- Infrastructure providers



What?
- Low and high level management
- Compute + storage + network
- Autonomic systems

How?
- Single abstraction – multiple scenarios
- General tools for key functionality
- Flexibility in deployment and configuration

Erik Elmroth elmroth@cs.umu.se

Critical performance requirements - to be cost-efficiently met

Extremely rapid growth (from global scale)

- YouTube (16 months) 100 mil/movies per day, 20 mil. unique users per month
- AppStore (19 months): Over 100000 Iphone programs, over 3 billion downloads

Regular/planned peaks

- Banks, tax filing
- Market campaign effects

Unexpected peaks

- News related video streaming
- Stock trading peaks at financial crises

Regional aspects in usage patters

- Regional concerns (news, events, etc)
- Time-dependent usage-patterns

Critical performance requirements - to be cost-efficiently met

Extremely rapid growth (from global scale)

- YouTube (16 months) 100 mil/movies per day, 20 mil. unique users per month
- AppStore (19 months): Over 100000 Iphone programs, over 3 billion downloads

Regular/planned peaks

- Banks, tax filing
- Market campaign effects

Unexpected peaks

- New related video streaming
- Stock trading peaks at financial crises

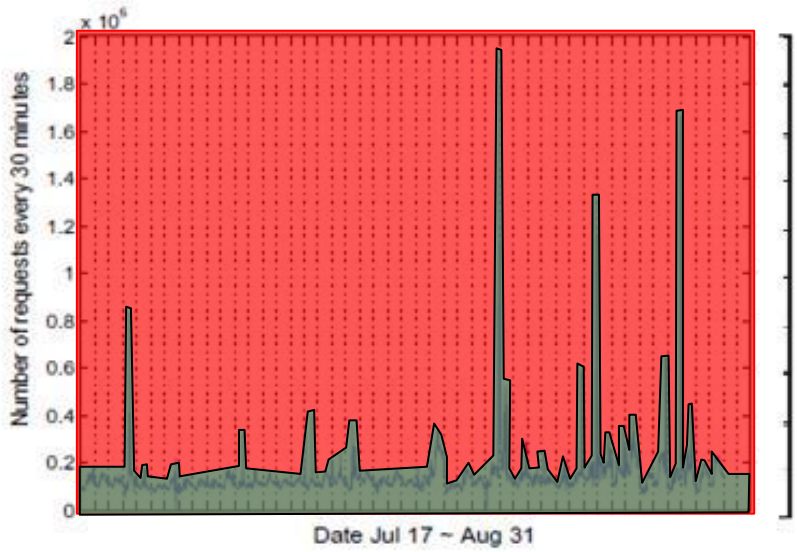
Regional aspects in usage patters

- Regional concerns (news, events, etc)
- Time-dependent usage-patterns

Low level management example

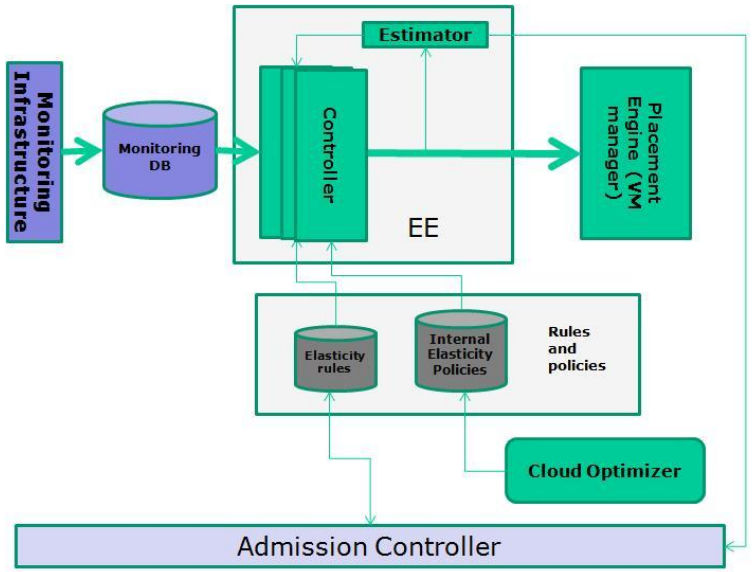
Elasticity control

- Manage peaks & lows
- Ability to meet SLAs
- Resource consumption
- Adaptivity, Scalablility, Simplicity

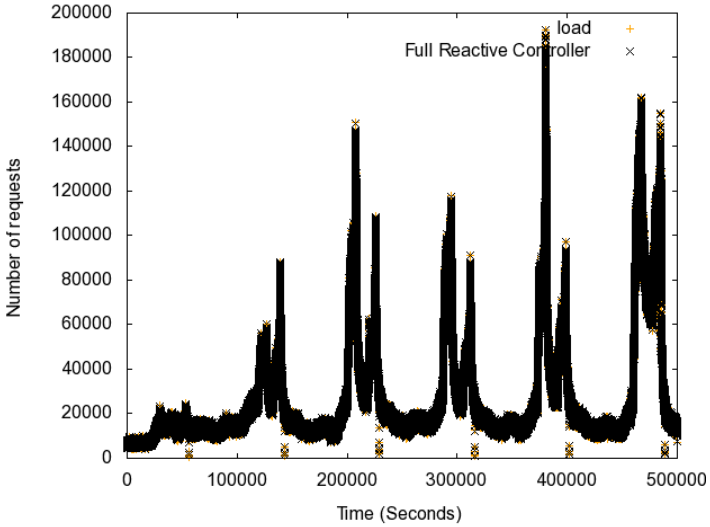


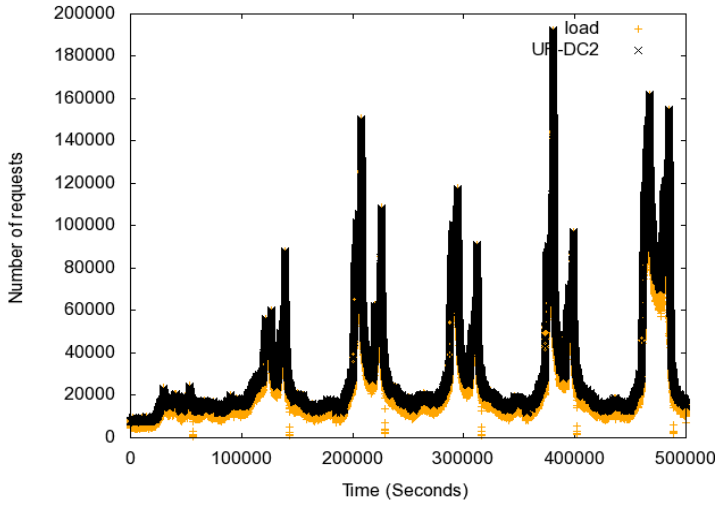
Simple closed loop P-controller

- Error signal multiplied by constant
- Adapt to load dynamics:
  - Adaptive gain
  - Adaptive rate

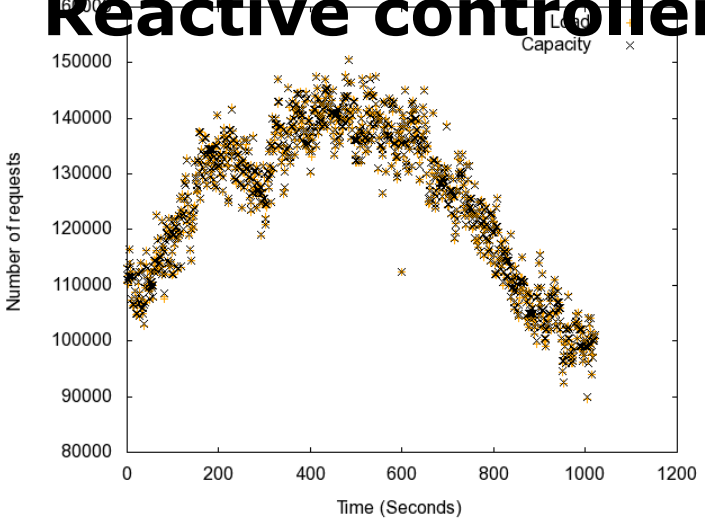


Results (6 days & 17 min close-up)

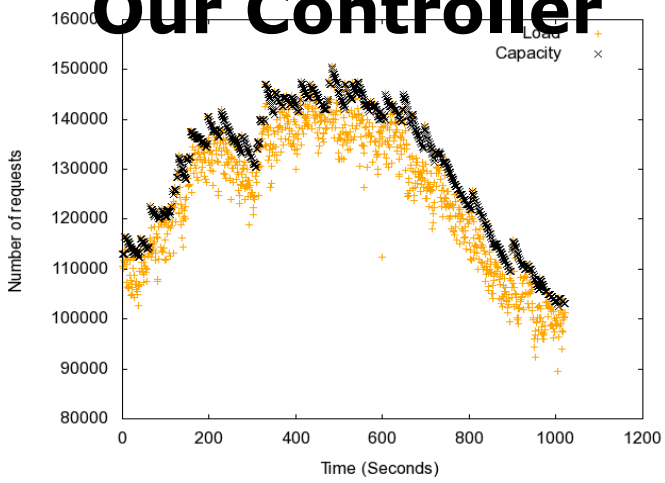




Reactive controller



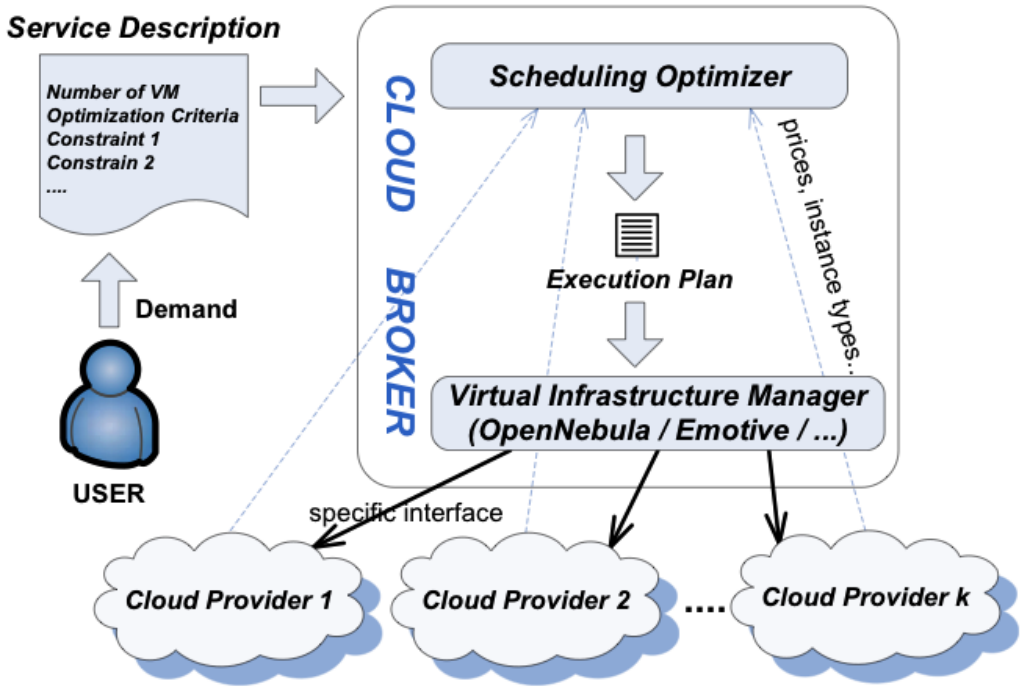
Our Controller



Dynamic Cloud Scheduling

Cost/performance-based: Assume

- changes in prices and available resource types
- cross-site migration feasible
- load balancing constraints



Cloud Scheduling (VM Placement)

Modeling (Cost Goals)

Total cost

Minimize $TIP = P * \sum_{j=1}^{l} \sum_{k=1}^{m} P_{jk}(\sum_{i=1}^{n} x_{ijk})$

Subject to

Capacity constraints

$$TIC = E + \sum_{j=1}^{l} C_j (\sum_{i=1}^{n} \sum_{k=1}^{m} x_{ijk}) - \sum_{i=1}^{n} (\beta_i * \Delta_i) \leq Threshold \quad (1)$$

$\forall i \in [1..n]$:

$$\sum_{j=1}^{l} \sum_{k=1}^{m} x_{ijk} = 1 \quad (2)$$

$\forall k \in [1..m]$:

Load balance constraints

$$LOC_{min} \leq (\sum_{i=1}^{n} \sum_{j=1}^{l} x_{ijk})/n \leq LOC_{max} \quad (3)$$

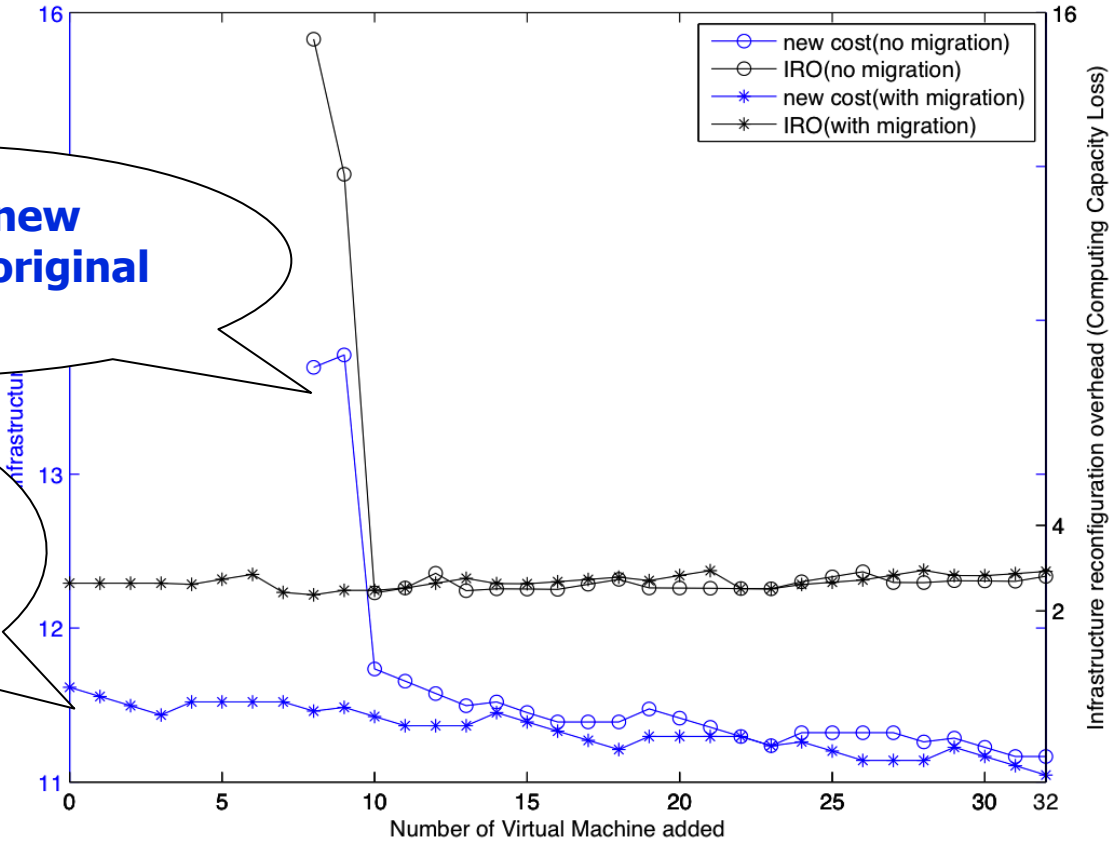
Vertical/horizontal elasticity trade-off scenario

Starting point:

- 32 VMs, 5 VM types, 3 clouds, load balance constraints

Objective:

- Double the performance using 0–32 additional VMs



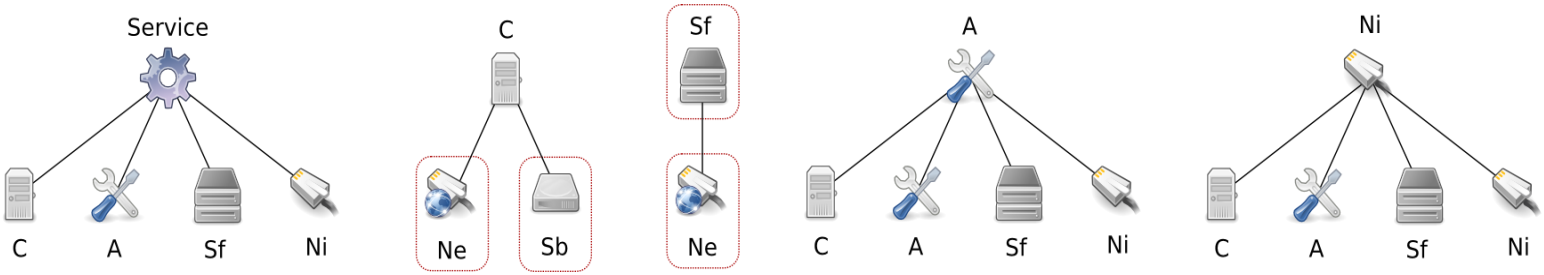
Cost incl. 0-32 new VMs assuming original VMs untouched

Cost incl. 0-32 new VMs assuming move of running VMs

Service Structure

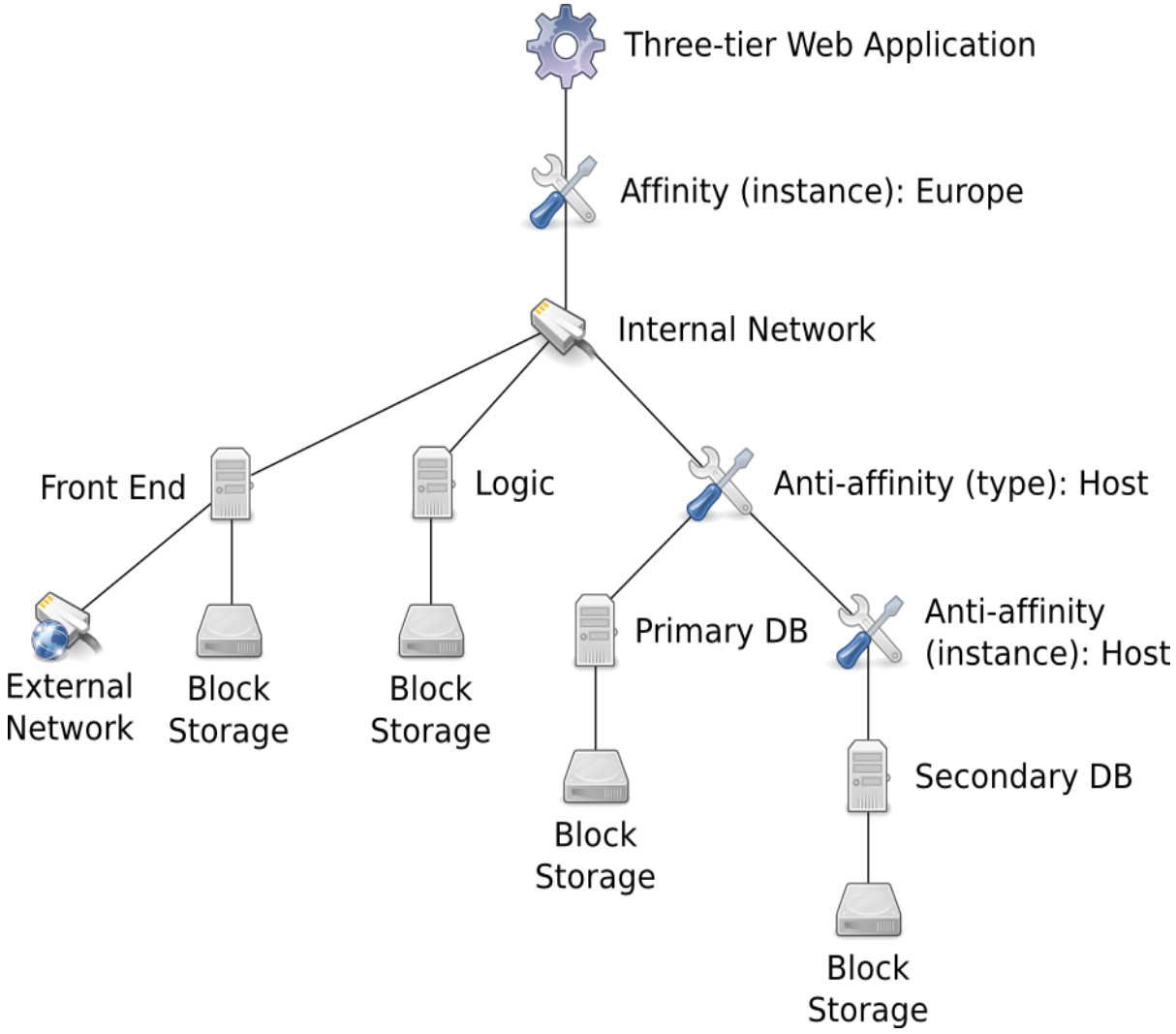
- Parts of services may have different *Affinities* affecting placement
  - Affinity may be geographical or relate to other components in the service
  - *Anti-affinity* is an unwanted relation and follows the same patterns as Affinity
  - The union of these are called *AA-constraints*
- Service definition model also includes computational resources, networks, and storage nodes in our representation
- The representation is a Directed Acyclic Graph (DAG), as some children may have more than one parent

Relations and Types

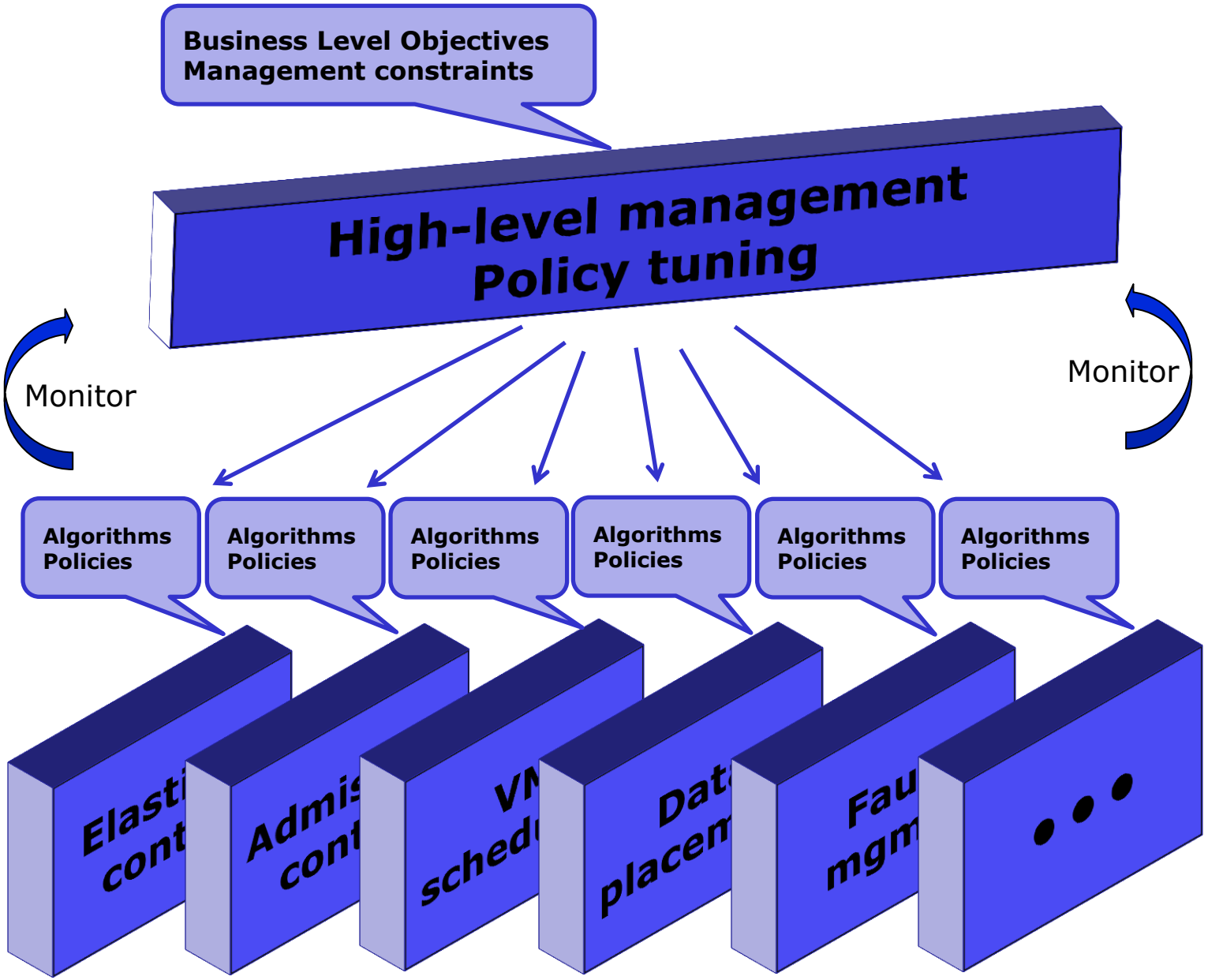


Node Type	Abbr.	Description
Service Root		Common ancestor for all service components.
Compute Resource	C	Compute resource
AA-constraint	A	Constraints to determine placement
Block Storage	Sb	A mountable data storage for a Compute resource. Cf. Amazon EBS.
File Storage	Sf	Data storage which may be accessed by multiple Compute resources simultaneously. Cf. Amazon S3
Internal Network	Ni	Internal network connection
External Network	Ne	External network connection

Service Example

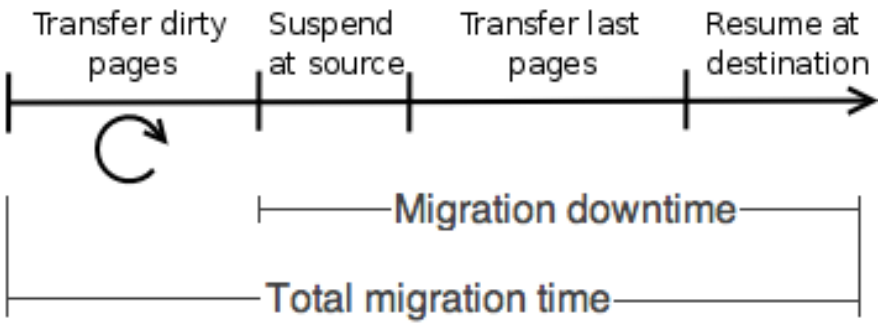


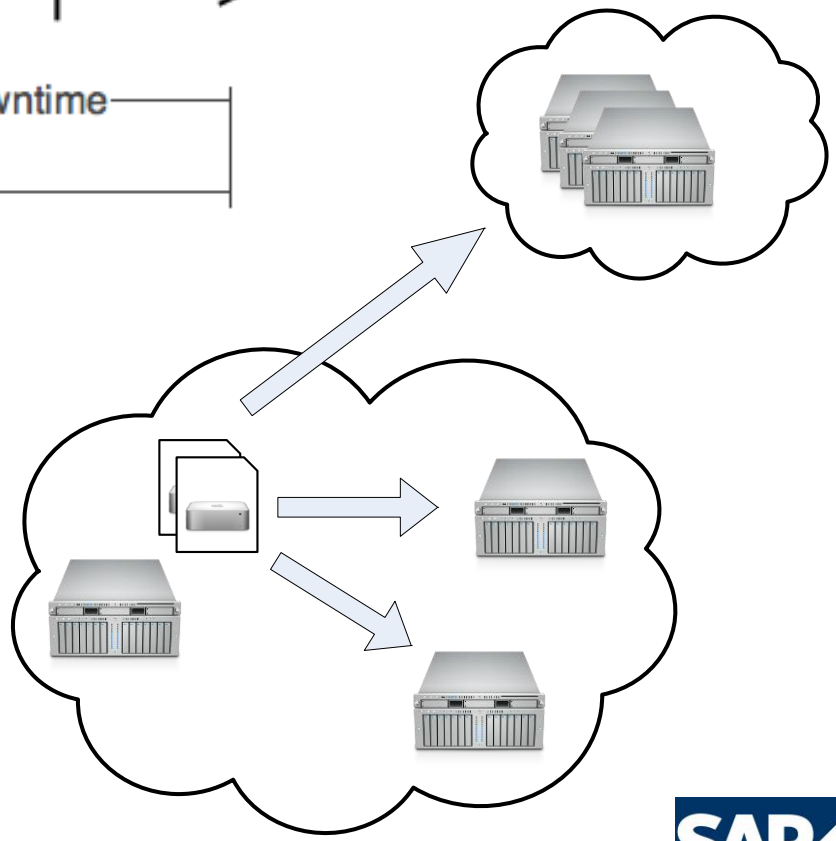
Holistic cloud management



Erik Elmroth elmroth@cs.umu.se

Live VM migration (without service interruption)





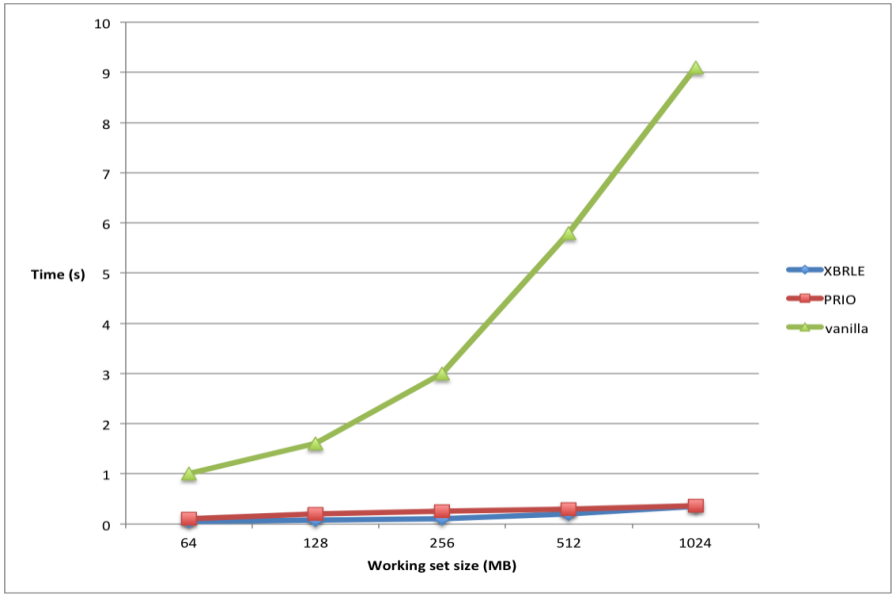
Reduce:
Service downtime
- **Caching & compression**

Total migration time (resource consumption)
- **Page priorities**

Illustrative performance results

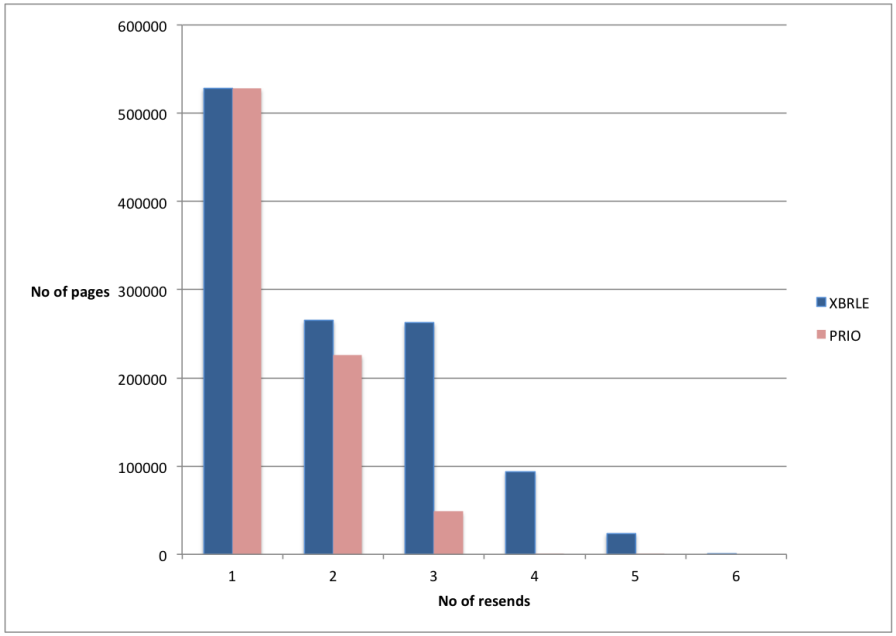
Migration downtime

- Standard algorithm exceeds 3 s at approx 256 MB WS size



Page re-sends

- Approx 400k fewer pages re-sent using dynamic page transfer reordering



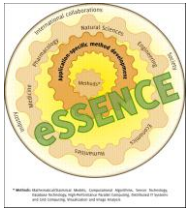
Tests performed using LMbench

Large-scale collaborations

EU FP7 IP. Introduced federated clouds. EUs first major cloud project.

EU FP7 IP. Optimized cloud services over complete lifecycle. Non-functional aspects.

EU FP7 IP. Pioneering federated storage clouds. Raised level of abstraction. Media- and telecom applications.



Governments strategic efforts. Methods and software for eScience applications.

UMIT
Research Lab

Umeå initiative for innovation and industry benefits within simulation, visualisation, computation and infrastructure

Key partners: IBM Haifa Research Labs, SAP Research, ATOS Orgin, Universidad Complutense de Madrid, Leeds University, Barcelona Supercomputer Center, Telefonica I+D, and British Telecom

Senior researchers

Erik Elmroth, Professor

Francisco Hernandez, Assistant Professor

Johan Tordsson, Assistant Professor

Lei Xu, Post Doc

P-O Östberg Post Doc

Project admin.

Lennart Edblom, Senior lecturer

PhD students

Ahmed Ali-Eldin

Daniel Henriksson

Ewnetu Bayuh Lakew

Wubin Li

Mina Sedaghat

Petter Svärd

Systems developers/systems experts

Tomas Forsman, Systems expert

Peter Gardfjäll, Systems Developer

Sebastian Gröhn, Research assistant

Anders Häggström, Research assistant

Lars Larsson, Systems Developer

cloudresearch.se

Erik Elmroth elmroth@cs.umu.se