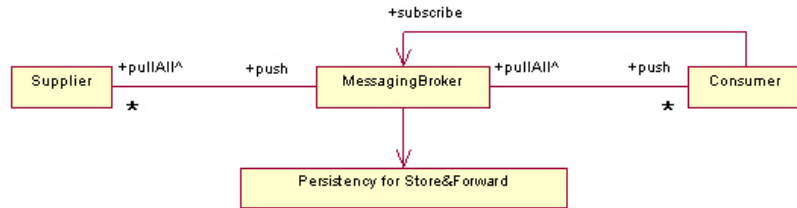


Messaging, Logging, & Directory Patterns  
for  
January 8, 2003 OGSA-WG Teleconference  
Discussions

OGSA-WG Logging Focus Area  
Bill Horn  
January 8, 2003

# Messaging, Logging, & Directory Patterns

Messaging

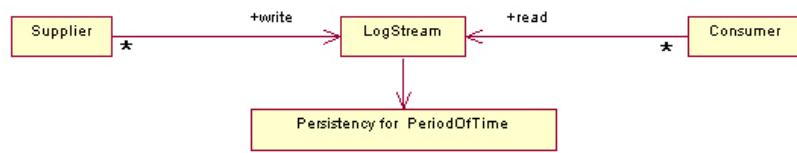


JMS  
CORBA Notification  
WS-Events

Move data from suppliers to interested consumers w/ a variety of reliability QoSs

4.1.3 Service Communication Services  
6.14 Messaging & Queuing  
6.15 Event

Logging

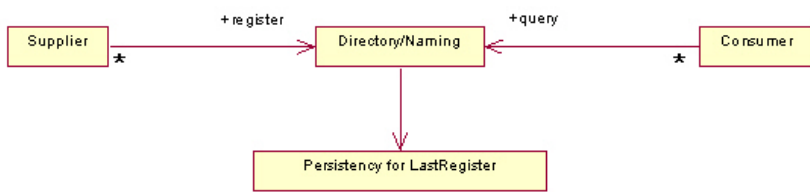


POSIX SYSLOG  
zOS System Logger  
NT Events

Read-only (no update) access to sequentially-written records  
Optimized for performance

4.1.3 Service Communication Services  
6.13 Logging Services  
6.15 Event

Directory/Naming

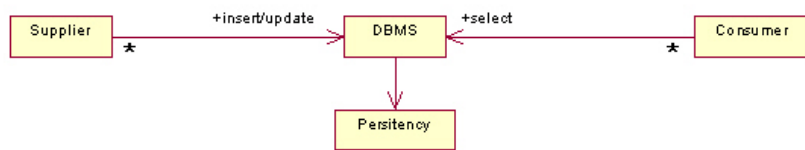


LDAP  
DNS  
NIS / YP  
CORBA COS Naming  
MS Active Directory

Optimized for Searches  
Quick Response to Hi-volume Queries  
Replication w/ temp Inconsistencies

4.1.1 Service Interaction Services  
6.3 Service Groups & Discovery Services  
6.19 Discovery Services

DBMS

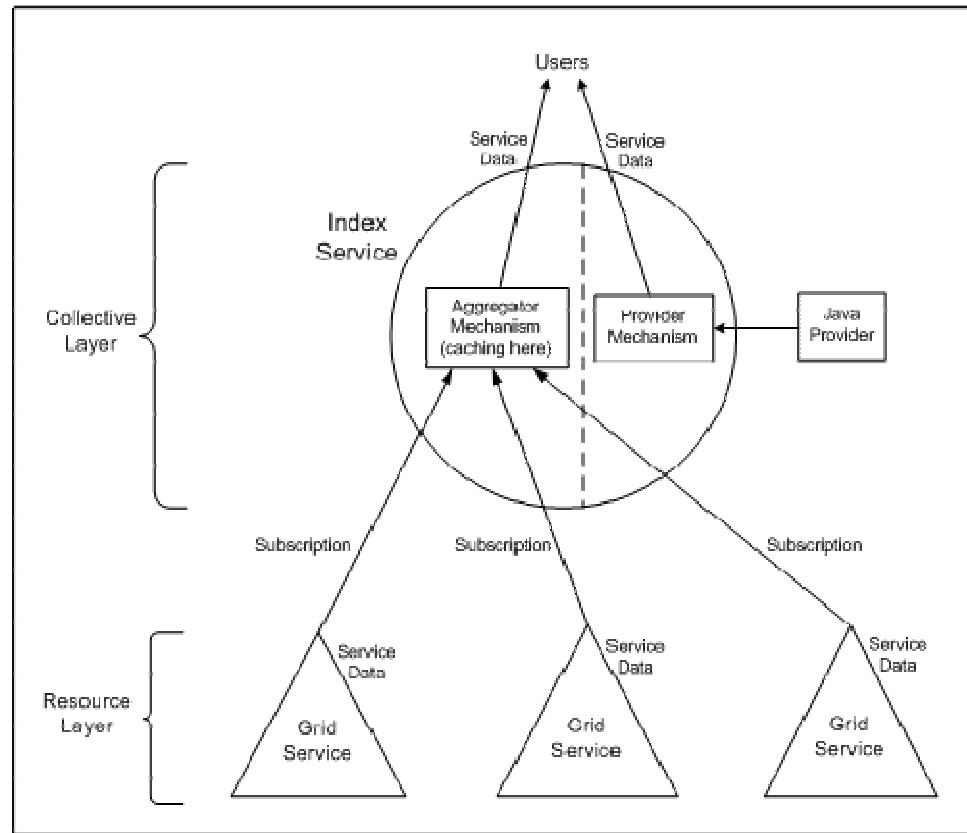


RDBMS  
OODBMS

Transactions / Rollback  
Hi-volume Selects, Inserts, Updates, Deletes  
Complex Expressions  
Inconsistencies not tolerated

4.2 Data Services  
6.17 Base Data Services  
6.18 Other Data Services

# MDS in GT3



from: <http://www.globus.org/ogsa/releases/final/docs/infosvcs/MDS.html>

# Example from GMA

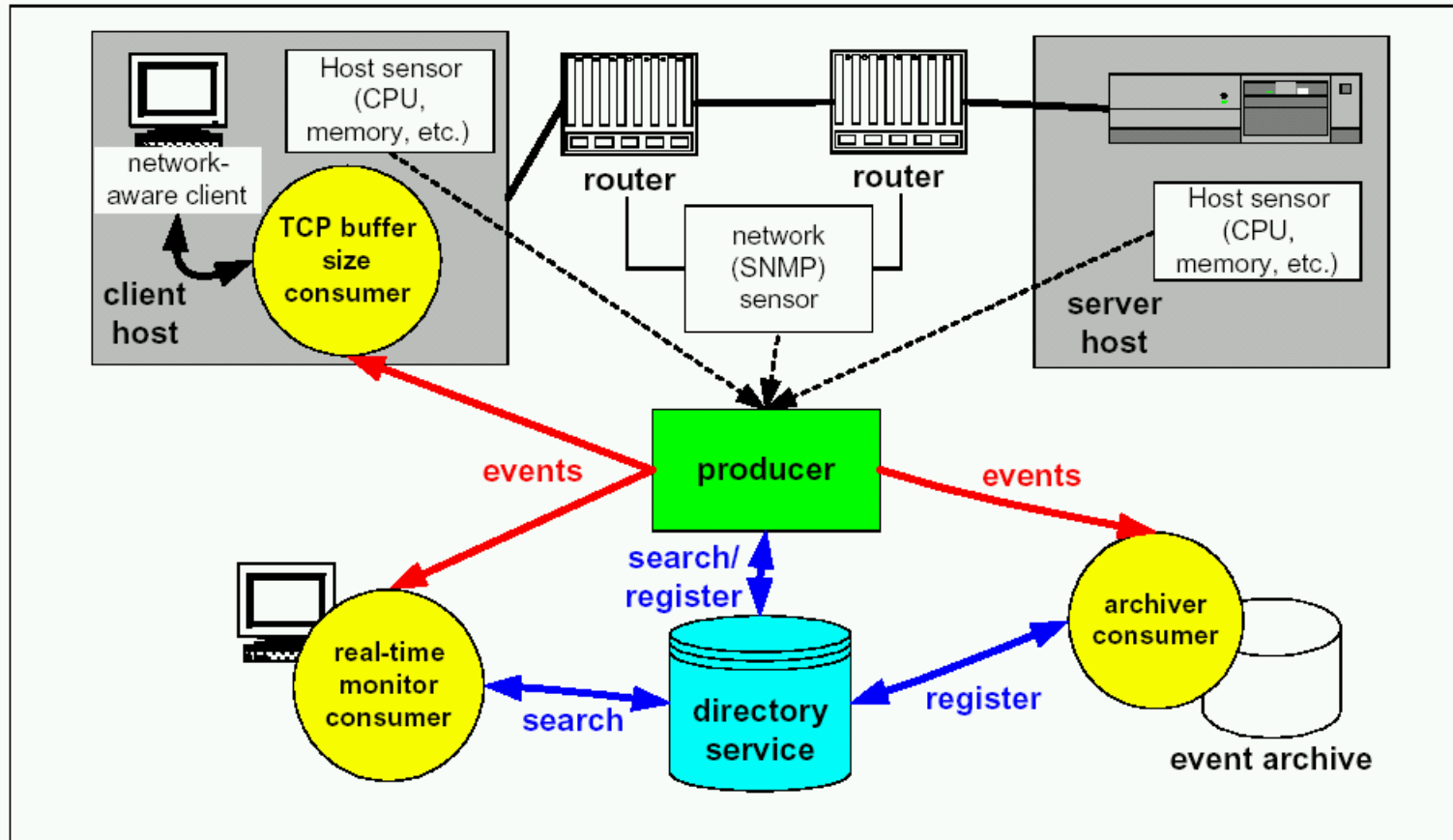


Figure 3: Sample Use of Monitoring System

from: <http://www-didc.lbl.gov/GGF-PERF/GMA-WG/papers/GWD-GP-16-1.pdf>

# OGSA Logger System: Requirements

## Abstraction of Legacy Logger Systems

- eg zOS System Logger, NT events, and UNIX syslog

## Persistency

- fundamental requirement

## Support for Sequential Record Input and Output

- sequential I/O with record granularity

## Ordering of Log Records

- (at least a partial) ordering based on timestamp

## Standard Schema for Log Records

- standard xsd for log artifacts

## Stateful Read Cursor

- not invalidated by concurrent reads / writes

## Decoupling of Suppliers from Consumers

- eg audit, management, problem determination

## Broker

- concurrent access by multiple suppliers & multiple consumers

## Filtering

- everywhere, as close to source as possible

## Merged Log

- support for underlying mechanisms that permit the merging of multiple log streams

## Synchronous and Asynchronous Write Semantics

- .with: ack, callback for ack, and fire & forget

## Duplication of Log Streams

- .create a new stream from an existing stream

## Deletion of Log Records

- time-stamp based deletes

## Coexistence w/ Messaging Fabric

- log-type persistency for the fabric

## Standard Set of Specialized Logstreams

- Secure Logs (security QoS)
- Globally Ordered Logs
- Circular Logs
- Duplexed Logs (reliability QoS)
- Compressed Logs

# OGSA Logger System (V7)

**cluster**

**merge**

**logstream**

**connect**

**read cursor**

