Notification Service

Ananth Krishna, Victor Tan, Richard Lawley,
Simon Miles, Luc Moreau

School of Electronics and Computer Science
University of Southampton.
Introduction

• Notification service is a part of myGrid that essentially enables asynchronous delivery of messages between distributed components in myGrid.

• myGrid is an e-Science project to help biologists and bioinformaticians perform workflow based in silico experiments.

• In myGrid, some services described in the workflow may depend on others for results. It is inefficient and unnecessary to run all services/user agents at all times.

• A notification service, in myGrid is used to forward messages to these user agents by storing messages in their absence.
Features

- Asynchronous delivery and persistence of messages
  - Asynchronous messaging refers to a situation where publishers initiate the communication regardless of whether subscribers are actually available.
  - In myGrid biologists and services are not always available.

- Topic based Publish/Subscribe messaging
  - Topics are used to identify the kind of message sent or received and to group and distribute messages addressed to it.
  - Topics also facilitate a many-to-many relationship model.
Features

- **Push/Pull models**
  - The notification source can decide when to push notifications to the sink or alternatively the sink can poll the source for notifications.

- **Transient and Durable subscriptions**
  - Transient subscriptions exist as long as the subscriber is running.
  - Durable subscriptions use persistence to provide a reliable service and they survive server failure.
  - Clients in myGrid require both forms of subscriptions as subscribers may or may not choose to run at all times.
Features

• Quality of Service

  – Publishers and subscribers may have different requirements.

  – Personalisation of the service is achieved by allowing clients to define a Quality of Service.

  – Quality of Service in the notification service allows publishers to specify a publish limit, subscribers to control frequency with which notifications are received and either of them can specify topic type on creation.
Features

• JMS Compliance

  – The notification service is JMS compliant.

  – JMS has been chosen because it guarantees message delivery once and only once through message acknowledgement.

  – JMS also offers support for session management, message persistence, security, naming and directory services, etc which otherwise would have to be provided by notification service components.
Features

- Message Filtering

  - Subscribers are often interested in specific notifications under a topic.

  - Filters which are key/value pairs can be specified by subscribers.

  - Filters are converted to JMS compliant filters.
Novel Features

• Durable Topics

  – Durable topics are introduced to solve a potential race condition in publish/subscribe messaging where a publisher publishes a message on a topic before a subscriber has subscribed on the same.

  – Durable topics retain all messages sent on a topic and serve them to subscribers subscribing on such topics.

  – Durable topics allow more flexibility to services and users in myGrid.
Novel Features

• Topic Management

  – Allows creation and deletion of topics dynamically.
  
  – Hierarchical namespace in topics is supported.
  
  – Hierarchical structure allows publishers to publish under specific topics rather than broad based generic topics.
  
  – Subscribers can use wild cards to subscribe to more than one topic in a hierarchical namespace.
Novel Features

- Web Service and Grid Service compliance
  - The notification service allows clients to access all its features through web service calls.
  - A WSDL interface is defined which exposes all functionality of the service and is deployed inside an Axis engine.
  - The notification service is moving towards Grid service compliance.
Novel Features

• Federated notification services
  – Some organisations have a network of notification services working collectively.

  – This can be achieved by simple routing via hubs in a peer-to-peer manner.
Novel Features

Negotiation over QoS

[Diagram showing negotiation process between Client, Client NC, Provider NC, and Provider]
• Architecture of myGrid Notification Service.
• Architecture of myGrid Notification Service.
• Architecture of myGrid Notification Service.
• Architecture of myGrid Notification Service.
Architecture

- Architecture of myGrid Notification Service.
• Architecture of myGrid Notification Service.
• Architecture of myGrid Notification Service.
Architecture

- Architecture of myGrid Notification Service.
Evaluation and Results

- Performance of a messaging middleware can be measured in various ways. In this section we have concentrated on measuring performance in terms of average time taken to publish 1000 messages.

- The experimental setup involved deploying the notification service on a Windows platform with a Pentium 4 processor and 750 MB memory and measuring publish time for publishing up to 10000 messages with some or all of the features turned on/off.
Evaluation and Results

Performance stats for non persistent messages

Time taken to publish in milliseconds

MGNS
JMS
MGNS as web service

Number of Messages in thousands

Performance stats for non persistent messages

MGNS
JMS
MGNS as web service
Evaluation and Results

Performance stats for persistent messages

Time taken to publish in milliseconds

Number of Messages in thousands

MGNS

JMS
Evaluation and Results

Performance stats for durable topic and message filter

- MGNS with durable topic
- JMS
- MGNS with durable topic and message filter

Number of Messages in thousands vs. Time taken to publish in milliseconds
Conclusion

- The notification service is aimed at the bioinformatics community but also presents some interesting aspects like durable topics, negotiation over QoS, federated services in p2p, hierarchical topic structure and web service compliance which are new in distributed computing and messaging middleware.

- The results obtained from initial tests show that most of the features supported in the service justify the overhead they incur in terms of performance degradation as they offer compensating benefits.
Future work

• Implementing federation of services in a peer-2-peer manner.

• Supporting a dynamic hierarchical topic structure.

• Implementing the standalone negotiation engine into the notification service.

• Measuring round-trip time of a message and performance implications of emerging web service technologies like JAX-RPC, etc.