

Gridstat Research Milli-Updates

- 1) Actuator RPC w/ QoS & Safety
- 2) Global and Hierarchical Mode Change Mechanisms & Management

Prof. Dave Bakken

**School of Electrical Engineering and Computer Science
Washington State University
Pullman, Washington USA**

**51st IFIP 10.4 WG Meeting, Guadeloupe
January 14, 2007**



<http://gridstat.net>

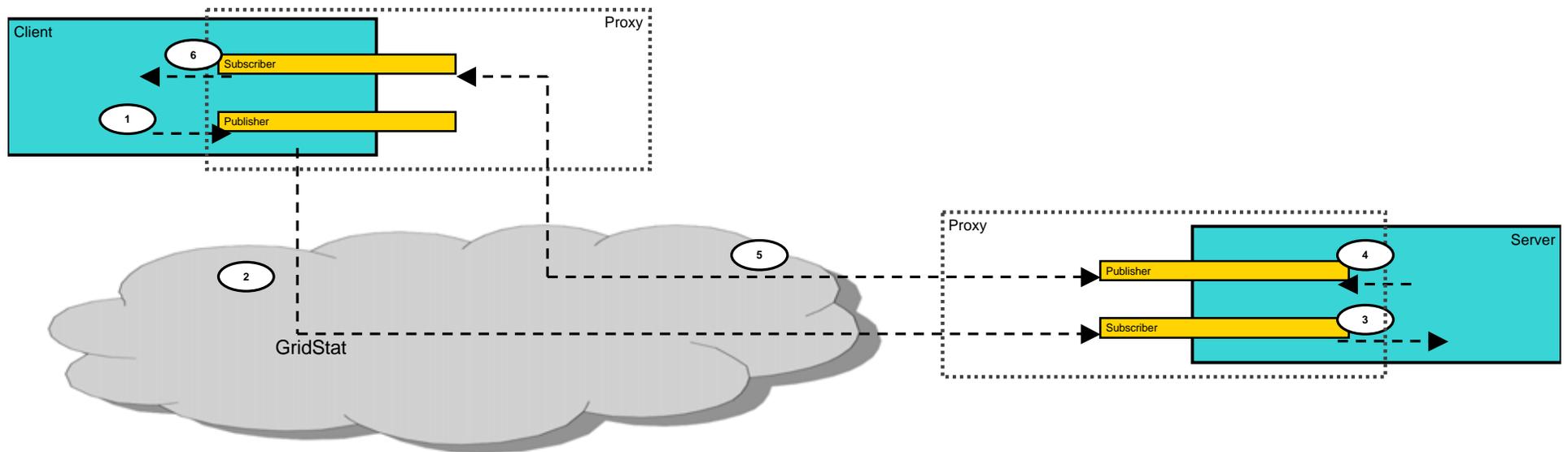
1) Actuator RPC w/ QoS & Safety

- Student: Erlend Viddal
- Background: GridStat version 3 only supports a publish-subscriber style of interaction
- Extend this to support a client-server style of interaction
 - Design and implement a wide-area RPC mechanism for grid-control, *Ratatoskr*
 - Primarily intended for control messages between control-centers and actuators in substations
 - Run on top of GridStat's one-way pub/sub connections for timely delivery and fault-tolerance
 - Provide some customization of failure semantics

Research Questions

- How to exploit QoS and redundancy management in GridStat's push-style data plane to build a round-trip RPC?
- How to reduce inherent RPC call status ambiguity while bounding latency as much as practical?
- What are the tradeoffs of fault tolerance, safety, and security against the call's timeliness?
- How to place pre- and post-conditions on calls in a flexible and programmer-friendly manner

Pre-conditions and Post-Conditions



- Pre-conditions & post-conditions are predicates over GridStat status variables
 - Pre-conditions check for safety: short circuit actuator call & throw exception to client
 - Post-conditions allow additional physical verification (after optional delay) that the actuator call succeeded: generate alert

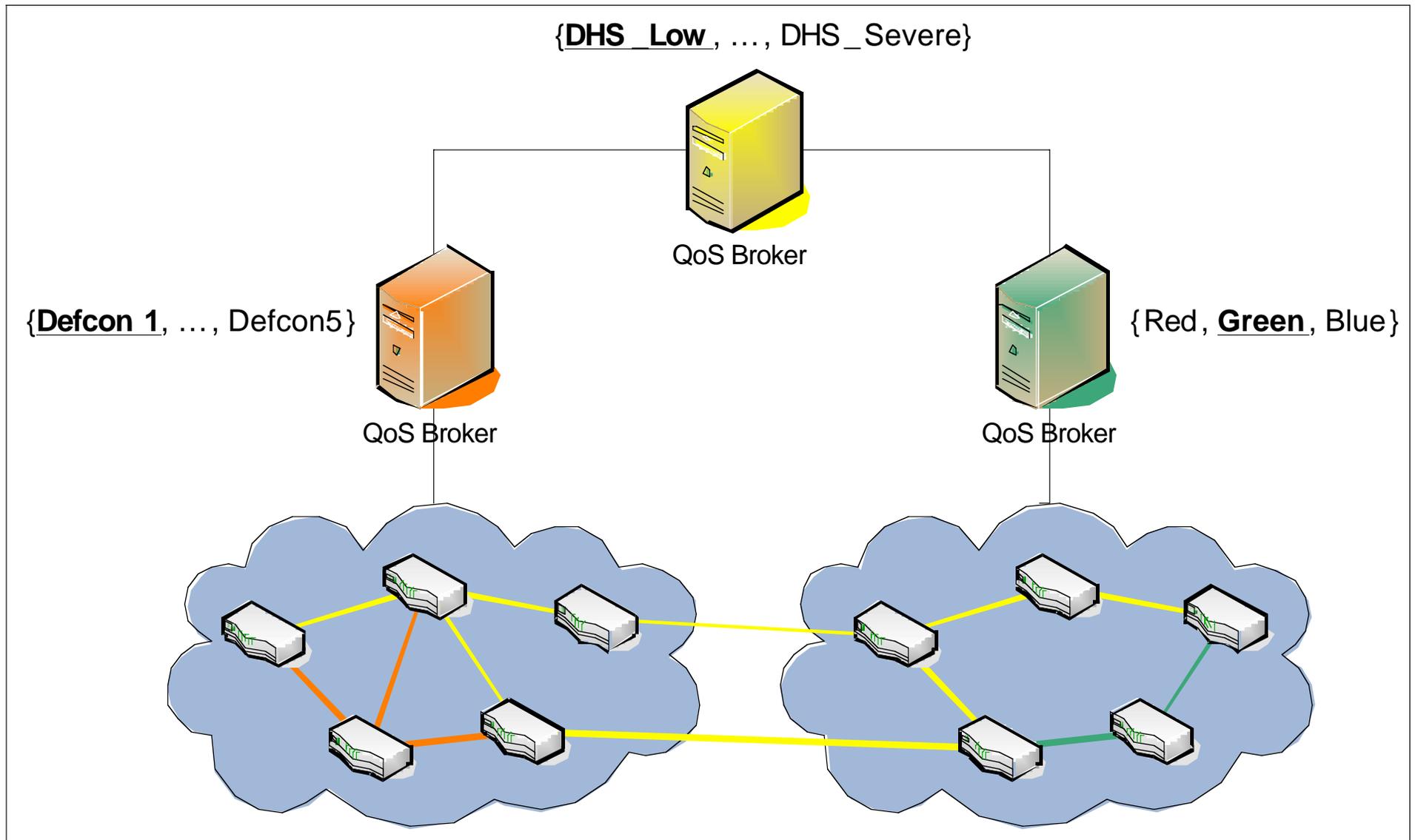
2) Global and Hierarchical Mode Change Mechanisms & Management

- Student: Stian Abelsen
- Background: *modes* in GridStat version 3: a quick way to switch routing tables in the data plane's SRs; limited to one cloud (leaf)
- Extend modes for GridStat version 4 to be
 - *Global*: available up to any level in hierarchy
 - *Hierarchical*: some modes have non-global scope
- Extend SRs to support multiple active routing tables needed for above

Research Questions

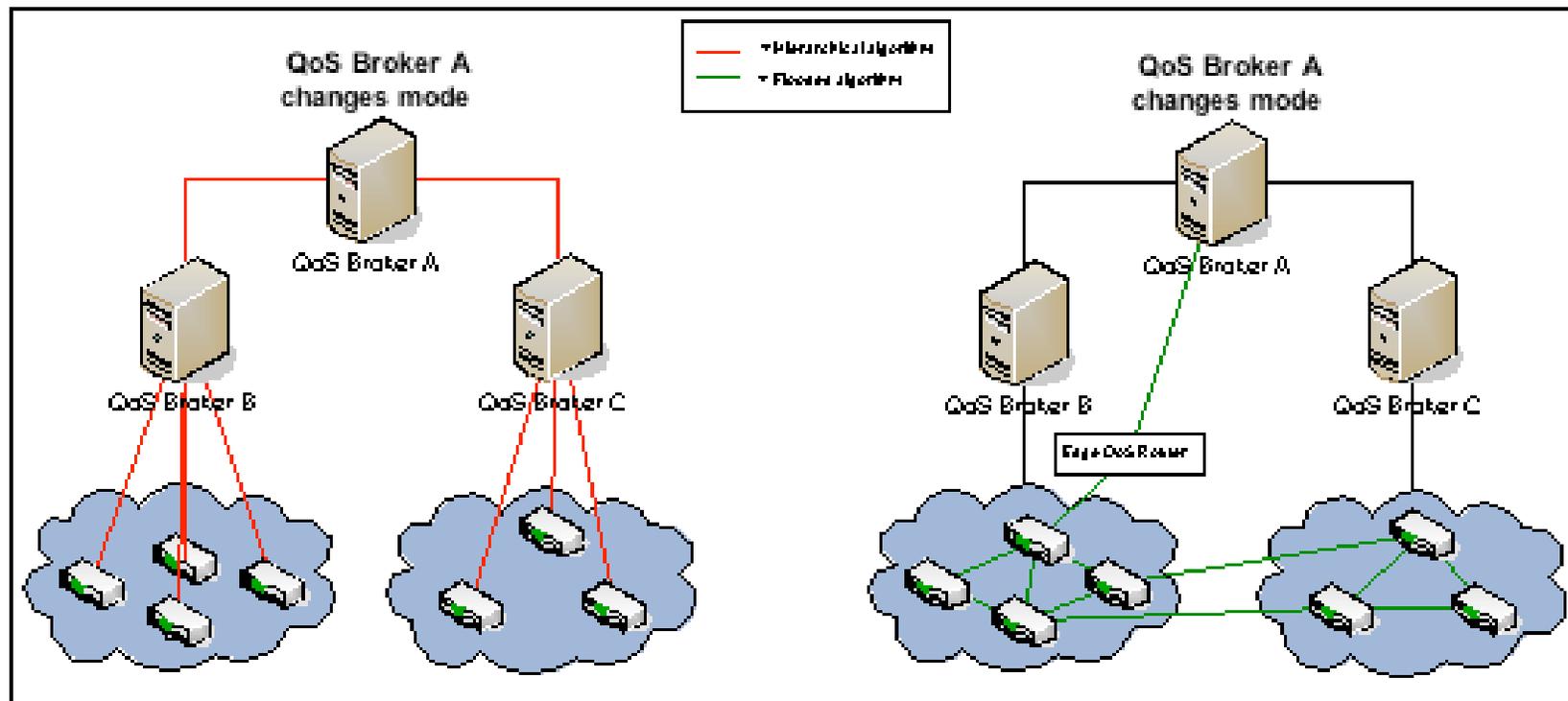
- Can we devise mode change operations to
 - Work over an entire hierarchy?
 - Efficiently take advantage of temporal redundancy inherent with status update flows?
 - Provide consistency “close enough” to heavyweight strong consistency?
- How can mode change mechanisms benefit from GridStat QoS guarantees?
- What are the tradeoffs between different mode change algorithms?
 - Can we tolerate some status update loss?
 - What are the implications of network failures and which ones can we to some degree tolerate?

Global and Hierarchical Modes



Results to Date

- Consistency survey completed; conclusion: heavyweight 2/3-phase commits unnecessary
- Design and implementation of two distributed mode management algorithms with different tradeoffs
 - Hierarchical: better guarantees on mode handoff
 - Flooding w/timestamp: much quicker, uses QoS



Backup Slides (Hah!)

Ratatoskr: an efficient, fault-tolerant RPC for GridStat

Erlend Viddal

WSU



Wikipedia:

In Norse mythology, Ratatoskr (drilling tooth) is a squirrel who runs up and down with messages in the world tree Yggdrasill and spreads gossip. In particular he ferried insults between the eagle at the top of Yggdrasill, and the dragon Níðhöggr beneath its roots.
