

SIENA: Wide-Area Event Notification Service

Antonio Carzaniga
David S. Rosenblum
Alexander L. Wolf

Event Services Are Not New!

- ◆ Event models
 - simple (UNIX signals)
 - sophisticated (database triggers)
- ◆ Event notification services
 - LAN message servers (e.g., SUN ToolTalk, DEC Fuse, HP SoftBench)
 - USENET News (NNTP)
- ◆ Basic communication infrastructure
 - point-to-point (TCP/IP, RPC, CORBA, RMI)
 - multicast (MBONE)

Challenges

◆ *Scalability*

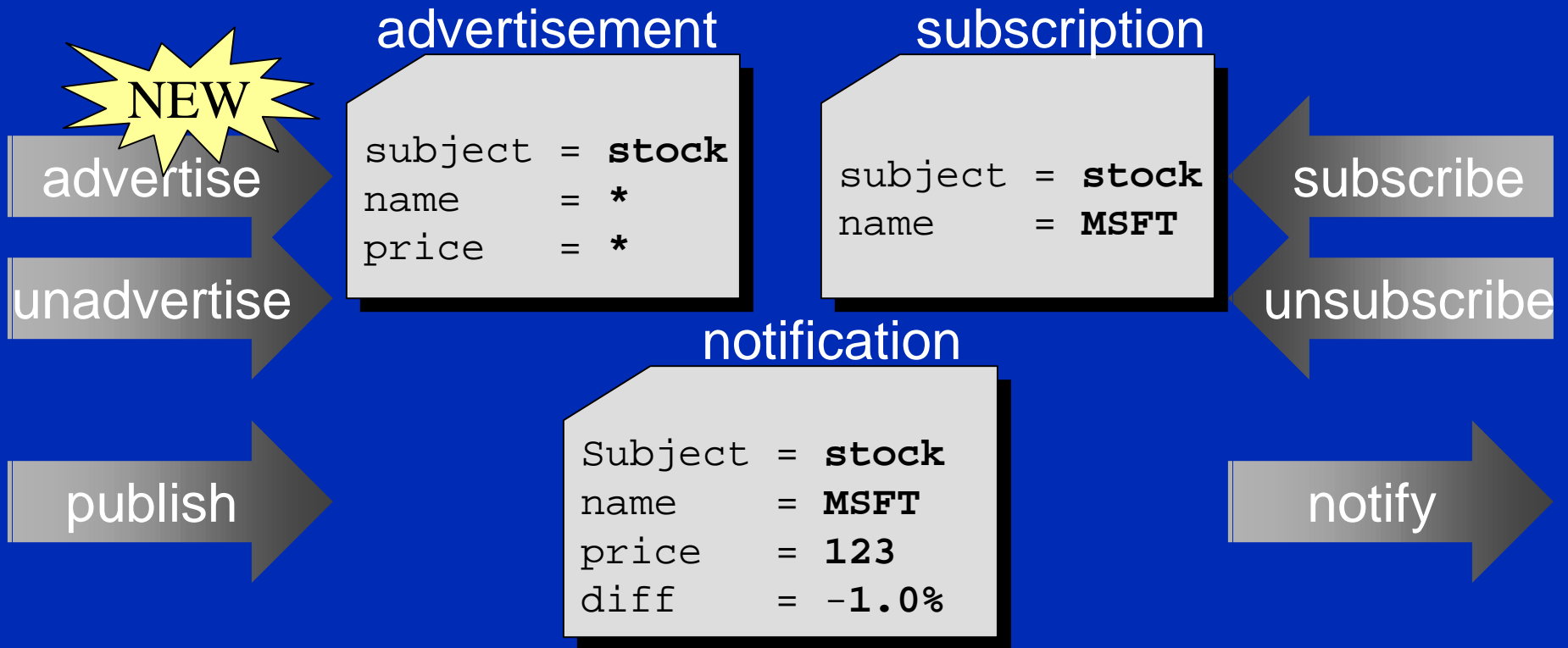
- many-to-many communications
- large number of objects and events
- decentralized control
- heterogeneous platforms and protocols

◆ *Expressiveness*

- rich event notification model
- expressive subscription language (an IP multicast address just doesn't cut it)

◆ Tradeoffs between scalability and expressiveness

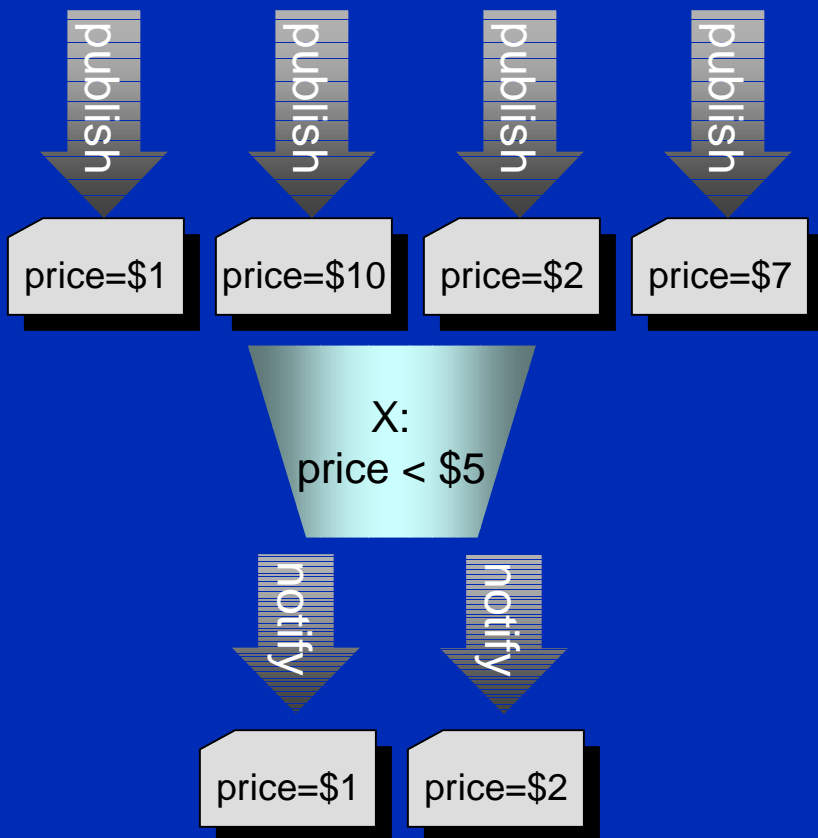
SIENA Event Service



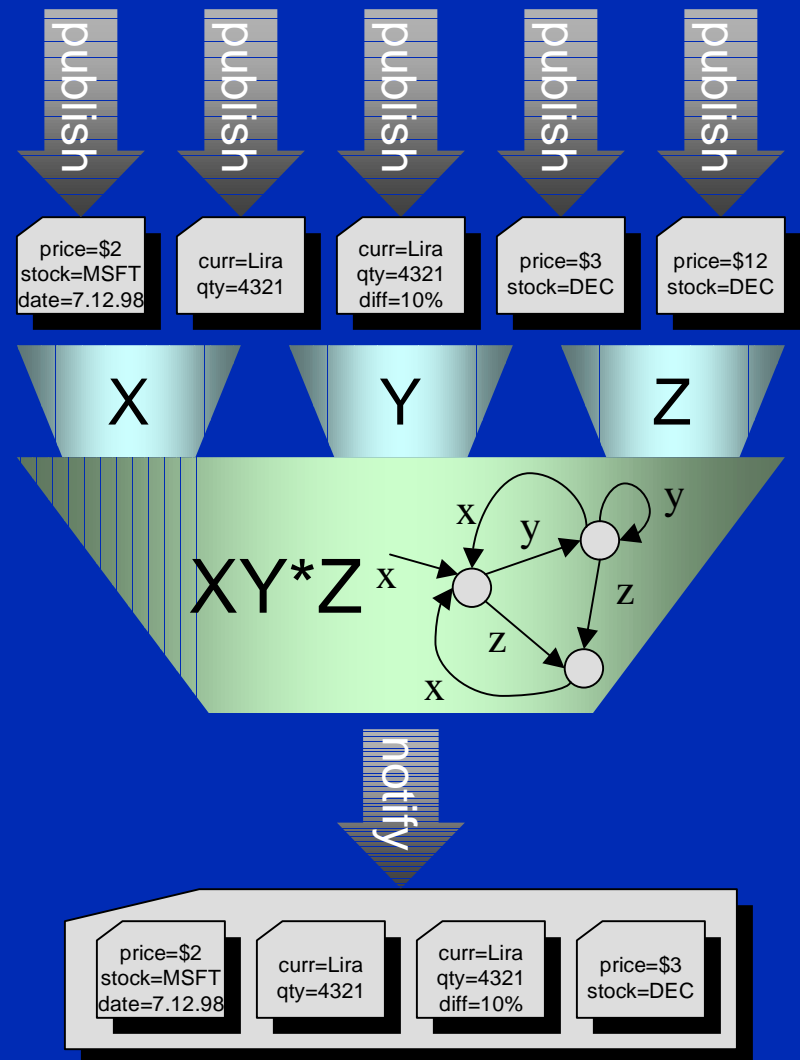
Filters and Patterns

Filter

X: price < \$5

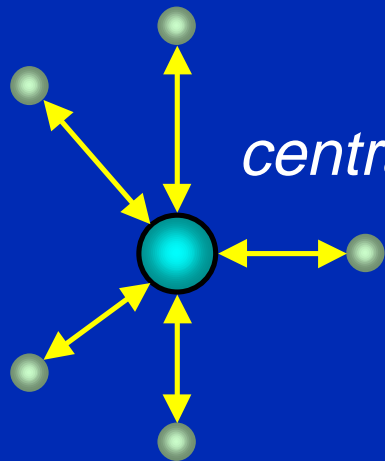


Pattern

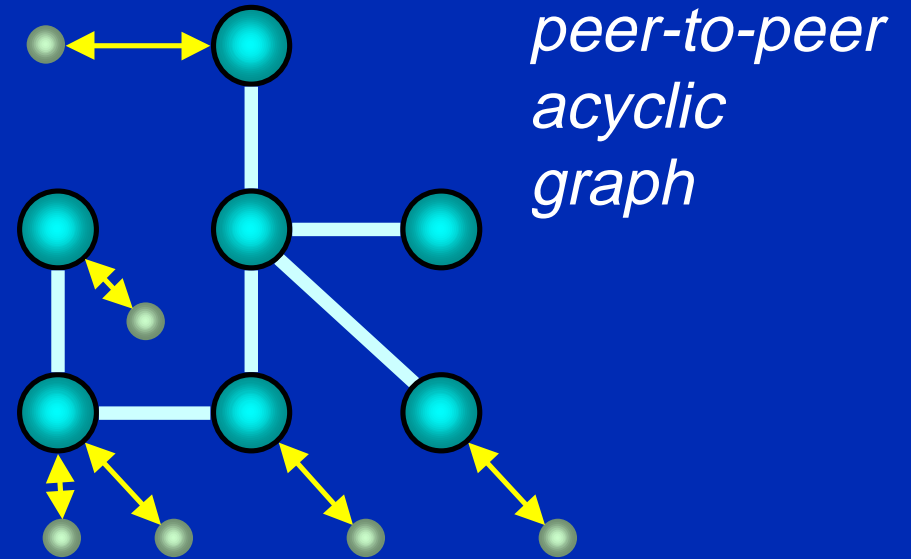
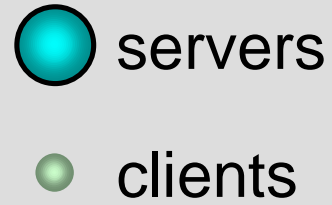




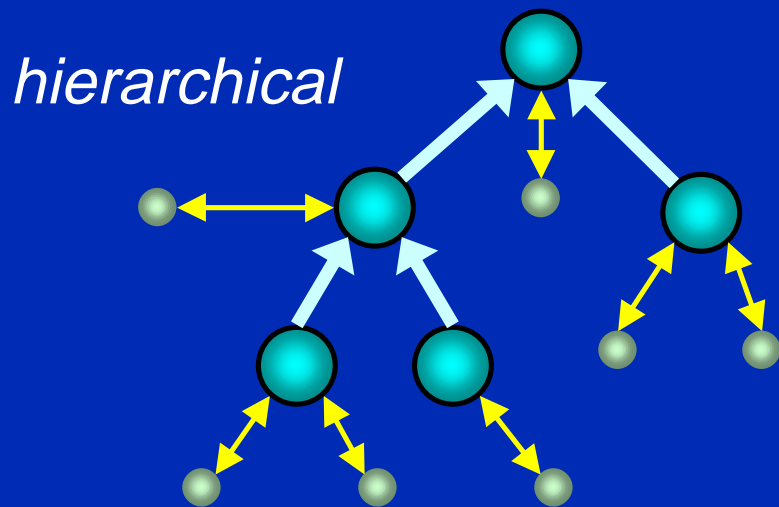
Event Service Topologies



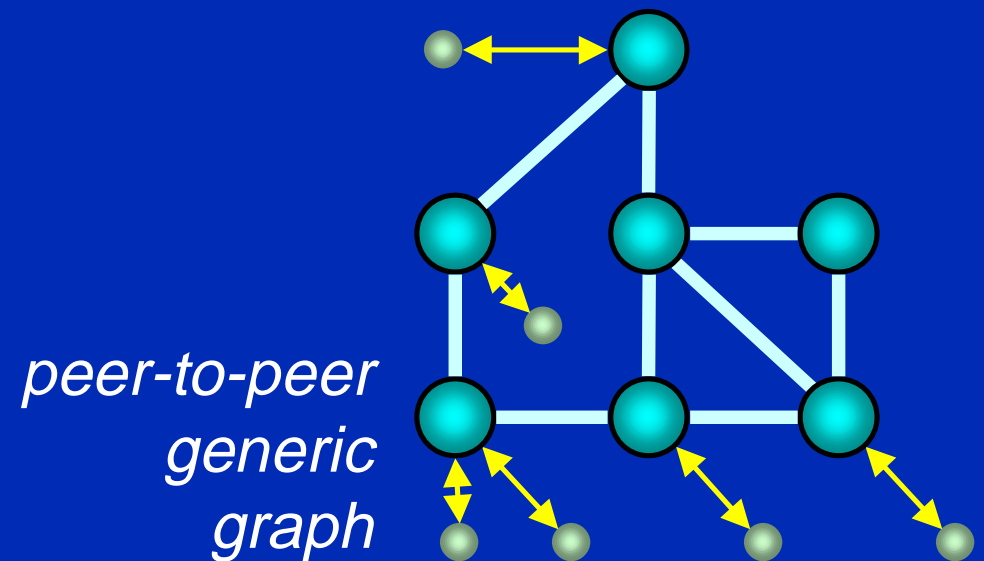
centralized



*peer-to-peer
acyclic
graph*



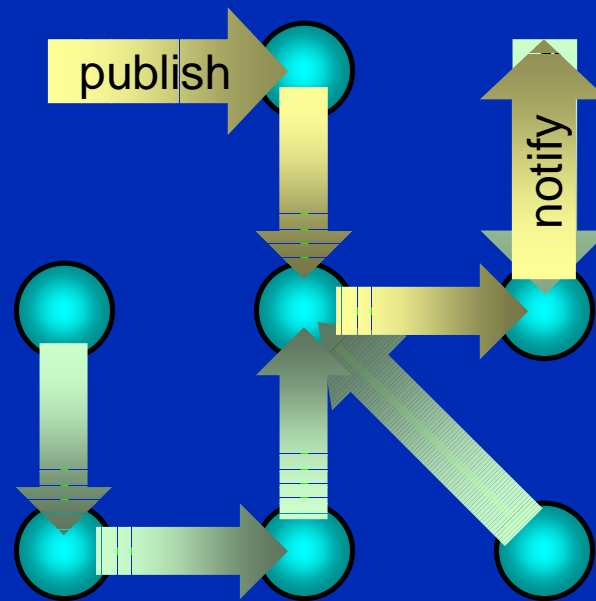
hierarchical



*peer-to-peer
generic
graph*

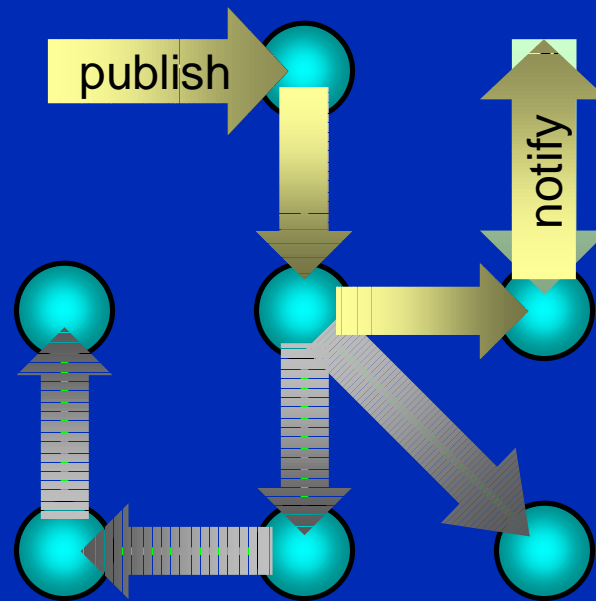
Subscription Forwarding

- ◆ subscriptions are forwarded to every server
- ◆ notifications follow (backwards) the paths set up by subscriptions



Advertisement Forwarding

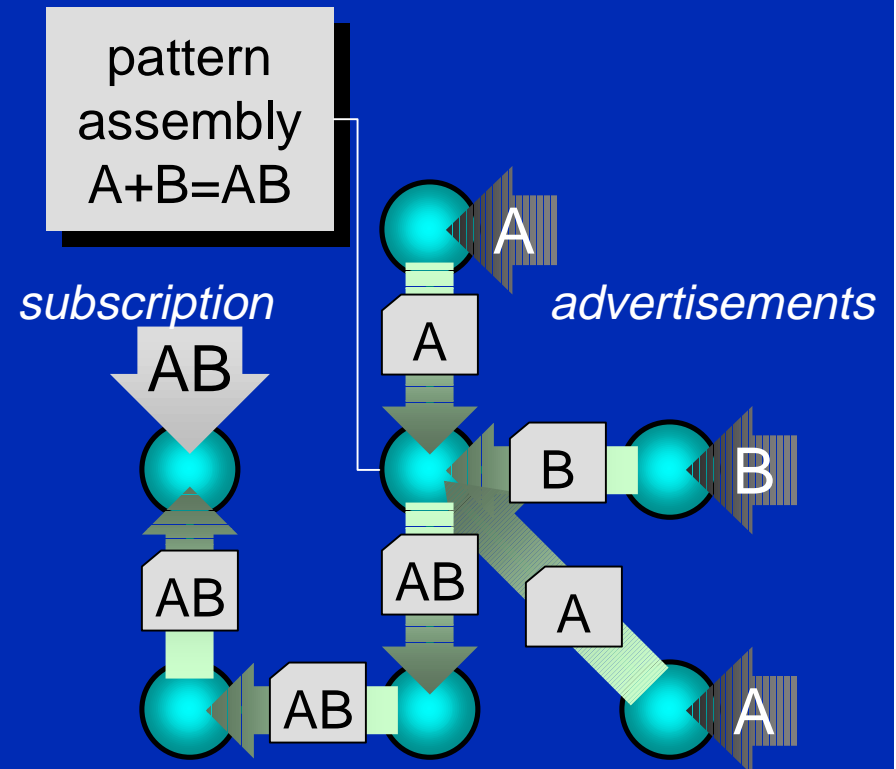
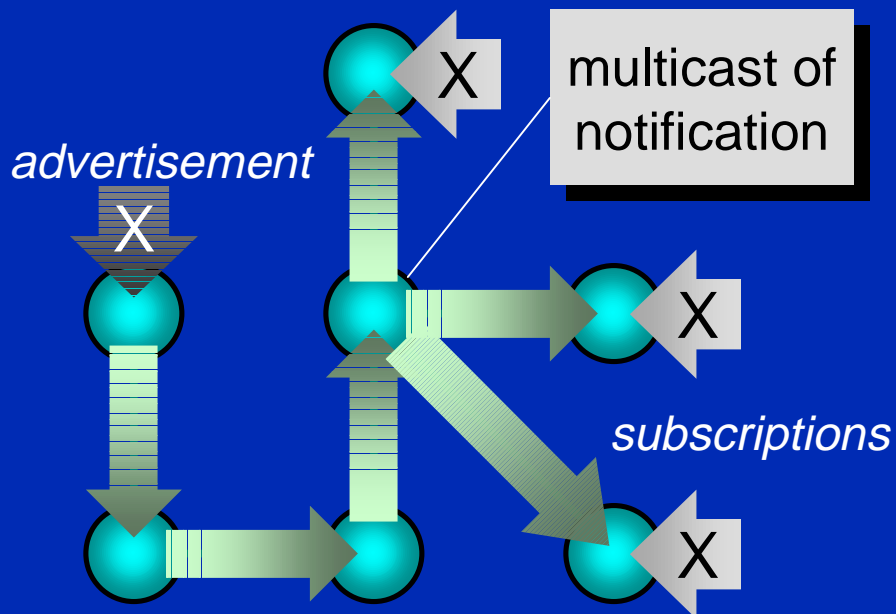
- ◆ advertisements are forwarded to every server
- ◆ subscriptions follow (backwards) and “activate” the paths set up by advertisements
- ◆ notifications follow the paths “activated” by subscriptions



Optimization

- ◆ Multicasting is done downstream (closer to interested parties)

- ◆ Filters and patterns evaluated upstream (closer to objects of interest)



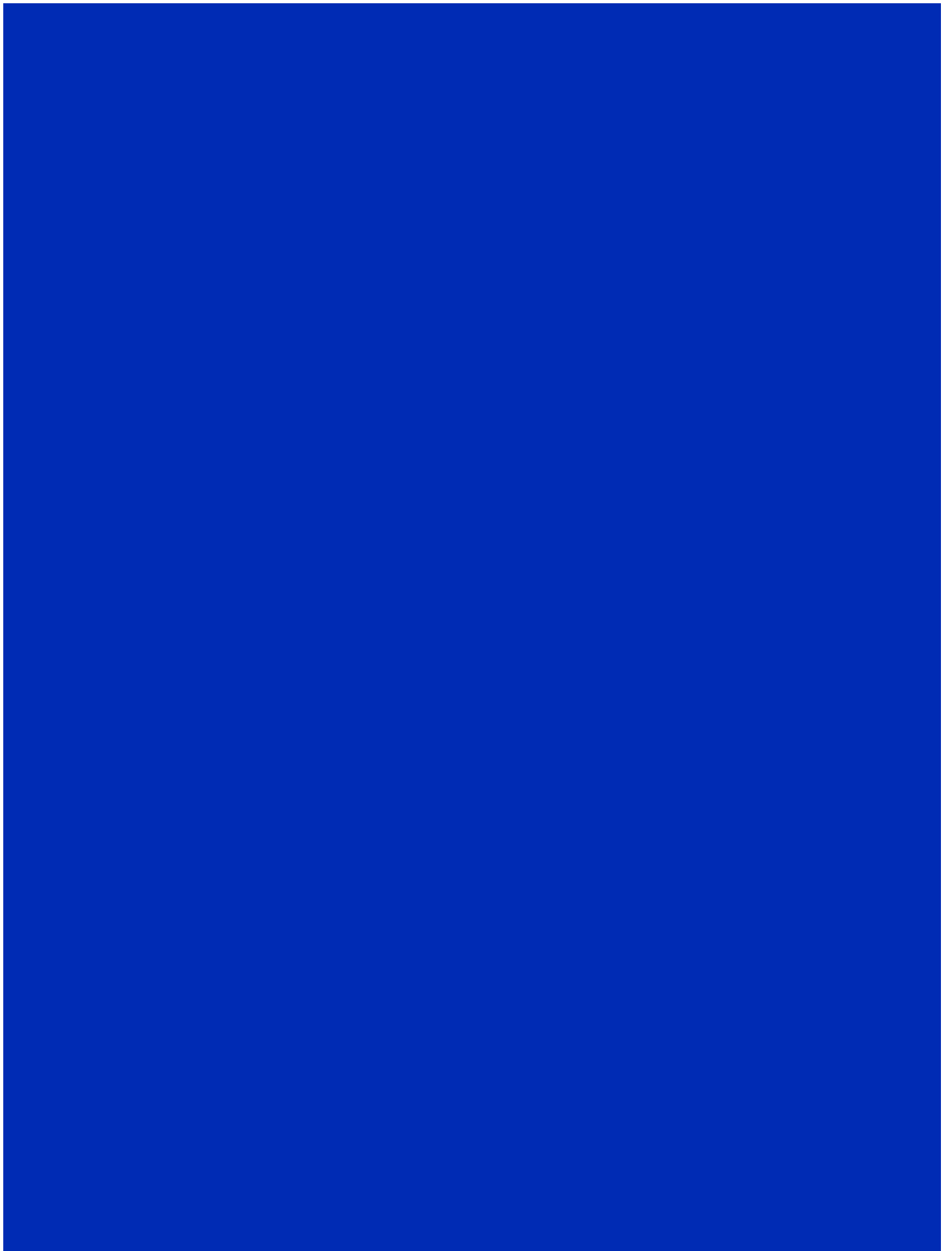


Design Validation

- ◆ Problem: *validation* of server topologies and routing strategies with respect to
 - network traffic
 - memory usage
 - CPU usage
- ◆ SIENA approach: *simulation*

Parameters for Simulations

- ◆ Network:
 - topology, latency, bandwidth, cost function
- ◆ Servers:
 - location, topology, routing strategy
- ◆ Objects of interest and interested parties
 - number, location, primary server, behavior



Prototype

- ◆ C++/Java servers
- ◆ C++/Java client interface
- ◆ XML data representation
- ◆ Several communication protocols
 - SMTP (e-mail), HTTP, plain TCP/IP

Conclusions

- ◆ *SIENA is an architecture that implements a scalable and flexible event service*
- ◆ event service interface
- ◆ notification model
- ◆ service topologies
- ◆ notification dispatching strategies
- ◆ validation through simulation
- ◆ prototype