Virtual Developer Day
Oracle WebLogic Server 12c
Modern, Lightweight Development with Java EE 6 and Oracle Coherence

Pyounguk Cho, WebLogic Server Product Management
Supercharging WebLogic Server Applications with Coherence
The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle’s products remains at the sole discretion of Oracle.
Coherence and Cloud Application Foundation Overview
Oracle Cloud Application Foundation
Foundation for Oracle Fusion Middleware in the Cloud

Web

Social

Mobile

User Engagement

Business Process

Content

User Engagement

Identity Management & Security

Service Integration

Data Integration

Identity Management & Security

Development Tools

Cloud Application Foundation

Oracle Public Cloud

Traffic Director

WebLogic Server

Coherence

Tuxedo

Virtual Assembly Builder

Cloud Application Foundation

Exalogic Elastic Cloud
Oracle, Custom, and 3rd Party Apps
Runs on Cloud Application Foundation

- Best of Breed Standards Based Development
  - Driving Industry Standards
- High Performance Engineered Infrastructure
  - Industry’s Fastest on Conventional Hardware
  - And Optimized Even More for Exalogic and Exadata
- On Premise or in the Cloud
  - Virtualized, Elastic, Self Service
Oracle Coherence In-Memory Data Grid
Application Infrastructure for Scalability

- Distributed caching
- Scalable extreme transaction processing
- Real-time eventing, query, and map/reduce aggregations
- Back-end data source offload
- Reliable data tier providing high-availability to applications
Example WebLogic Coherence Data Grid Uses

• Caching
  – WebLogic and client applications request data objects and state from the Data Grid rather than backend data sources

• Analytics
  – WebLogic and client applications ask the Data Grid questions from simple queries to scenario modeling

• Compute
  – WebLogic and client applications execute compute – business logic and transactions - across the Data Grid

• Events
  – WebLogic and client applications take action based on events from the Data Grid
Meet SLAs as Usage Increases…

Hotel Search Response Time

- Search Response Without Oracle Coherence
- Expected Service Level Agreement
- Search Response With Oracle Coherence

Detailed Results and Source:
...While Protecting the Database

Detailed Results and Source:
Elastic Data
Virtual Memory for Your Data

- Revolutionizes the scale of data grids
- Use block storage for cached data
  - Optimized for flash
  - Works well with NAS, Disk, etc.
- Simplifies capacity planning and deployment
  - Configure amount of on-heap data to store
  - Overflow data written to block storage
  - Data stored in buffers until flushed to journal
  - Reduces chances of Out-Of-Memory errors
Coherence Exabus Optimizations

Direct Memory I/O for Java and C++

- Leverage new Java APIs and Exalogic Elastic Cloud Software
  - Low Latency support for Infiniband
  - Optimized implementation for Exalogic Infiniband
- Scalable to massively multi-core systems
- Surfacing low-level advanced networking capabilities

4x Throughput, 6x Better Response Time
Oracle Enterprise Pack for Eclipse (OEPE)

Developer Productivity

- Coherence Project Configuration
  - Facets
  - Library Management
  - Descriptor Generation

- Runtime Configuration
  - Launch Config Editor
  - Run/Deploy/Debug from IDE

Configuration Support
- coherence-cache.config.xml editor
  - Validation
  - Context Sensitive Help Integration
  - Cache Configuration Editor
Oracle Enterprise Manager 12C

Deep Performance Visibility & Alerts-Based Monitoring
Real-time JVM Diagnostics for Coherence Nodes*
Centralized Cache Data Management
Automated Provisioning

Applications and Business Services
Platform as a Service
DBaaS
MWaaS
Infrastructure as a Service

Plan
Build
Setup
Test
Deploy
Monitor
Manage
Meter & Charge
Optimize
Coherence Integration with WLS: Active Cache
WebLogic with Coherence in WebLogic Suite

Increase Capacity by Scaling Session, Caching and Data Grid Separately

Weblogic Server Cluster

In Memory Data Grid

WebLogic Server with Coherence

Supported Workload
(Sustained throughput: Requests per Second, identical hardware)
What is Active Cache?

• Integration of Coherence and WebLogic Server
  – Incremental progress since 11g
• The whole is greater than sum of its parts!
  – Coherence*Web SPI Support for HTTP Sessions
  – Dependency Injection
  – Configuration, Lifecycle and and Monitoring of Coherence Clusters and Servers inside WebLogic Server domain
  – TopLink Grid: Caching for JPA and write-behind
Active Cache Shared Libraries

This page displays a list of Java EE applications and stand-alone application modules that have been installed to this domain. Installed applications and modules can be started, stopped, updated (redeployed), or deleted from the domain by first selecting the application name and using the controls on this page.

To install a new application or module for deployment to targets in this domain, click the Install button.

### Deployments

<table>
<thead>
<tr>
<th>Name</th>
<th>State</th>
<th>Health</th>
<th>Type</th>
<th>Deployment Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>active-cache(1.0,1.0)</td>
<td>Active</td>
<td></td>
<td>Library</td>
<td>100</td>
</tr>
<tr>
<td>coherence(3.7.1.0,3.7.1.0)</td>
<td>Active</td>
<td></td>
<td>Library</td>
<td>100</td>
</tr>
<tr>
<td>coherence-web-spi(1.0.0.0,1.0.0.0)</td>
<td>Active</td>
<td></td>
<td>Library</td>
<td>100</td>
</tr>
</tbody>
</table>
public class CounterServlet extends HttpServlet {

    @Resource(mappedName = "dist-counter")
    private NamedCache counterCache;

    public void service(HttpServletRequest request) throws ...
    {
        // no need to instantiate counterCache
        Integer previousCount = (Integer) counterCache.get(COUNTER_KEY);
    }

    </web-app>
    <resource-ref>
        <res-ref-name>coherence/counterCache</res-ref-name>
        <res-type>com.tangosol.net.NamedCache</res-type>
        <mapped-name>dist-counter</mapped-name>
    </resource-ref>
</web-app>
Module Scoping and Coherence Services

Applications join clusters via module deployment descriptors:

```xml
<wls:coherence-cluster-ref>
  <wls:coherence-cluster-name>myCluster</wls:coherence-cluster-name>
</wls:coherence-cluster-ref>
```
Module Scoping and Coherence Services

- Visible to all Modules
- Visible to Web Module Only

Assembly Root (EAR)

APP-INF/classes/coherence-cache-config.xml

Web Module

EJB Module/coherence-cache-config.xml

Resource Adapter

WEB-INF/classes/coherence-cache-config.xml
Manageability

- **Coherence**
  - Logically named, controlled by Admin Server Runtime MBean via Node Manager
  - Lifecycle operations e.g. start/stop
  - Bound to Machines defined in WLS configuration
  - A JVM running a subclass of DefaultCacheServer that accepts WLS Domain Configuration:
    - `weblogic.nodemanager.server.provider.WeblogicCacheServer.class`
  - Customizable configuration
  - provisioning via pack/unpack
- **WebLogic Tooling support**
  - Domain Configuration Mbeans for Coherence Cluster configuration and Coherence Server definitions
  - Domain Admin Server Runtime Mbeans for Coherence Server lifecycle
  - Node Manager Manages Lifecycle and High Availability
  - WLST Scripting can interact with the MBeans
Coherence*Web

• **What**: Distributed HTTP Session Management
  – Span Applications
  – Span Heterogeneous Environments
  – Handle Large Sessions

• **Why**:
  – Decouple session management from web container
  – Handle more users without adding more application servers
  – Restart/maintain applications/containers without losing sessions
  – Handle very large sessions efficiently
  – Keep session data coherent under heavy load
Coherence*Web

Deployment Model

- **In-Process**
  - Session state maintained within app server process
  - Default model for ease of demonstration
  - **DO NOT USE IN PRODUCTION**

- **Out-Of-Process**
  - Two tiers – app server and cache server
  - Session data storage is offloaded from the app server tier
  - Each tier can be scaled independently
  - Default recommendation due to flexibility

- **Out-Of-Process-Extend**
  - Communication between tiers is over Coherence*Extend (i.e. TCP/IP)
  - Ideal in environments where network does not support UDP
Coherence*Web

Session Model

• Traditional
  – Manages each session in a single Coherence cache entry
  – Manages each session attribute’s (de)serialization separately

• Monolithic
  – (de)serializes all session attributes together in a single object stream
  – Solves the shared object issue

• Split (Recommended)
  – Session meta-data and “small” attributes stored in one cache
  – “Large” attributes stored in a separate cache
  – Easily supports very large session objects
  – Leverages near caching for “small” attributes
Coherence*Web
Locking Mode

• Optimistic Locking (default)
  Allows multiple nodes in a cluster to access an HTTP session simultaneously. Concurrent updates are detected and rejected using an optimistic approach.

• Member Locking
  Does not allow more than one node in the cluster to access an HTTP session.

• Application
  Does not allow more than one application in a single JVM to access an HTTP session.

• Thread Locking
  Does not allow more than one thread in the cluster to access an HTTP session.
Coherence*Web

Cluster Node Isolation

- **Application-Server Scoped**
  - All deployed applications in each app server instance will be part of one Coherence node
  - Smallest number of Coherence nodes
  - Smallest resource utilization (only one copy of the Coherence classes loaded per JVM)

- **EAR Scoped**
  - All deployed applications within each EAR will be part of one Coherence node
  - Reduces the deployment effort as no changes to the application server classpath are required

- **WAR Scoped**
  - Each deployed web application will be its own Coherence node
  - Largest number of Coherence nodes
  - Largest resource utilization (only one copy of the Coherence classes loaded per deployed WAR)
Coherence*Web
Session and Session Attribute Scoping

• Session Scoping
  – Coherence*Web allows session data to be shared by different Web applications deployed in the same or different Web containers.

• Session Attribute Scoping
  – Extension of Session Scoping allowing for scoping of individual session attributes so that they are either globally visible or scoped to an individual web application.
  – Behavior is controllable via the AttributeScopeController interface. Two out of the box implementations:
    – ApplicationScopeController and GlobalScopeController
TopLink Grid: Coherence Optimized Persistence

- Transparently harness power of Coherence
- TopLink Grid combines:
  - Simplicity of application development using standard Java Persistence API (JPA)
  - Scalability and distributed processing power of Oracle’s Coherence Data Grid.
- Supports 'JPA on the Grid' Architecture
  - EclipseLink JPA applications using Coherence as a shared (L2) cache replacement
  - Uses data grid to process queries to avoid database access and decrease database load
GEICO Insurance

Profile

- 4\textsuperscript{th} largest private auto insurance company
- 3\textsuperscript{rd} largest P&C insurer in US

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Problem</th>
<th>Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>- One of the largest on-line insurance providers in the US with self-service website for customers</td>
<td>- Large Database was heavily loaded by persistence of enormous user profiles (&gt;1MB each) for thousands of concurrent users</td>
<td>- Oracle Coherence allows all customer data to be managed in-memory for fast access to user profiles</td>
</tr>
<tr>
<td>- Scaled out application tier built to handle high volume of traffic</td>
<td>- Challenge expanding environment for spikes in usage or additional services offered</td>
<td>- Updates to profiles are actively held in the Data Grid and only persisted to the database once, at the end of the user session</td>
</tr>
<tr>
<td></td>
<td>- Over 10X increase in application tier capacity</td>
<td></td>
</tr>
</tbody>
</table>
# Coherence Integrations

## Generally Available
- Oracle IGBU
- Oracle CGBU
- Oracle GlassFish
- Oracle Access Manager
- Oracle CEP
- Real-time Decision
- TopLink
- WebLogic Portal
- Service Delivery Platform
- WebLogic Server
- Oracle Service Bus
- SOA Infrastructure
- Oracle Data Integration

## Near Term
- Oracle Golden Gate
- Meta-data Services (MDS)
- Oracle ADF
- Oracle PeopleSoft
- Oracle Secure Token Service
- Oracle Identity Federation
- Oracle

## Medium/Long Term
- WebCenter
- SOA BPEL
- SOA Human Workflow
- SOA Business Rules
- SOA Coherence Adapter
- SOA Technology Adapters
- BI Publisher
- Oracle Entitlement Services
- Oracle BEAM
- Oracle Web Services Manager
- ATG Hosted Services

### Advanced Capabilities
- Data Grid
- Data Cache
- Simple Clustering
For More Information

- General Information: [http://coherence.oracle.com](http://coherence.oracle.com)
- Coherence YouTube Channel: [http://www.youtube.com/user/OracleCoherence](http://www.youtube.com/user/OracleCoherence)
- Coherence Training: [http://education.oracle.com](http://education.oracle.com)
- Coherence Discussion Forum: [http://forums.oracle.com](http://forums.oracle.com)
- Coherence User Group on Linkedin
- “Oracle Coherence 3.5” by Aleks Seovic
- Email:
  - craig.blitz@oracle.com,
  - pyounguk.cho@oracle.com
Join the Coherence Community

Oracle.com/coherence

Twitter
Twitter.com/OracleCoherence

Facebook
Facebook.com/OracleCoherence

Oracle’s WebLogic blog
Blogs.oracle.com/OracleCoherence

Youtube
YouTube.com/user/OracleCoherence

Local Special Interest Group (SIG) Meetups
Coherence.oracle.com/display/CSIG/Home