Objectives

• Develop a peer-to-peer (P2P) middleware to support and enable autonomic interactions.

• Demonstrate the feasibility and benefits of this P2P middleware on a sample application.

Talk Overview

• Overview of the Autonomic Oil Reservoir Optimization [AORO]
  – Components
  – Scenario
• Presentation of Pawn, AORO’s enabling P2P middleware
  – Architecture
  – Components
  – Interactions
  – Services
• Using Pawn for optimizing an oil reservoir
• Conclusions
Components of the AORO Application

- **IPARS**: Integrated Parallel Accurate Reservoir Simulator
  - Parallel reservoir simulation framework
- **IPARS Factory**
  - Configures instances of IPARS simulations
  - Deploys them on resources on the Grid
  - Manages their execution
- **VFSA**: Very Fast Simulated Annealing
  - Optimizes the placement of wells and the inputs (pressure, temperature) to IPARS simulations.
- **Economic Modeling Service**
  - Uses IPARS simulations outputs and current market parameters (oil prices, costs, etc.) to compute estimated revenues for a particular reservoir configuration.
- **DISCOVER Computational Collaboratory**
  - Interaction & Collaboration substrate
  - Distributed Interactive Object Substrate
  - Collaborative Portals

---

**PAWN**:

1. Client configures and launches IPARS Factory and VFSA Optimization Service on resource of choice
2. IPARS Factory discovers and initializes VFSA Optimization Service
3. Client configures VFSA Optimization Service and presents
4. VFSA Optimization Service and presents
5. IPARS Factory discovers and initializes VFSA Optimization Service
6. VFSA Optimization Service generates new well
7. One optimal well placement is determined, IPARS Factory launches IPARS run
8. Current oil price, market state, etc.
9. Scientists/Engineers collaboratively interact with IPARS

---

**Pawn : Conceptual Overview**

> In Pawn: “Peers compose messages handled by services through specific interaction modalities”

Architecture of the Optimization Application components

- IPARS Factory
- VFSA Optimization
- DISCOVER Collaboration
- Economic Model
- Pawn
- GRID/JTIA

- Services
  - Application Execution, Application/Hermeneutic Contract, Application Monitoring and Debugging, Collaboration
- Messages
  - Motions, Interoperation, Coordination, Guarantee
- Pears
  - Client, Member, Application
Pawn: Functionalities Overview

- Provides messaging mechanisms to enable interactions on the computing Grid
- Provides publish/subscribe mechanisms across peer-to-peer domains
- Builds high-level messaging semantics on top of low-level interaction modalities:
  - PUSH: e.g. dynamic data injection
  - PULL: e.g. monitoring
  - REQUEST/RESPONSE: e.g. data interrogation
  - TRANSACTION: e.g. steering
  - FILTERED MULTICAST: e.g. group collaboration
- Built on top of Project JXTA

Project JXTA: Concepts

- Peer
  - Any compute-capable device that "understands" a subset of the common protocols
- PeerGroup
  - A group of peers that share similar interests
- Pipe
  - Communication channels between peers
- Module
  - A general behavior described by a peer or a peer group. JXTA separates the definition of the behavior from its implementation.
- Advertisement
  - A published neutral document [XML] describing a resource
- Security
  - Using secure sockets for every transmission
  - Enforcing membership policies at every peer

Project JXTA: Protocols

- PDP (Peer discovery protocol): used by peers to advertise their own resources
- PIP (Peer information protocol): monitoring peers status and load
- PBP (Pipe Binding Protocol): to establish a virtual communication channel between peers
- PRP (Peer Resolver Protocol): sending and receiving queries and responses
- RVP (Rendezvous Protocol): to propagate messages in a peer group
- ERP (Endpoint Routing Protocol): to find routes from a source to a destination
From JXTA to Pawn

- JXTA provides core capabilities
  - Publication:
    - endpoints publish uniquely identified messages.
  - Advertisement:
    - language-independent document describing a resource
  - Caching:
    - RV peers cache advertisements made by every endpoint and maintain consistent replicas.
  - Routing:
    - path to destination is determined by the nearest rendezvous peer using the endpoint router protocol.
- Pawn extends JXTA to provide
  - Distributed object interaction on top of a peer-to-peer substrate
    - Serialization of objects to XML streams
    - Method invocation on remote objects (RMC)
  - Interest Subscription
    - content-based information dissemination. Every message carries metadata allowing peers to register interest on an attribute basis.

Pawn Interactions (1)

- JXTA communication enabled through:
  - Pipes
    - synchronous: blocking
    - asynchronous: non-blocking
  - Resolver
    - point-to-point messaging
      - TCP streams
      - Datagram packets
    - mixed multicast
      - group distribution
- Pawn services build on these communication mechanisms

Pawn Interactions (2)

- Stateful messages
  - Every message is self-describing and self-sufficient
  - Can be resent by intermediary peers in case of failure
- Message Guarantees
  - Using FIFO queues and timeouts on a per-message basis
- Synchronous/Asynchronous Communication
  - Using blocking pipes (for sync)
  - Using non-blocking pipes and resolve service (for async)
- Dynamic Data Injection
  - Using event channels for dynamic data input
- Remote Procedure Calls
  - Synchronous RPC using XML for invoking remote methods and provide seamless application deployment across loosely-coupled domains
Pawn Services

- Application Execution [AEX]
  - Start, stop and get status of Applications
- Application Monitoring and Steering [AMS]
  - Application querying and management
- Application Runtime and Control [ARC]
  - Publishes application responses and status
- Remote Procedure Calls [PawnRPC]
  - Provides synchronous/asynchronous RPC calls in a platform and language independent manner
- Group communication
  - Handles text messages between groups of clients

Scenario: Peer Deployment

- Client authenticates to the DISCOVER Server running Globus toolkit using GSI
- Once authenticated Clients can deploy IPARS Factory and VFSA optimization peers using Globus GRAM protocol on available machines

Scenario: Peer Discovery

- Peers publish advertisements describing their identity and functionalities
- Using underlying JXTA Discovery services, peers discover the advertisements and can start interacting
**Scenario: Optimization Process**

- VFSA sends a well position guess to IPARS Factory
- IPARS Factory checks in Database if guess has already been run
  - If guess found, result is returned to the clients and a new guess from VFSA is generated
  - If not found an IPARS instance is run
- IPARS returns the normalized revenue value to VFSA Optimization

---

**Autonomic Optimization Process (2)**

- IF checks for revenue stabilization
- IF sends optimized guess to OS
- OS generates optimized guess
- OS sends guess to IF
- IF checks for revenue stabilization
- IF repeats until revenue stabilizes

---

**Scenario: Production Run for Monitoring and Steering**

- Experts use client portals to collaboratively connect to the running application for monitoring and steering
Collaborative Monitoring and Steering

IPARS Factory VFSA Optimization service

IF starts IPARS instance using optimal well parameters
IPARS connects to DISCOVER Middleware exporting interfaces and Objects for monitoring and steering

Clients connect to DISCOVER Middleware for collaboration and application control

Sample Results

Conclusions and Future Work

- Pawn is a P2P middleware suitable for autonomic-oriented applications
- Presented its high-level design requirements in terms of services and mechanisms
- Presented its use in the oil reservoir optimization process
- Current and Future work include the integration and extension of Pawn with the other components of Project AutoMate to enable fully distributed, autonomic, and secure interactions.
Web Links

- Pawn’s web page
  - http://www.caip.rutgers.edu/~vincentm/PAWN
- Project JXTA
  - http://www.jxta.org
- O’Reilly p2p web site
- Brendon Wilson’s book on JXTA
  - http://www.brendonwilson.com

References

- DISCOVER: A Web-based Computational Collaboratory for Interaction and Steering
- A Study of Discovery Mechanisms for Peer-to-Peer Applications
- AutoMate: Enabling Autonomic Grid Applications
- Autonomic Optimization of an Oil Reservoir using Decentralized Services
  V. Matossian and M. Parashar; To Appear in CLADE 2003, 2003
- Enabling Peer-to-Peer Interactions for Scientific Applications on the Grid
  V. Matossian and M. Parashar; To Appear in proceedings for EuroPar 2003
Appendix A: Pawn Components

- **Client Peer:**
  - Deploy applications for monitoring and steering
  - Collaborate with other peers
- **Rendezvous Peer**
  - All peers are connected to rendezvous for discovery.
  - Rendezvous cache messages.
  - Dynamic message aggregation
- **Application Peer**
  - Provides an interface to the application controls
  - May act as a proxy for relaying queries and responses

Appendix B: Pawn: AMS, AEX, Group communication Services

- Build on JXTA’s Resolver service
  - XML Messages contain:
    - destination, source, application id, queryID, queryType, unique Handler name
  - Reliability provided by caching
- ARC provides API to:
  - announceApplication
  - sendAppResponse
  - publishUpdateMessage
  - notifyEndApplication
- AMS provides API for:
  - zeroAppRequest
Appendix B: Pawn: ARC and RMC Services

- RMC builds on non-blocking JXTA pipes
- Defines an XML interface to the remote method call
- Uses message queues for ordering
- Messages carry unique identifiers to maintain consistent, coordinated application events