

NaradaBrokering

Grid Computing: Making the
Global Infrastructure a Reality

Chapter 22

CSE 719 Seminar

Dr. Russ Miller

Presented by Martins Innus



Introduction

- Resource brokering system
 - Resource sharing
 - Peer interactions
 - Advertising
 - Search
 - Discovery
 - Request
 - Response



Proposal

- Scalable
- Durable for Peer to Peer (P2P) grids
- Clients
- Resources
- Dynamic P2P collections



Requirements

- Flexible
- Fault tolerant
- Efficient
- High Performance



NaradaBrokering

- Event brokering
- Network of cooperating brokers
- Link clients to resources
- Events are messages with timestamps
- Scale from PDAs to HPC



NaradaBrokering Outline

- Publish/Subscribe model
- Handles access to services
- Allow P2P clients at the edge to talk directly
- Allow P2P and traditional centralized broker model



Definition

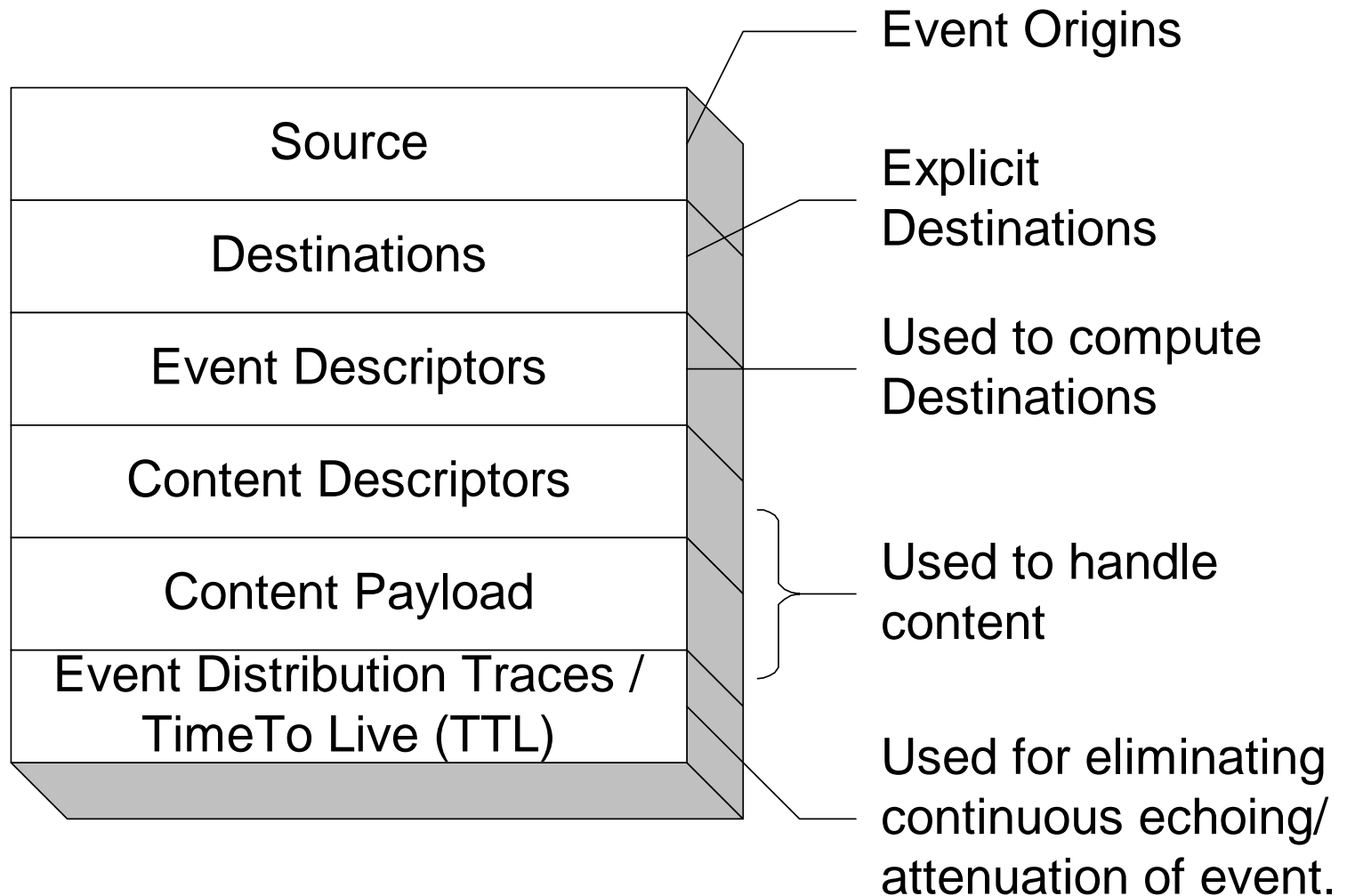
- Event brokering system
- Large network of broker nodes
- Content based routing
- Publish/Subscribe model



Layout

- Cluster topology
 - Calculate routing information
 - Work around failures
- Asynchronous communication
- Publish interest in a resource
- Receive response
- Deliver matched events after reconnect

Event

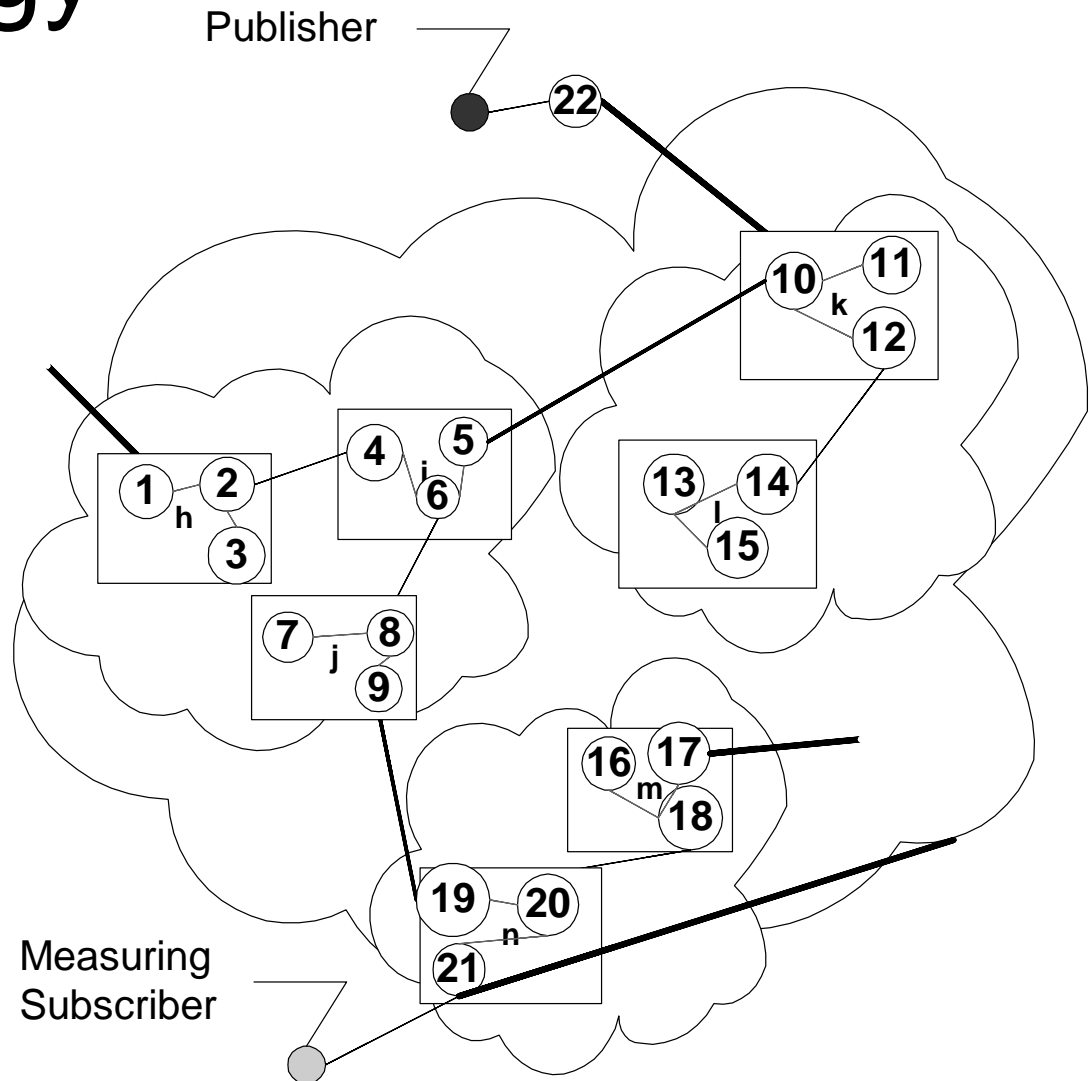




Failure and Recovery

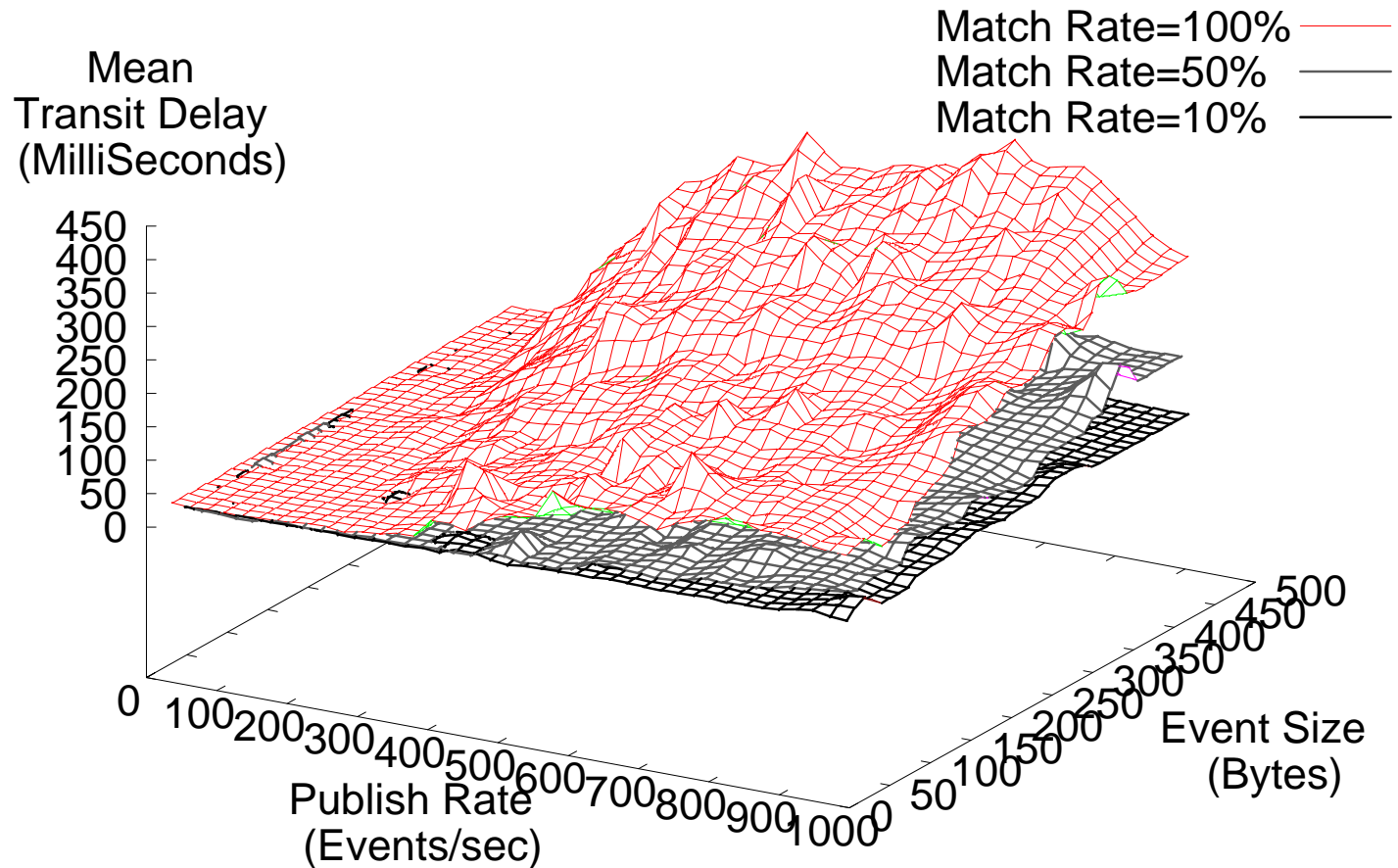
- Independent storage
 - State storage
- Multiple locations in the topology
- Thus, brokers are stateless

Test Topology



Results

Transit Delay under different matching rates: 22 Brokers 102 Clients



JMS Compliance

- JMS

- Unified API for pub/sub model
 - Like MPI for cluster computing

- Support JMS clients

- Transparency
- Access to JMS applications

- Bring NaradaBrokering functionality to JMS clients

- Replace the single server with a distributed solution
- Scalability, resilience, load balancing



JMS Support

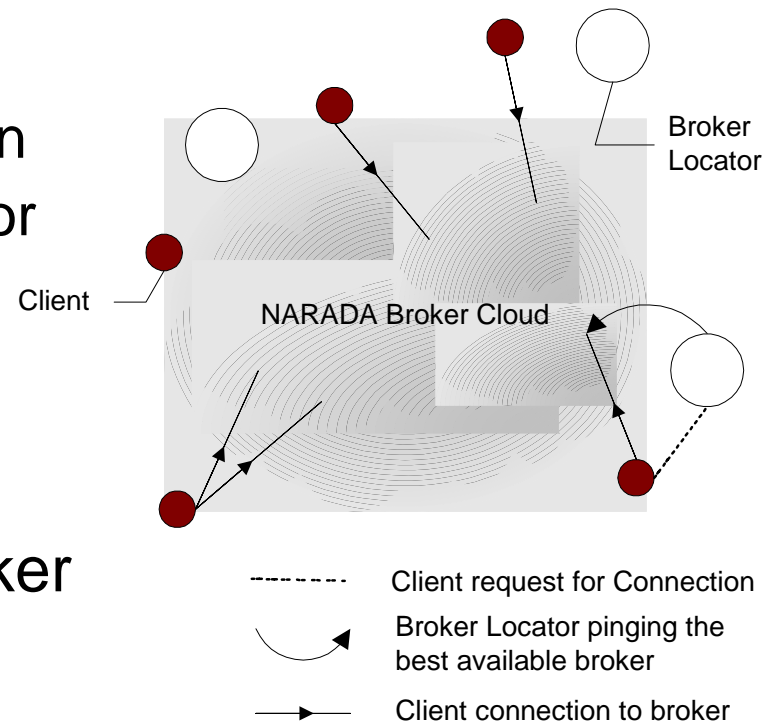
- Bridge
 - Operations complete locally or mapped to NaradaBrokering infrastructure
- High availability
- Support JMS message types
- Encapsulate JMS messages with NaradaBrokering headers

JMS Transparency

- Insulate JMS clients from knowledge of all brokers
- Use broker locators to find valid brokers
 - Load balancing
 - Prefer new brokers
 - Multiple brokers available
 - Like DNS, no single point of failure
 - If one fails, no big deal

Broker Locators

- Locate valid broker
- Propagate broker information to client
 - Hostname/IP-address information
 - Port number on which it listens for connections
 - Transport protocol over which it communicates
- Client then uses info to establish communication channel with broker
 - Done transparently.
- Clients with multiple connections
 - A client could sometimes have connections to multiple brokers.



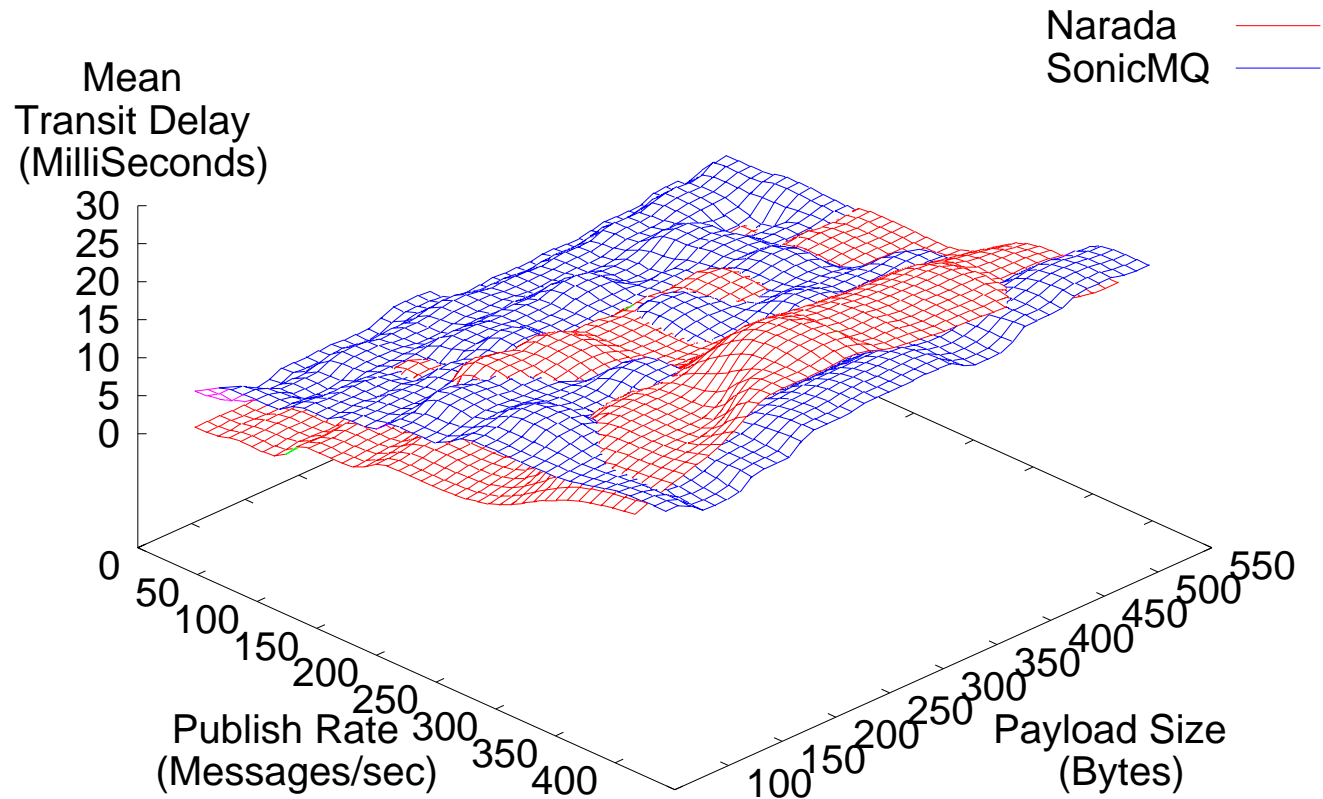


Performance

- Compare to SonicMQ
- Publish and subscribe to the same topic
- 100 subscribers
- Measure the transit delay

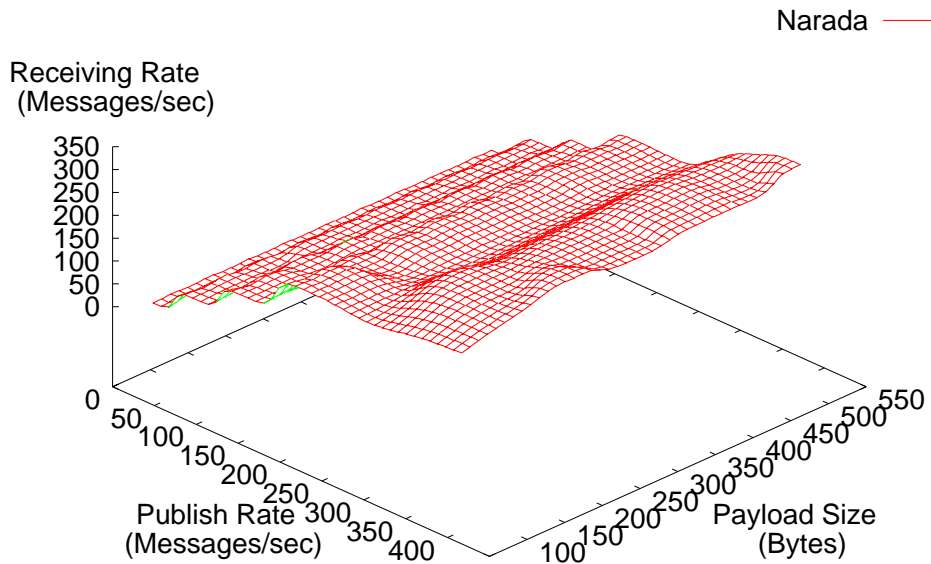
Performance Graphs

Transit Delays for Message Samples in Narada and SonicMQ

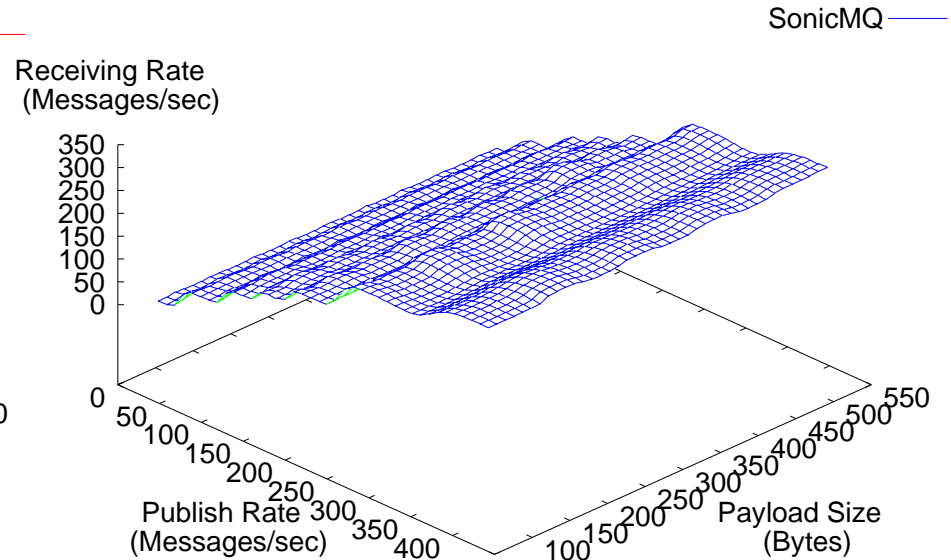


Performance Graphs

System Throughputs - Narada



System Throughputs - SonicMQ





NaradaBrokering and P2P

- Discovery of services
- Routing
- Deliver content efficiently
- Locating peers
- Forward requests only to relevant peers
- Connect islands of peers
- Hybrid model for local peers

JXTA

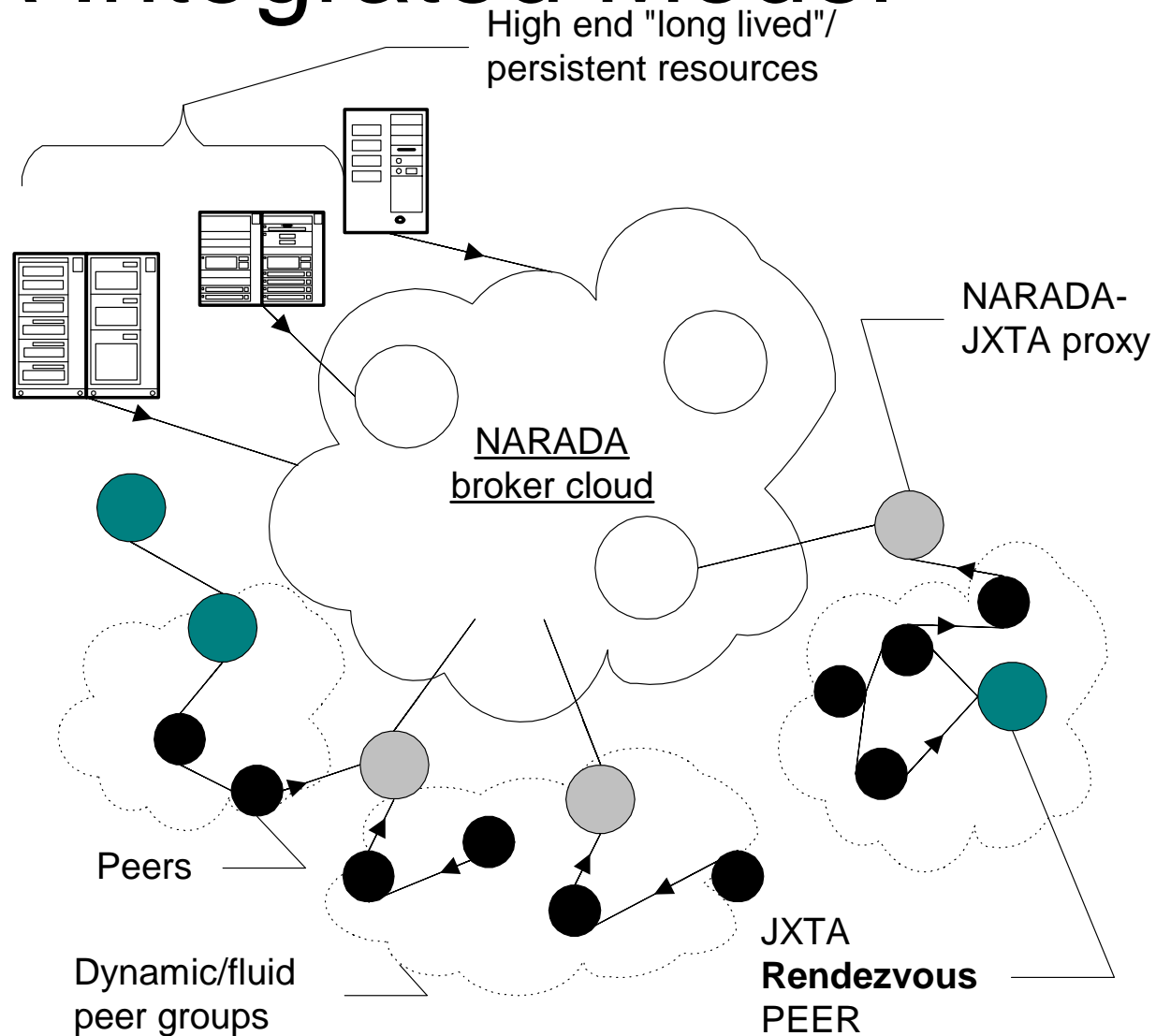
- Open protocol to support P2P
 - Indexing, file sharing, searching, security
- Implemented by local forwarding of messages
 - Use TTL to prevent flooding
- Tends to be localized



JXTA integration

- Keep the NaradaBrokering and JXTA cores intact
- Peers don't communicate with NaradaBrokering directly
- Develop a proxy
- Peers unaware that NaradaBrokering is routing some requests

JXTA Integrated Model



Proxy

- Initialized as a NaradaBrokering client and JXTA Peer
- Advertise as a JXTA proxy
- NaradaBrokering handles only sending events to peers that are appropriate
- Claim that peer discovery is faster ???

References

- Grid Computing: Making the Global Infrastructure a Reality
- www.naradabrokering.org
- java.sun.com/products/jms
- www.jxta.org