Building and Monitoring a Transactional Data Grid
Today’s Presenters

AJ Brown
Co-Founder and CMO for CloudTran
www.CloudTran.com

Dave Felcey
Oracle Coherence Product Manager
www.Oracle.com

Everett Williams
Director of SL Corporation
www.SL.com
Today’s Webinar

- Oracle Coherence – history and key benefits of an in-memory data grid
- Shifting to a managed data layer architecture
- Key issues in using an in-memory data grid for transactions
- Introducing the CloudTran transaction manager for Oracle Coherence
  - what it does
  - edge cases worth noting
  - when to use it versus not
- Introducing SL Corporation
- Real-time monitoring and why it’s important
  - the complexity of distributed systems
  - effective resource utilization
- Demonstration
- Questions & answers
A Brief Introduction to Oracle Coherence

Dave Felcey
Coherence Product Management
Oracle Fusion Middleware
Complete, Open, Integrated, Best-in-Class

- Web
- Mobile
- Social

- User Engagement
- Content Management
- Business Intelligence
- Business Process Management
- Service Integration
- Data Integration
- Cloud Application Foundation

- Development Tools

- Identity Management
- Enterprise Management

Copyright (c) Oracle Corporation 2012 - All Rights Reserved
Oracle Coherence In-Memory Data Grid

Application Infrastructure for Scalability

- Java application that stores key/value pairs in memory as caches or Maps
- Coherence JVM’s communicate to provide fast, consistent and reliable access to data
- It distributes cache data across JVM’s – and servers
- Enables scalable and very fast processing inside caches – like stored procedures
- Provides real-time eventing, query, and map/reduce aggregations
- Back-end data source offload
- Dynamically scalable platform with seamless failover and recovery for data and processing
Using Oracle Coherence

Scalable Distributed Caching for packaged, database orientated and real-time event driven applications

<table>
<thead>
<tr>
<th>Toplink Grid</th>
<th>IIS/J2EE HTTP Session</th>
<th>Java / C++ / .NET</th>
</tr>
</thead>
<tbody>
<tr>
<td>JPA API</td>
<td>Map API</td>
<td>REST (Python, PHP, ...)</td>
</tr>
<tr>
<td></td>
<td>Query Filter API</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Aggregation (Map-Reduce) API</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Grid Processing API</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Real-Time Client API</td>
<td></td>
</tr>
</tbody>
</table>

Coherence Cluster (Java)

Per Member Cache Store/Loader API

Replication

Coherence Cluster (Java)

Data Source
For More Information

- General Information: http://coherence.oracle.com
- Coherence YouTube Channel: http://www.youtube.com/user/OracleCoherence
- Coherence Training: http://education.oracle.com
- Coherence Discussion Forum: http://forums.oracle.com
- Coherence User Group on Linkedin
- “Oracle Coherence 3.5” by Aleks Seovic
- My email: craig.blitz@oracle.com
Data Grids as the New System of Record = No Bottlenecks

**Current Web App Architecture**

- **Application**
  - Local Data
  - Shared Data

**Managed Data Layer Architecture**

- **Application**
  - Local Data
  - Shared Data – In memory and on disk

**Issues**

- Limited scalability resulting in performance degradation due to single entry point to shared data
- High latency when accessing shared data
- Underutilization of surplus equipment
- Data replication may be required to meet service level requirements for performance of enterprise apps

**Benefits**

- Lower operating costs: use commodity hardware for scale-out of data, memory, and/or application processing
- Future growth: highly scalable while maintaining predictable performance
- High throughput: single shared data source can accommodate all enterprise apps
Key Issue for Data Grids: Distributed Transactions

Transaction: Add customer order
- Read customer information
- Insert parent order record, generate primary key
- Insert each child line item
- Update inventory for each item

Application

Distributed Transaction

Coherence Grid Manager

Order Data
Customer Data
Inventory Data

Item Data
Key Issue for Data Grids: Distributed Transactions

**Transaction:** Add customer order
- Read customer information
- Insert parent order record, generate primary key
- Insert each child line item
- Update inventory for each item

---

**Coherence Grid Manager**

**Persistence Service**

**Distributed Transaction**

**Application**

**Inventory**

**Data**

Coherence Grid Manager

Order Data

Customer Data

Inventory Data

Persistence Service

Customer Profile Database

Order Database

Inventory Database

CloudTran
Introducing CloudTran: A Transaction Manager for Data Grids
Engineered to Handle the Expected as Well as Edge Cases
### When You Need CloudTran

<table>
<thead>
<tr>
<th>Coherence</th>
<th>Coherence + CloudTran</th>
</tr>
</thead>
<tbody>
<tr>
<td>All data participating in a transaction is in a single node</td>
<td>Data in transactions can be spread over 2 or more nodes</td>
</tr>
<tr>
<td><img src="" alt="Diagram" /></td>
<td><img src="" alt="Diagram" /></td>
</tr>
<tr>
<td>Or, for data using 2+ nodes, only execute transactions that can be re-run until successful</td>
<td>Any business transaction is supported with full roll-back recovery, if necessary</td>
</tr>
<tr>
<td>&quot;Delete all history for accounts more than 30 days past due and update status of account to CLOSED”</td>
<td>&quot;Distribute $100 from account A to account B and C”</td>
</tr>
<tr>
<td>Can engineer persistence software for various backend databases or persistence isn’t required</td>
<td>Data must be persisted to permanent storage within scope of client transaction</td>
</tr>
</tbody>
</table>

---

**CloudTran**
Build a Transactional Data Grid with CloudTran

CloudTran
Transaction Manager for Coherence

- Scalable, distributed transactions spanning multiple cache nodes and databases (without 2-phase commit)
- Asynchronous write-behind persistence with full ACID properties
  - Predictable performance for transactions
    - Lights-out recovery of the data grid
    - Transactional data center replication
Introducing SL Corporation

- Real-Time Application Monitoring
- Visibility into Health State of Critical Applications
  - Are all components working properly?
  - Are they performing fast enough?
  - How do I fix it if something goes wrong?
- Specializing around Middleware and Data Grids
  - Experience with Coherence since 2006 when still Tangosol
  - Highly Tuned and Optimized Oracle Coherence Monitor
Why Should You Care?

+ Business relies 24x7 on availability and performance of critical applications
  - Confidence and Validation
  - Troubleshooting
  - Ready-For-Business Checks

+ Distributed applications especially complex
  - Components like Coherence new and difficult to understand
  - Multiple components interacting with each other and dependent on one another
Coherence as a Black Box

Distributed Applications

Coherence

BLACK BOX

Is it working?

Is it fast enough?

How to fix it?
Monitoring Architecture

RTView for Coherence Architecture - Conceptual View

- Collect Data
- Process Events
- Perform Corrective Actions
- Correlate
- Transform
- Analyze
- Real-Time Data
- In-Memory Data Model and Cache
- Configure and Customize
- Visualize and Interact
- Report
- Alert
- Archive
- Dashboards
- PDF/Excel Reports
- Notifications
- Performance Management Database

Coherence Applications
- Application Tier
- Oracle Coherence Cluster
- Hardware OS Tier

CloudTran
RTView OCM Solution for Application/Coherence Monitoring

Caching in an SOA Environment

- Composite Applications
- Business Processes
- Business Services
- Cache
- Data Services
- Datasources: Database, Mainframe, Web Service

Unified Real-time display of data from all Application tiers

In-depth Monitoring of Coherence infrastructure
Collect / Analyze / Process Coherence
Monitoring Data with RTView OCM

RTView for Coherence - Simple Deployment

RTView Oracle Coherence Monitor connects through a firewall to a Cluster using a JMX port or RMI connection (could be authenticated)

Advantages:
1) NOTHING to install behind firewall
2) Simple, uses existing MBean server
Collect / Analyze / Process Coherence Monitoring Data with RTView OCM

RTView for Coherence - Performance Deployment

RTView Oracle Coherence Monitor Agent joins cluster as MBean Server. Data are processed in a local RTView Data Server and served on demand.

Advantages:
1) Fast, local transfer of raw data
2) Local data reduction and aggregation
3) Alerts generated close to the source
4) On-demand data minimizes network usage

Variations on this configuration are possible. E.g. The history module could be deployed outside the firewall as well.
Develop Custom Monitoring Views using RTView Builder

Rapid display and GUI development

CloudTran developed data model and custom views for their application MBeans
Cloud Tran

Broad Range of Application Monitoring Customers

- OOCL World Wide Shipment Tracking
- Hospitality Card application at Harrah’s casino gaming tables
- Online Gaming Systems
- Tax Season at Intuit
- PJM Real-time Energy Pricing
- Banking application in Korea
RTView – High-level Overview of Application Health State

+ Typical large implementation, distributed over several regions with many custom applications

+ Heatmap View showing current state of entire system – size represents number of servers for application

+ Color represents how close metric is to SLA – large red boxes are worst – drilldown to detail
RTView – Custom Monitoring of Application Metrics

+ Drilldown to detail level metrics showing internal metrics from each application

+ Sophisticated history and alert view allows fine-tuning of thresholds for each metric
RTView Oracle Coherence Monitor enables monitoring and alerting on conditions within critical distributed caching infrastructure.
Video Demonstration

OCM and CloudTran

Correlating Application and Coherence Performance Metrics
Next Steps and Questions

- Read more at www.SL.com and www.CloudTran.com
- Identify a proof of concept project
- Download and try
  - 30-day trial of RTView from SL Corporation
  - Developer copy of CloudTran (no charge)
- Schedule a private briefing for more information