To appear in *Supplement of the 2002 International Conference on Dependable Systems and Networks (DSN-2002)*, IEEE/IFIP, Washington, DC, June 23-26, 2002.

# Fault Tolerance Issues in Publish-Subscribe Status Dissemination Middleware for the Electric Power Grid

Ioanna Dionysiou, Kjell Harald Gjermundrød, David Bakken
*Washington State University, Pullman, Washington USA*
{idionysi, kgjermun, bakken}@eecs.wsu.edu

## 1. Introduction

The current data infrastructure of the electric power grid was based on the existence of vertically integrated utilities that controlled the entire power operation within a geographical region. Further, it was designed decades ago by engineers who knew little about the state of the art (back then) in computer networking or distributed systems. Recent deregulation allows a wider set of participants to get involved in the power industry by having independent parties generating power or by permitting various traders negotiating power prices [1]. Delivering real-time and historical data to the interested parties needs to be done reliably and fast without jeopardizing the stability of the grid itself.

A new data infrastructure is needed for distributing information to legitimate entities in a timely, secure and accurate manner. We are designing and implementing a new status service for the power grid; at the middleware layer to best support heterogeneity in devices and applications [2]. GridStat [3] is a flexible and manageable status dissemination middleware for the power grid. It provides a publisher-subscriber model built on the principles of space, time and flow decoupling. GridStat delivers Quality of Service (QoS) properties (such as redundancy and fault tolerance) as required by grid monitor and control applications.

## 2. GridStat: Status Dissemination Middleware

The entire GridStat architecture is built on best-effort UDP-level message communication (future work involves using DIFFSERV or other mechanisms that are better than best-effort, to provide better timeliness guarantees and also thwart some denial-of-service attacks). Each message captures a single event, which is expressed in an event language. Any event (or status item) generated by either publishers (original/simple event) or QoS managers (multiple status events constructed from singleton events) has static properties such as name, owner, value and security information as well as time-related dynamic properties such as rate of change. FIFO order is implemented as the message-ordering scheme, because we have found no need in grid applications to provide causal or total ordering across status flows. One possible avenue of future work here allows for compound events (which we support) to be extended to include temporal correlation.

QoS policies are enforced by QoS Managers. The QoS properties specified include, among others, fault tolerance and timeliness.

Publishers, subscribers and QoS managers are the active entities that initiate communication and status flow within GridStat. The QoS manager is responsible for performing admission control on subscription requests. A QoS manager tries to implement requests as set by subscribers, publishers or other higher-level QoS managers and deliver status items from the publishers to the subscribers according to the QoS policy. Control decisions are made at this node. Many hierarchies are possible, based on the geographical and power industry considerations for a given portion of the grid. Figure 1 illustrates such a hierarchy.

## 3. Fault Tolerance Issues in GridStat

One of the essential and novel features of the GridStat architecture is the ability to embed QoS properties within the middleware. Fault tolerance and timeliness are among the QoS requirements that are supported by the architecture, and which we are implementing now.

Spatial redundancy is one of the fault tolerance techniques employed by GridStat that allows the deployment of redundant paths for transmitting data in order to tolerate a number of benign crash or timing failures. Subscribers specify their desired QoS properties of latency and redundancy for a status item whereas QoS managers reserves the path(s) prior to the status transmission. Path reservation is based on the subscriber's indicated latency and the network link transmission delays.

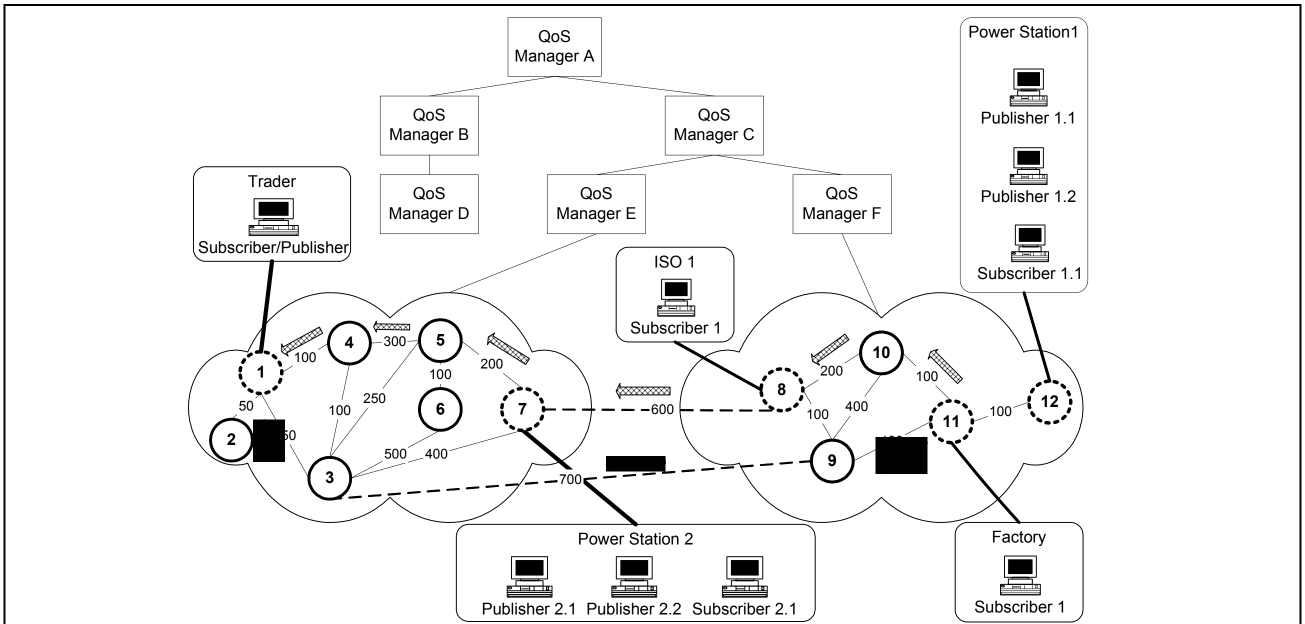


**Figure 1. Publish-Subscribe with Hierarchical QoS Managers**

Figure 1 illustrates the link latencies (in milliseconds) for a sample QoS manager network. Suppose that the entity *Factory* wants to receive all the power price changes posted by entity *Trader*. The subscription request is delivered to its edge QoS Manager 11 with the properties of worst time timeliness of 1500 ms and path redundancy of 2. QoS Manager 11 handles the request to the 'cloud' manager, QoS Manager F, for establishing the paths satisfying the indicated properties. Since the publisher entity does not reside within its controlled area, QoS Manager F contacts its manager with the request information. QoS Manager C will query both QoS Manager E and QoS Manager F for paths starting from QoS Manager 7 or QoS Manager 3 to QoS Manager 1 and for paths from QoS Manager 11 to QoS Manager 8 or QoS Manager 9 respectively. Let's assume that part of the results of the queries received are <3,1>, <7,5,4,1>, <11,10,8> and <11,9>, where each tuple indicates a path composed of QoS Manager nodes. QoS Manager C will setup the paths <11,10,8,7,5,4,1> and <11,9,3,1> as the two redundant optimal paths of latency 1500 ms and 950 ms respectively.

A probabilistic model can be also used for calculating status delivery probabilities based on the probability density function (pdf) describing delays for each network link; this is future work. The link specification includes a steady state availability probability, indicating the fraction of time the link is available, and the link latency pdf, which characterizes the delay experienced on the link. The subscriber will still submit its subscription request with a desired latency, but the GridStat network of QoS Managers will compose the pdfs for all the links involved in the path in order to measure various probabilities. The probability of missing no packet (steady state), one packet in a row, *n* packets in a row, etc. will be derived from those link functions.

GridStat provides subscriber-side caches of the latest value, and we are investigating techniques to compensate for a missing or late update even by extrapolation: keeping a recent history of updates, and then updating the cached value the subscriber uses with a moving average to replace the missing update.

## Acknowledgements

This work was funded in part by the US Dept. of Commerce, National Institute of Standards and Technology Grant #60NANB1D0016 (Critical Infrastructure Protection program), in a subcontract to Schweitzer Engineering Labs Inc., www.selinc.com.

## References

[1] North American Electric Reliability Council, NERC Operating Manual, Policy 10. See http://www.nerc.com.

[2] Bakken, David, "Middleware", Chapter in *Encyclopedia of Distributed Computing,* J. Urban and P. Dasgupta, eds., Kluwer Academic Publishers, 2002, to appear. See http://www.eecs.wsu.edu/~bakken/middleware.pdf.

[3] Bakken, D, Bose, A., Hauser, C., Dionysiou, I., Gjermundrød, H., Xu, L, and Bhowmik, S. "Towards More Extensible and Resilient Real-Time Information Dissemination for the Electric Power Grid", *Proceedings of Power Systems and Communications Systems for the Future*, International Institute for Critical Infrastructures, Beijing, September 2002, to appear. Available via http://gridstat.eecs.wsu.edu.