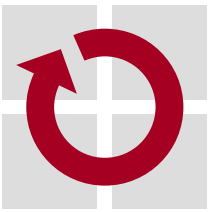
Decentralised Diffusion-based Quota Management

Rüdiger Kapitza, **Meik Felser**,
Hans P. Reiser, Franz J. Hauck

AspectIX Research Group
University Erlangen-Nuremberg



Motivation

- Grid infrastructures provide resources
- Resources are shared among multiple parties

- Resource access has to be restricted
- Otherwise a user could allocate all resources
  - Accident
  - Attack
  - Unawareness

- Solution: Quotas

Motivation

- Existing implementation of quota-based approaches

1. Quota manager issues fractions of quota to the clients
2. Client can acquire resources up to the quota

- Central quota manager is the bottleneck
  - Large-scale grids
  - Frequent quota allocations
  - Single point of failure

Motivation

→ decentralised quota management system

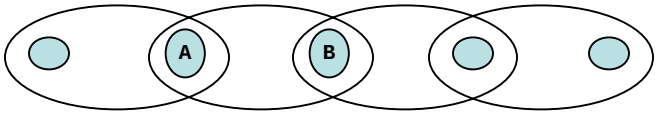
- No central quota service
  - Scalability
- Distributed quota information
  - Fault-tolerance

Outline

- Motivation
- Diffusive Balancing
  - Load Balancing
  - Quota Balancing
  - Basic Protocol
- Extension for Fault-Tolerance
- Handling of Non-Refundable Quotas
- Conclusion and Ongoing Work

Diffusive Load Balancing

- Load balancing for distributed multiprocessor systems (Cybenko, 1998)
- Nodes are organised in small overlapping groups
- Load information is exchanged
- If there is an imbalance, load is migrated
  - **A** , **B** represent nodes
  - w_A , w_B denote the current load of nodes **A** and **B**
  - If $w_A < w_B$ then move load $(w_A - w_B) / 2$ from **B** to **A** and vice versa if $w_A > w_B$



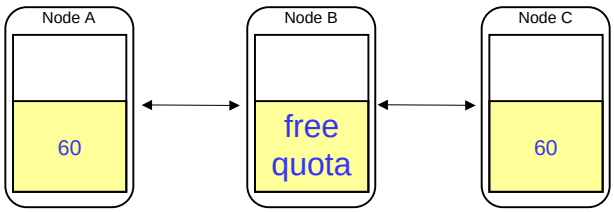
Diffusive Quota Balancing: Basic Idea

- Quota management:
  - Do NOT balance load
  - Balance the free quota information

→ Quota information is distributed
  - → Fast resource allocation
  - → Resource demand may be fulfilled locally

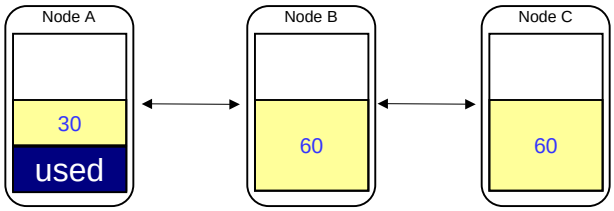
Diffusive Quota Balancing: Basic Idea

- Free quota is distributed and balanced among all nodes
  - Initial distribution of free quota



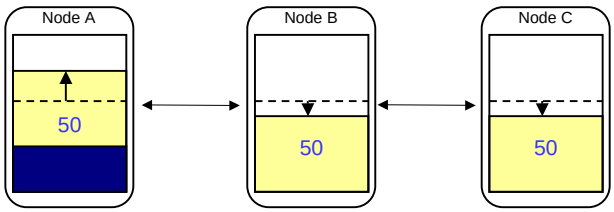
Diffusive Quota Balancing: Basic Idea

- Free quota is distributed and balanced among all nodes
  - Application has been started
  - Resource demand was fulfilled immediately



Diffusive Quota Balancing: Basic Idea

- Free quota is distributed and balanced among all nodes
  - System reached quota equilibrium again
  - Free quota is balanced again

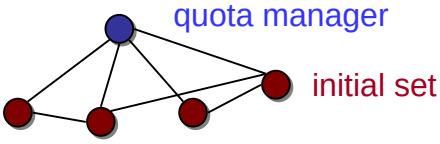


Basic Quota Balancing Protocol

- Quota Manager:
  - Initial node
  - Injection point of initial quota
  - Not a central component

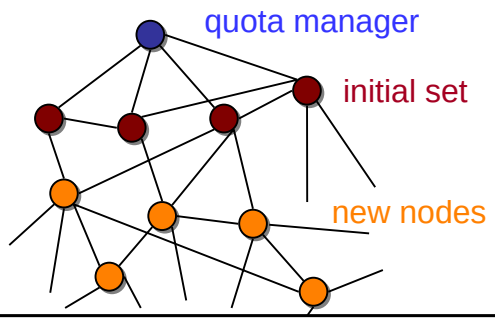
Basic Quota Balancing Protocol

- Quota manager
- Initial set:
  - Connected to the quota manager
  - Connected to some other nodes of the initial set
  - Forming an unstructured network



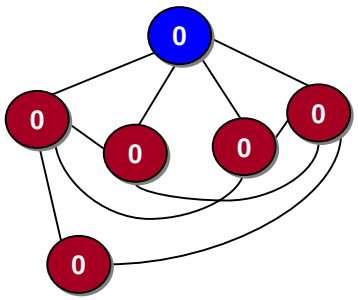
Basic Quota Balancing Protocol

- Quota manager
- Initial set
- Additional nodes:
  - connect to a random subset of the nodes



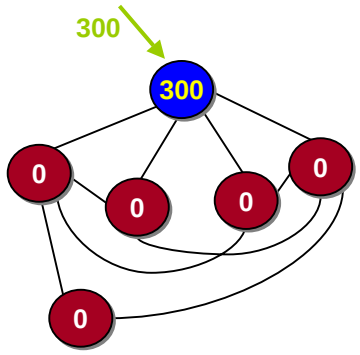
Basic Quota Balancing Protocol

- Each node stores its local free quota



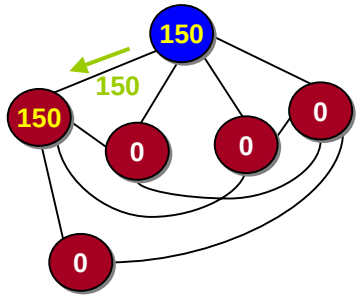
Basic Quota Balancing Protocol

- Each node stores its local free quota
- Global quota is injected into the quota manager



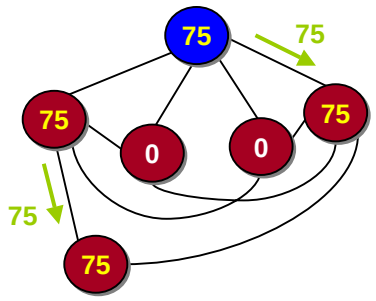
Basic Quota Balancing Protocol

- If the local free quota changes
  - The nodes exchange quota information and
  - Balance quota differences



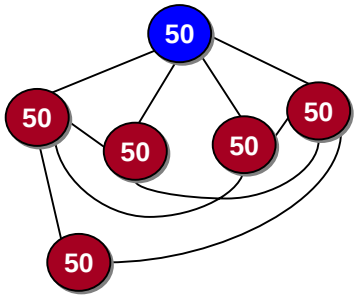
Basic Quota Balancing Protocol

- If the local free quota changes
  - The nodes exchange quota information and
  - Balance quota differences
- This continues …



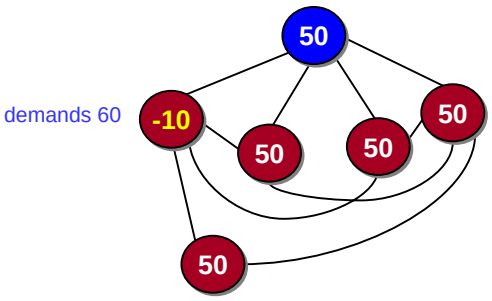
Basic Quota Balancing Protocol

- If the local free quota changes
  - The nodes exchange quota information and
  - Balance quota differences
- This continues until the equilibrium is reached



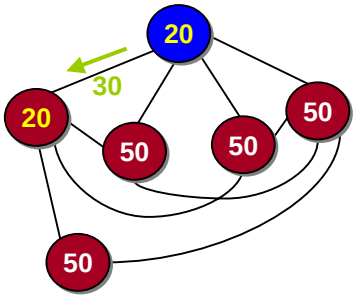
Basic Quota Balancing Protocol

- Resource demands reduce the free local quota
  - This can result in negative free local quota
- The algorithm starts balancing the free quota



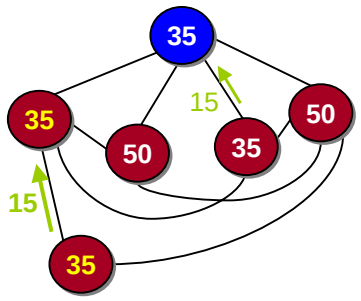
Basic Quota Balancing Protocol

- The algorithm is balancing the free quota



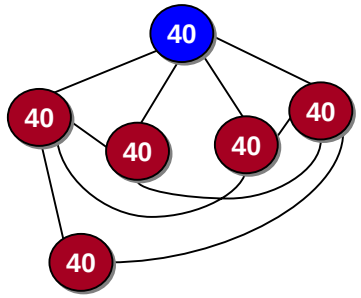
Basic Quota Balancing Protocol

- Eventually …



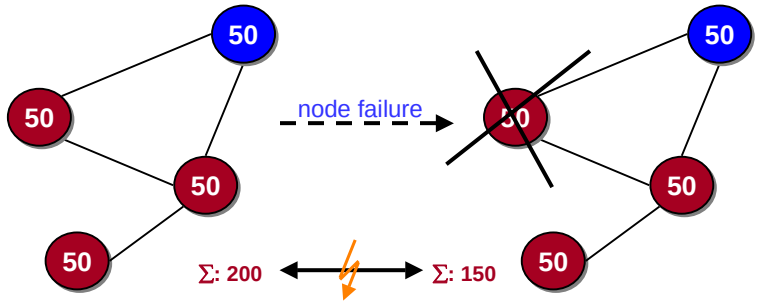
Basic Quota Balancing Protocol

- Eventually the system is in equilibrium again



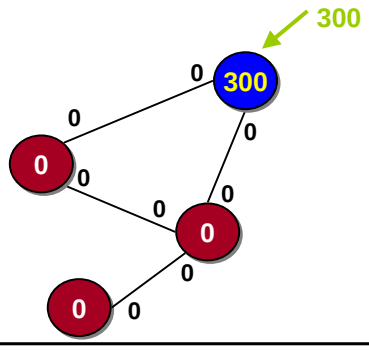
Basic Quota Balancing Protocol

- Node failures lead to the loss of quota information



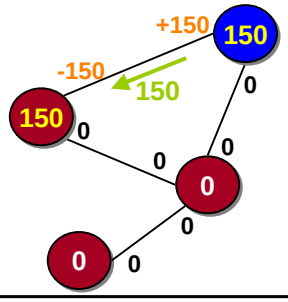
Extension for Fault-Tolerance

- Spontaneous node failures should be tolerated
  - Without losing quota information
- Solution: link counter



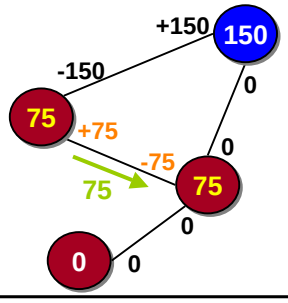
Extension for Fault-Tolerance

- Spontaneous node failures should be tolerated
  - Without losing quota information
- Solution: link counter
  - Outgoing quota units are added to the link counter
  - Incoming quota units are subtracted from the link counter



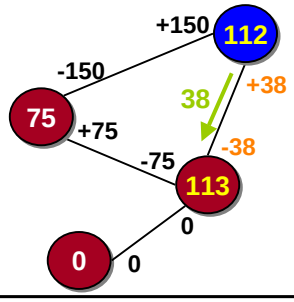
Extension for Fault-Tolerance

- Spontaneous node failures should be tolerated
  - Without losing quota information
- Solution: link counter
  - Positive values represent outgoing quota units
  - Negative values represent incoming quota units



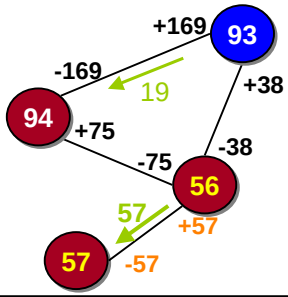
Extension for Fault-Tolerance

- Spontaneous node failures should be tolerated
  - Without losing quota information
- Solution: link counter
  - Positive values represent outgoing quota units
  - Negative values represent incoming quota units



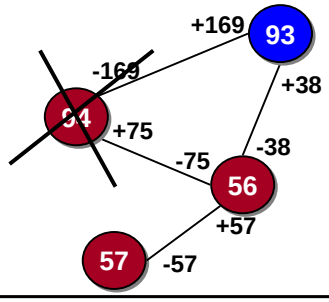
Extension for Fault-Tolerance

- Spontaneous node failures should be tolerated
  - Without losing quota information
- Solution: link counter
  - Positive values represent outgoing quota units
  - Negative values represent incoming quota units



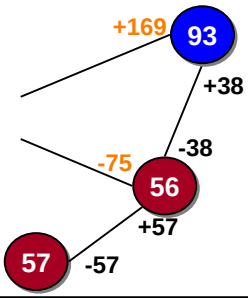
Extension for Fault-Tolerance

- Spontaneous node failures should be tolerated
  - Without losing quota information
- Solution: link counter
- Node failure



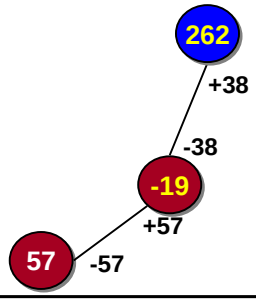
Extension for Fault-Tolerance

- Spontaneous node failures should be tolerated
  - Without losing quota information
- Solution: link counter used for quota reconstruction!
  - On node failure link counter is added to local quota.



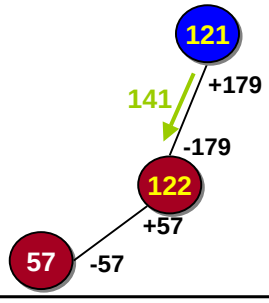
Extension for Fault-Tolerance

- Spontaneous node failures should be tolerated
  - Without losing quota information
- Solution: link counter used for quota reconstruction!
  - On node failure link counter is added to local quota.
  - This may result in negative local quota



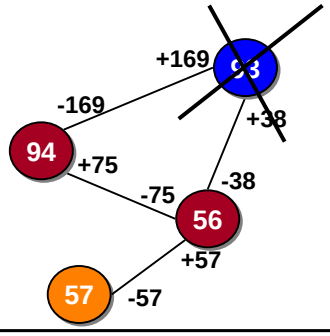
Extension for Fault-Tolerance

- Spontaneous node failures should be tolerated
  - Without losing quota information
- Solution: link counter used for quota reconstruction!
  - On node failure link counter is added to local quota.
  - This may result in negative local quota



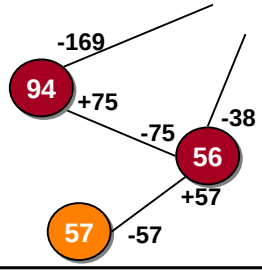
Failure of the Quota Manager

- Quota manager fails
  - Initial node set does not evaluate link counter
  - Quota manager is restored via passive replication
- System can run without the quota manager



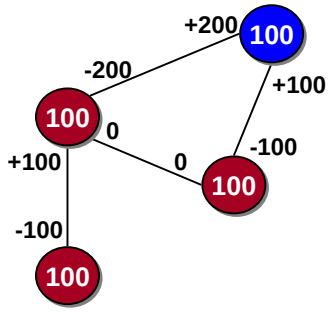
Failure of the Quota Manager

- Quota manager fails
  - Initial node set does not evaluate link counter
  - Quota manager is restored via passive replication
- System can run without the quota manager



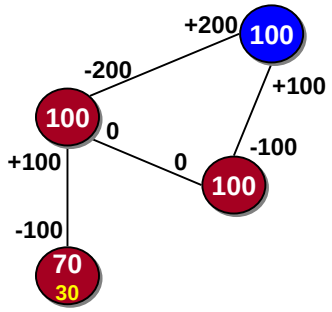
Handling of Non-Refundable Quotas

- Non-Refundable Quotas
  - Quotas for resources such as transfer volume or CPU cycles



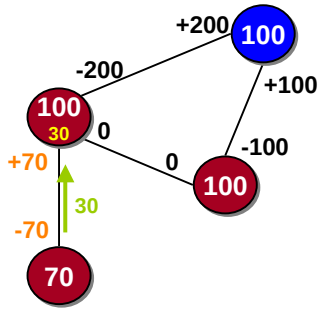
Handling of Non-Refundable Quotas

- Non-Refundable Quotas
  - Quotas for resources such as transfer volume or CPU cycles
  - Can be consumed
    - → link counters won't help in case of node failures



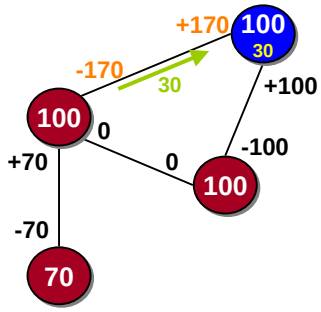
Handling of Non-Refundable Quotas

- Non-Refundable Quotas
  - Quotas for resources such as transfer volume or CPU cycles
  - Can be consumed → link counters won't help
  - Solution: Propagation of consumed quota to quota manager

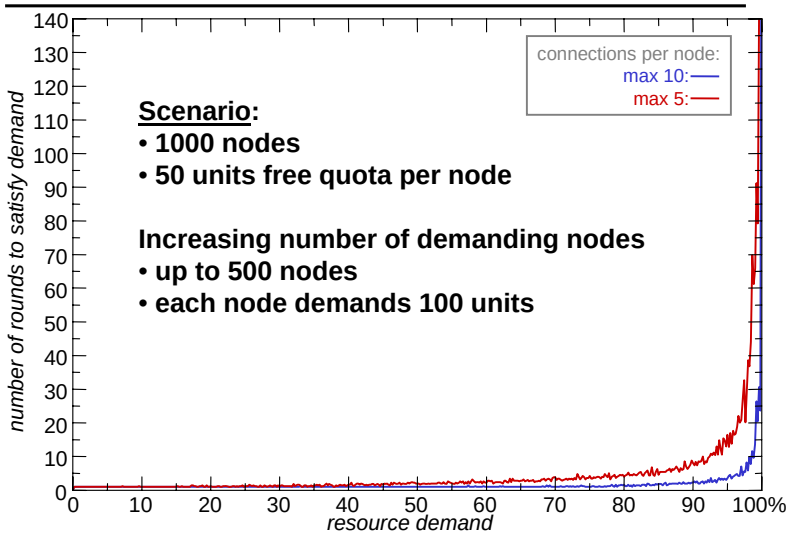


Handling of Non-Refundable Quotas

- Non-Refundable Quotas
  - Quotas for resources such as transfer volume or CPU cycles
  - Can be consumed → link counters won't help
  - Solution: Propagation of consumed quota to quota manager



Evaluation



Ongoing Work

- Integration into the AspectIX Middleware
  - EDAS - Environment for Decentralized Adaptive Services
  - www.aspectix.de
- Acquiring resource information from applications
  - Java Management eXtension (JMX)
- Communication service platform
  - JXTA Peer-to-Peer infrastructure
    - Peer group for each project
    - Every peer provides a JXTA service implementing the protocol
    - Communication via JXTA sockets

Conclusion

- Quota management protocol
  - Distributed quota information
  - Local knowledge of available quota fraction
- Properties
  - Decentralised
  - Scalable
  - Fault-tolerant
- Supports
  - Refundable and
  - Non-refundable resources

Thank you

for your attention!

Any Questions?

Evaluation

- Rounds to reach equilibrium