

ORACLE®

Oracle Coherence 12c

Strategy And Roadmap for Coherence Special Interest Group

Craig Blitz, Director of Product Management
Cloud Application Foundation



Safe Harbor Statement

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.

Program Agenda

 Introduction to Coherence & the 12c Release

Coherence Roadmap

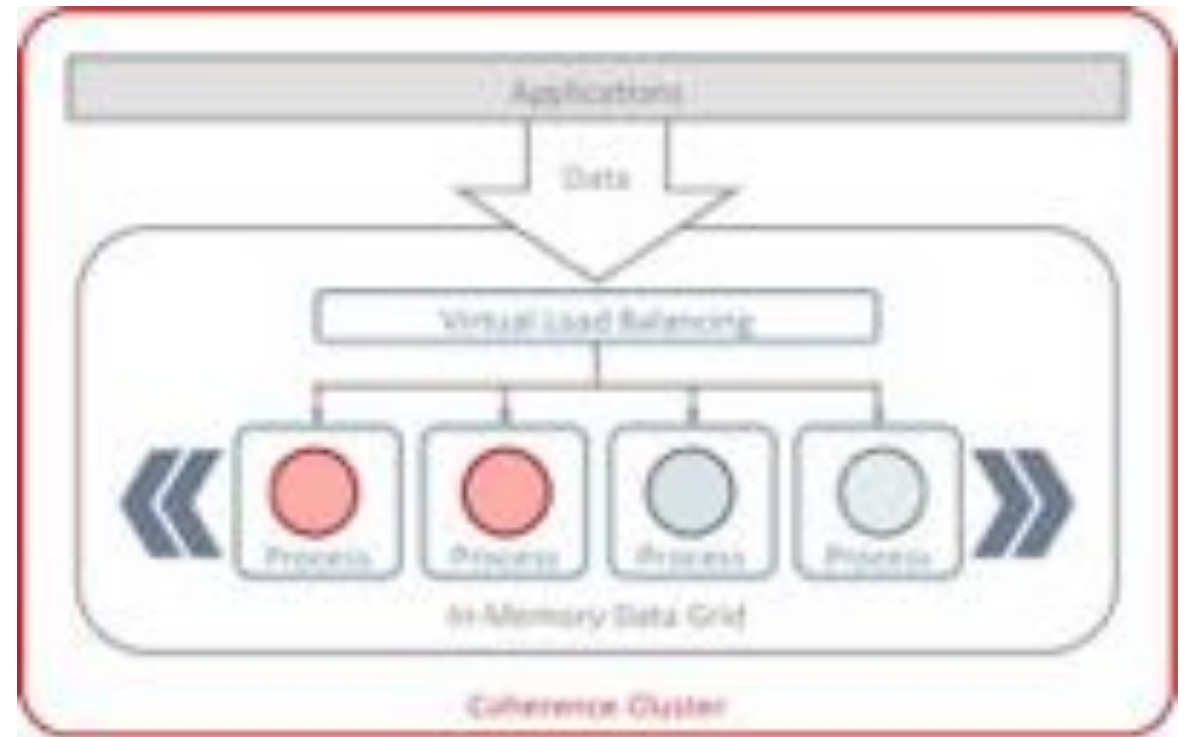
Customer Use Cases

Oracle Java Cloud Service

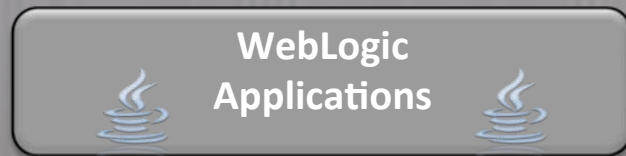
Coherence In-Memory Data Grid Overview

Scalable, Fault-Tolerant Application Infrastructure

- Reliable In-Memory Key-Value Store
- Dynamically Scalable
- Scale processing with data
- Java, .NET, C++, REST, Memcached, Jcache clients
 - **JCache and Memcached NEW in 12.1.3**
- Entries can be
 - Reliably processed in-place
 - Queried
 - Aggregated
- Rich Live Event Programming model
- Data source integration



Benefits of Using Coherence with WebLogic



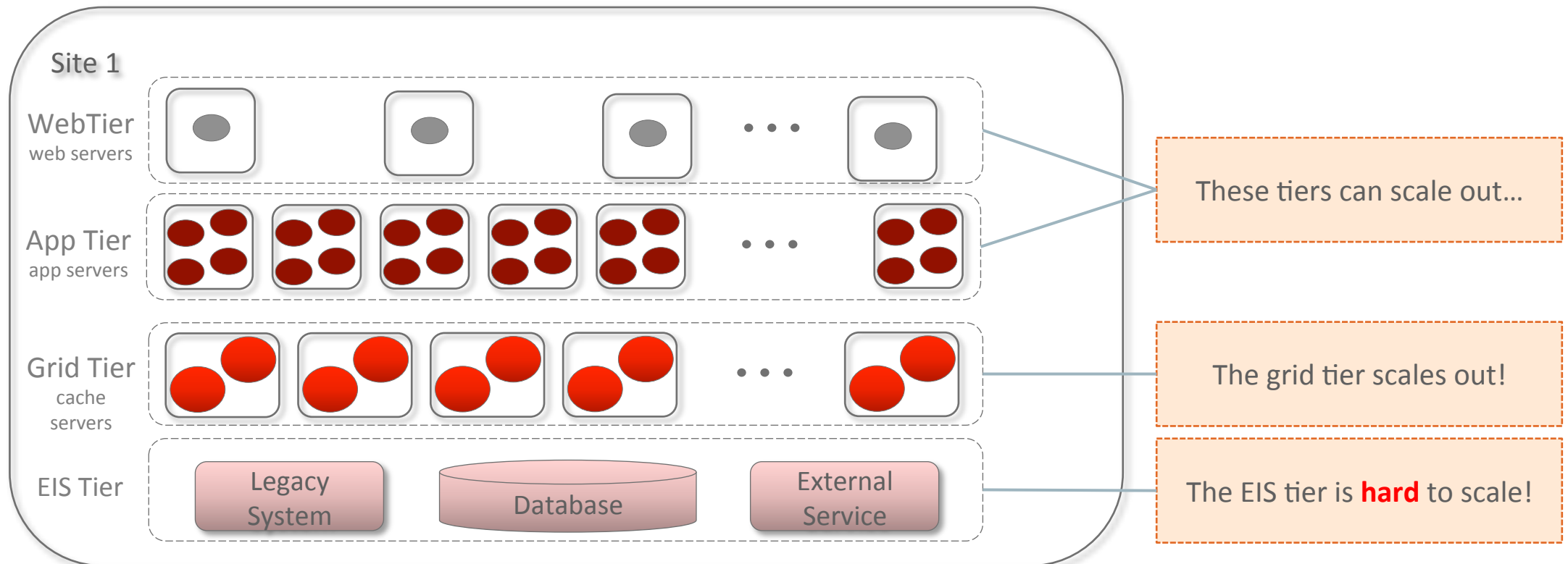
Scale, Increase Performance &
Improve Reliability of Middleware with
Peer to Peer, High Availability
In Memory Data Grid



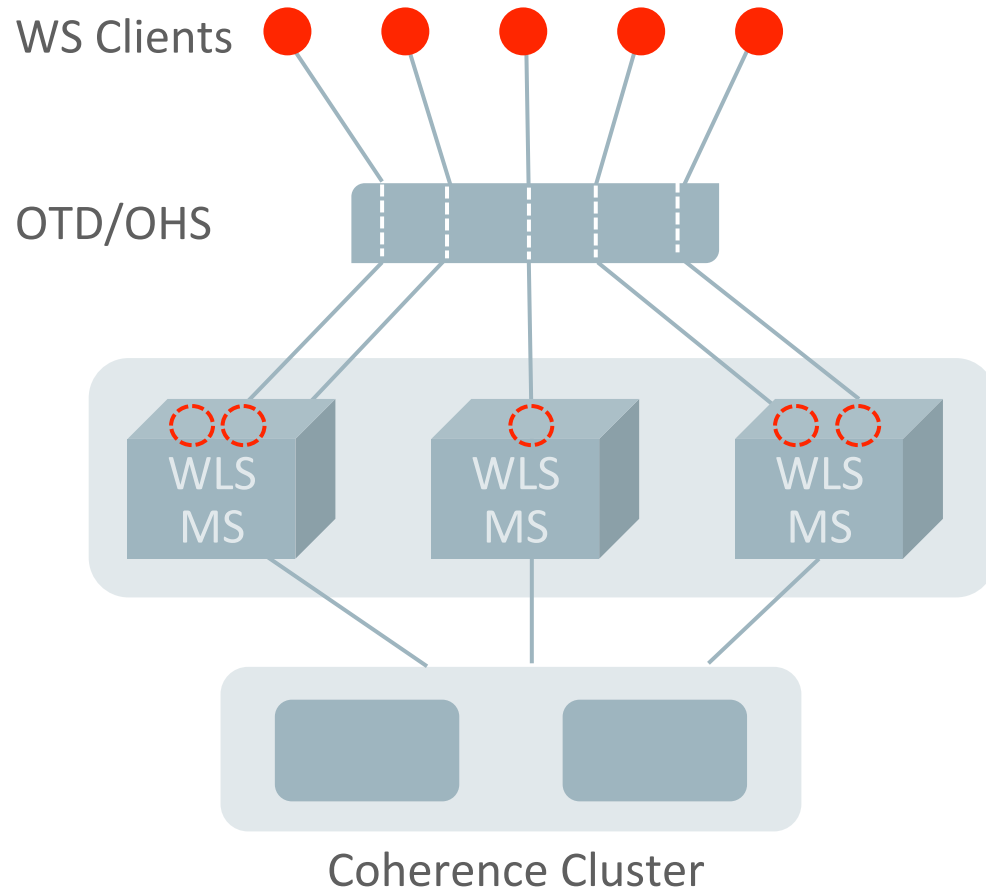
- **Increases Application Performance**
 - In-memory data access for middleware applications – Application Objects in memory
- **Increases Application Scalability**
 - Caching at middleware tier reduces backend workloads – DB, mainframe, web services
 - More than distributed caching: query, compute, map/reduce, and events on data grid
- **Increases Application Reliability**
 - Clustered application state and data management: peer-to-peer grid, 1000's of nodes, terabytes of data

Customer Example

Java EE Application Physical Tiering - and Scalability



WebSocket Clustering and High-Availability

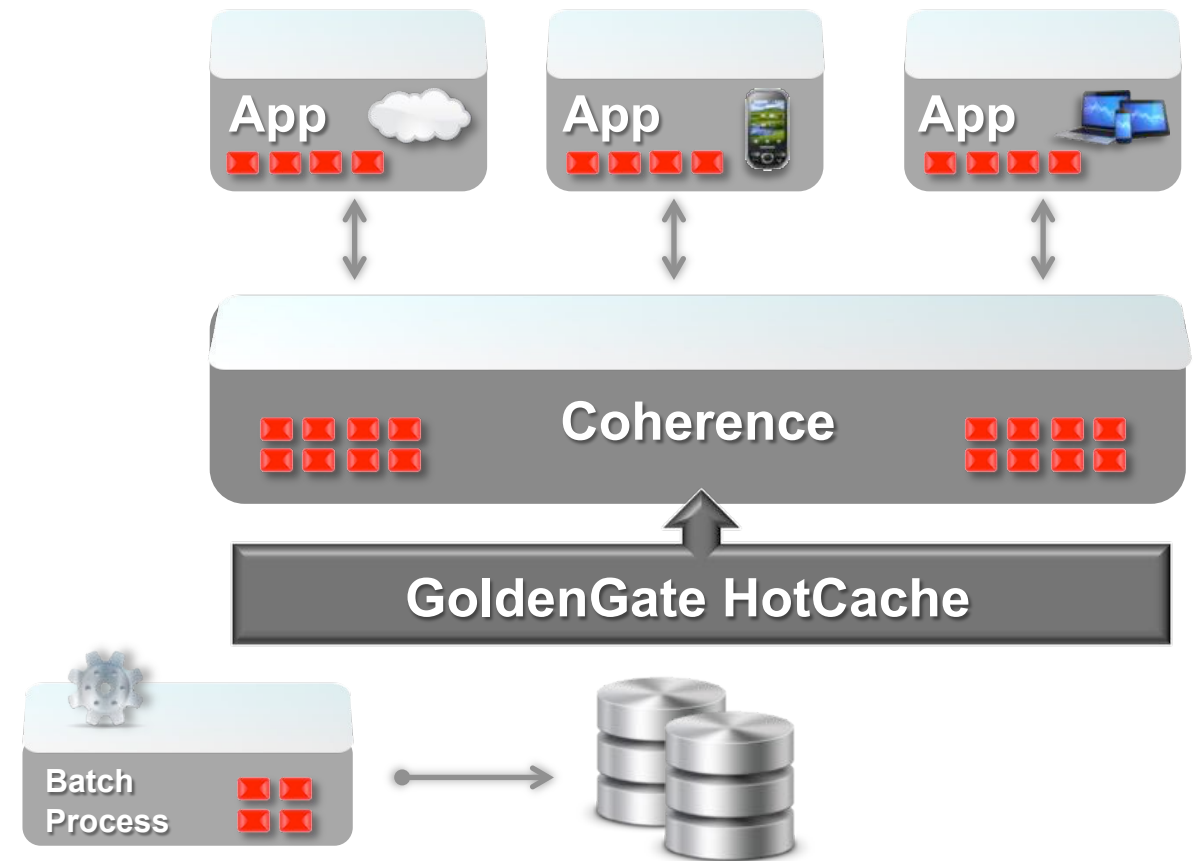


- JSR-356 has no direct cluster or HA requirement
 - Connection bound to local servers and local sessions
 - Multi-node session visibility, coordination
 - Recovery of state on connection failure and reconnect
- Coherence Cluster used as backing framework
 - Several NamedCaches employed to represent Endpoints, Sessions, Messages, Broadcast, DistributedProperties
 - Each distributed operation is deconstructed to a `Map.put()` with corresponding `MapListener` invocation
 - Endpoint can recover `userProperties` map on client reconnections

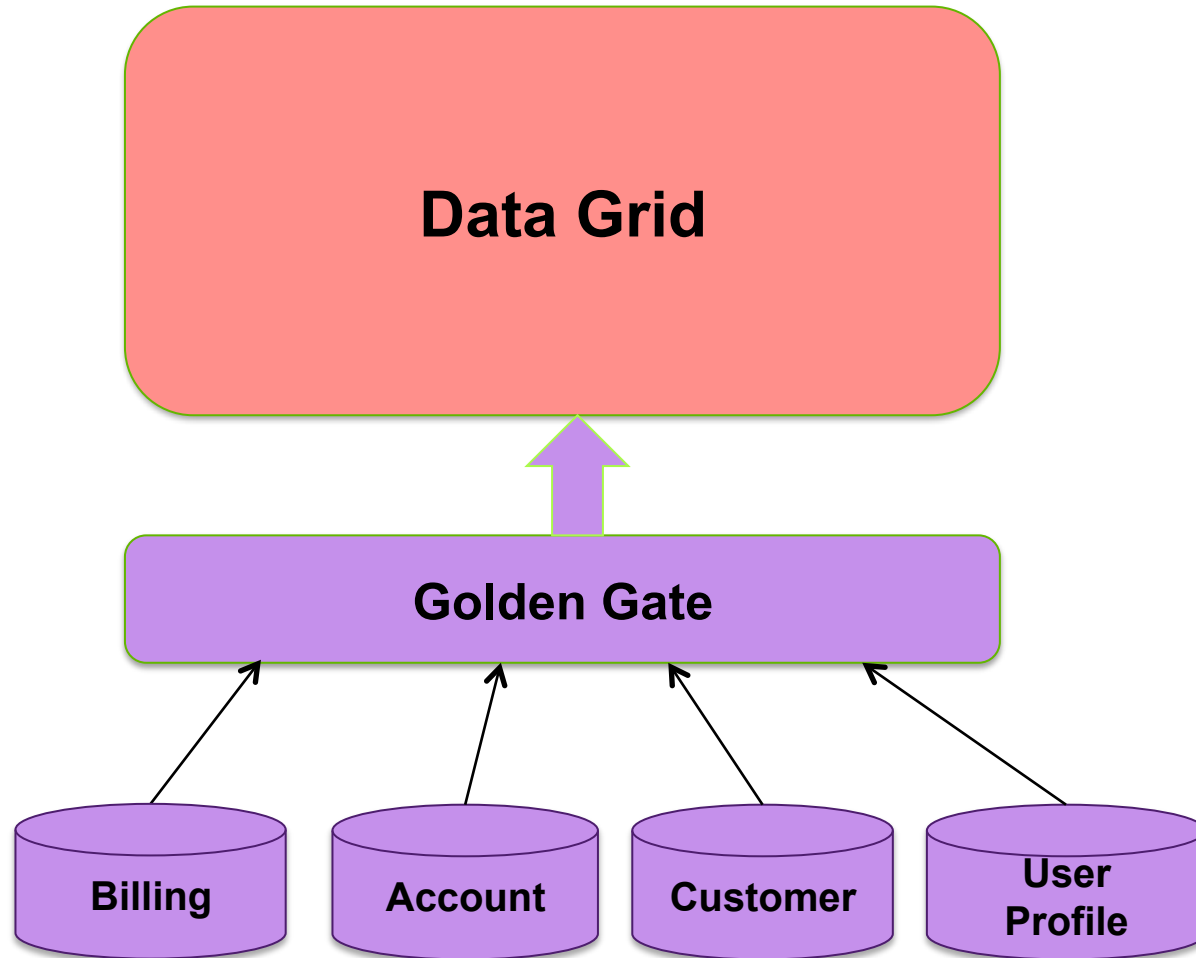
Coherence GoldenGate HotCache

Real Time Database Updates for Your Apps

- Detect and reflect database changes in cache in real time
- Leverage existing technologies
 - GoldenGate, TopLink Grid
- Broaden applicability/usability of Coherence
- No code change



Data Consolidation



Benefits:

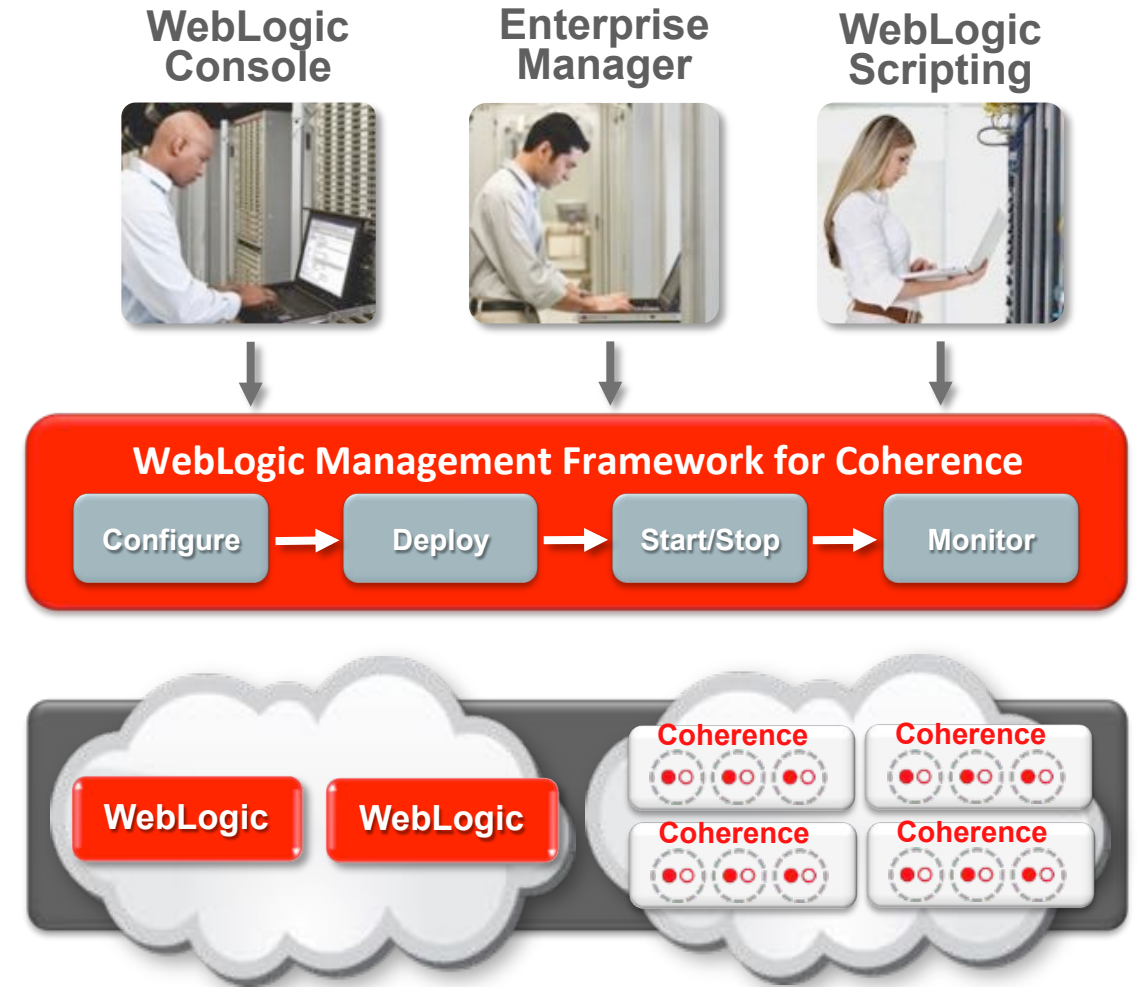
- Reduce data roundtrips
- Improve performance
- Less dependency on legacy data centers
- Canonical model across multiple source databases



Managed Coherence Servers

Administrative and Operational Efficiency

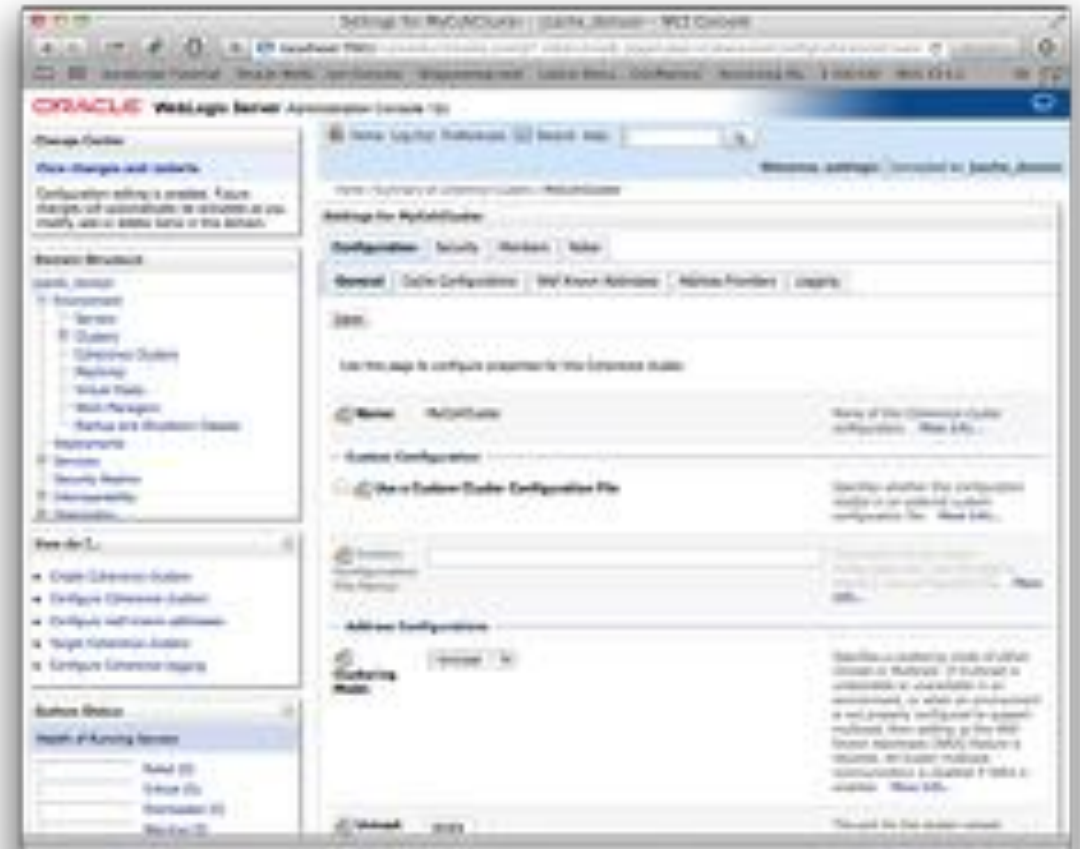
- Combined WebLogic and Coherence Infrastructure
 - WebLogic Management Framework
 - No Extra License Cost for Coherence Users
 - Configuration Wizard, WebLogic admin console, WLST, Node Manager
- Introduces the Grid Archive (GAR)
 - Package and Deploy
- Coherence “standalone” includes support for GARs



Coherence and WebLogic 12.1.2 Integration

WebLogic Management Framework for Coherence

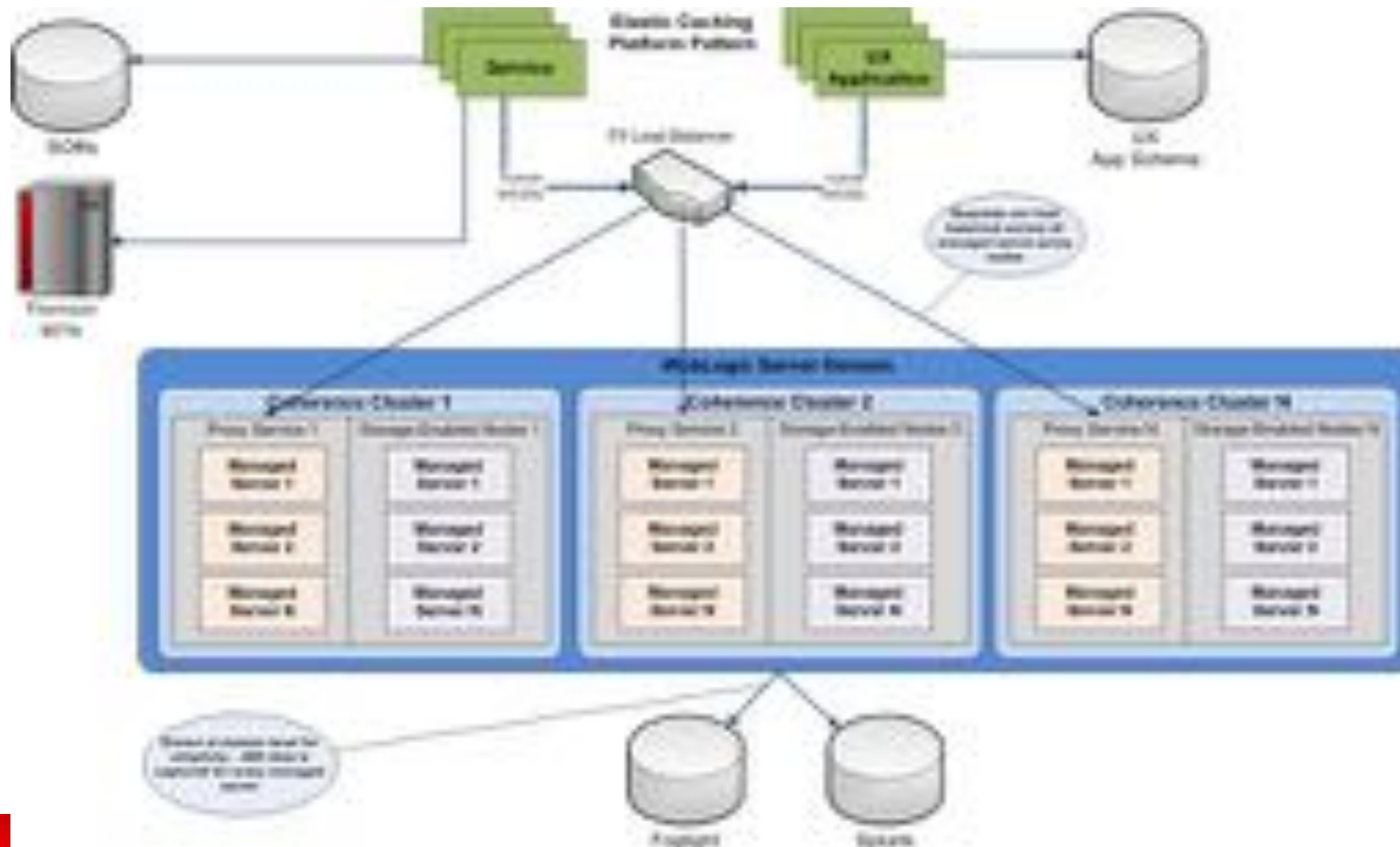
- Operational Configuration and Management
 - Key cluster parameters and security artefacts
 - Configuration wizard and domain templates
 - WLS cluster level “storage”, rolling-restarts scripts etc.
- Application Lifecycle Management
 - Application packaging and isolation using GAR
- Consistent Development Experience
 - Support added for Maven, ANT and WLST
- Seamless Support for Coherence*Web
- Available through WLS and the Admin Console



Coherence and WebLogic 12.1.3 Enhancements

- Extended Coherence Management
 - Added option to specify a Managed Coherence Server is a management node
 - Provided easier integration with Oracle Cloud Control
- Enhanced Coherence Extend Security Support
 - Removed the requirement that extend client use a WebLogic Subject when authenticating themselves with a Managed Coherence Server running a proxy service

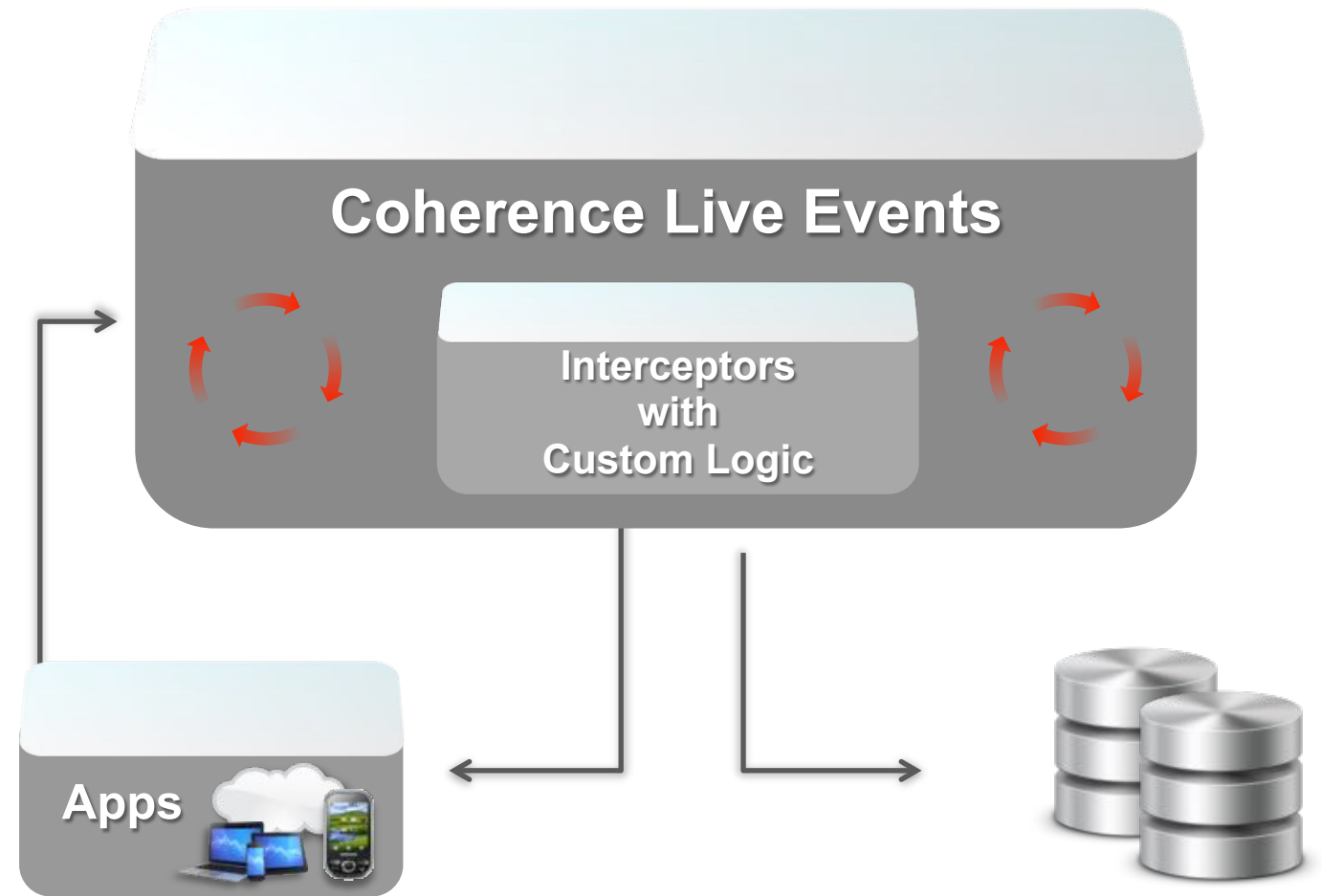
Customer Example: Leveraging MCS for Coherence as a Service



Coherence Live Events

Event-Driven Architecture

- One programming model for all events
 - Triggers, Backing Map Listeners, Partition Listeners
- Formalizes programming semantics for event driven architectures
- Declarative configuration



Asynchronous EntryProcessors

- Submit work asynchronously
- Save client resources
 - No more fork/invoke
 - Avoid hitting system limits
- Submit work quickly
 - Without waiting for response
 - Not one at a time
 - Order honored during rebalancing
- Protection against aggressive clients

Simple Invocation

```
AsynchronousProcessor procAsync =  
    new AsynchronousProcessor(  
        new NumberIncrementor(  
            (ValueManipulator)null, 1, false));  
  
cache.invoke(0, procAsync);  
procAsync.get();
```

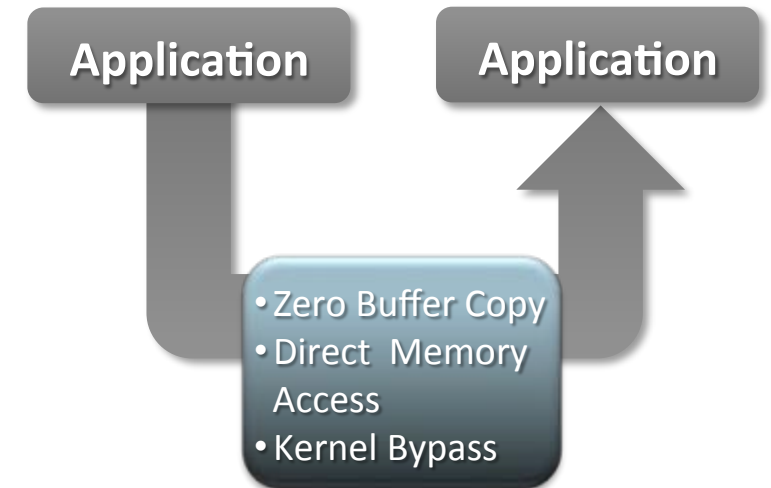
Invocation with Callback

```
final Entry[] aEntry = new Entry[1];  
AsynchronousProcessor procAsync = new  
    AsynchronousProcessor(new  
        NumberIncrementor((ValueManipulator)  
            null, 1, false))  
    {  
        @Override  
        public synchronized void  
            onResult(Entry entry)  
        { aEntry[0] = entry; }  
        @Override  
        public void onComplete()  
        {  
            Object oMonitor =  
                AsynchronousProcessorTests.this;  
            synchronized (oMonitor)  
            { oMonitor.notify(); }  
        }  
    };  
  
cache.invoke(0, procAsync);  
  
// call back when result received  
while (aEntry[0] == null)  
{  
    synchronized (this)  
    { wait(500) }  
}
```


Coherence On Exalogic Improvements

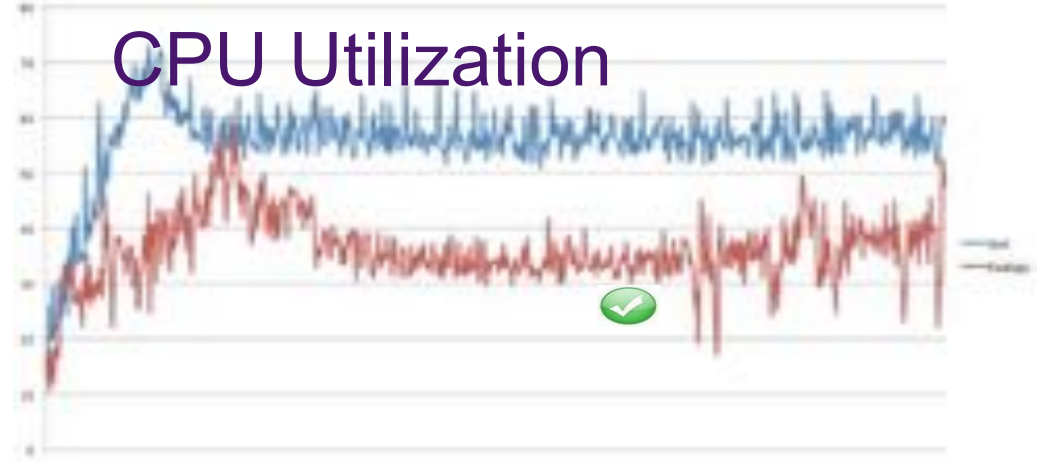
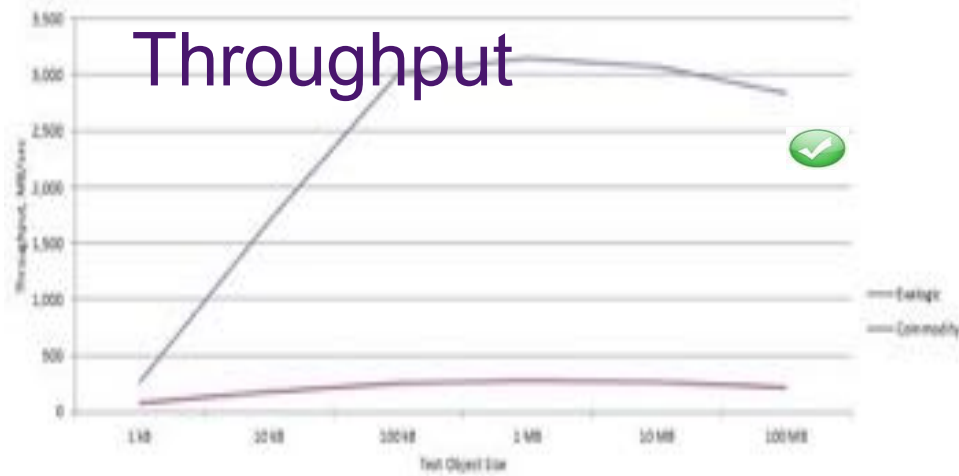
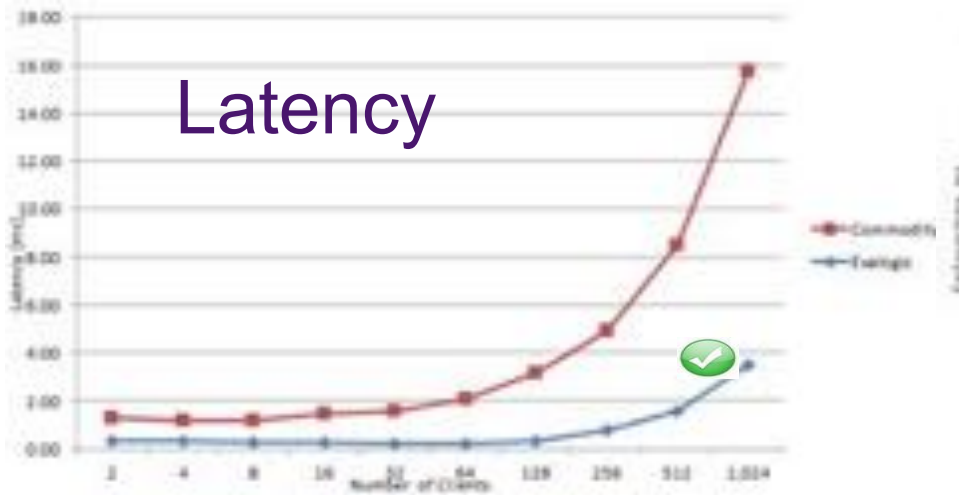
Infiniband Message Bus 2.0: Lightweight Message Bus

- Second generation implementation of the Infiniband MessageBus API for Exalogic
- MQL libraries shared with database and eventually other products
 - Leverages support and tuning investment across ExaData and ExaLogic platforms, and other product suites
- Greater latency improvements at scale
- Robust handling of component failure



4X Throughput, 6X Lower Latency, 16x Recovery Time, 2x Density

Data Grid Server - Exalogic vs Commodity



Full-Lifecycle Monitoring and Management



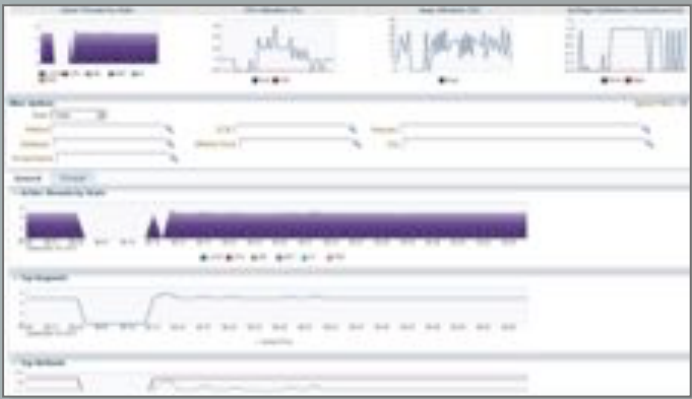
JVisualVM Plugin

- Available now for 3.x on Coherence Community Website
- Lightweight plugin to JVM



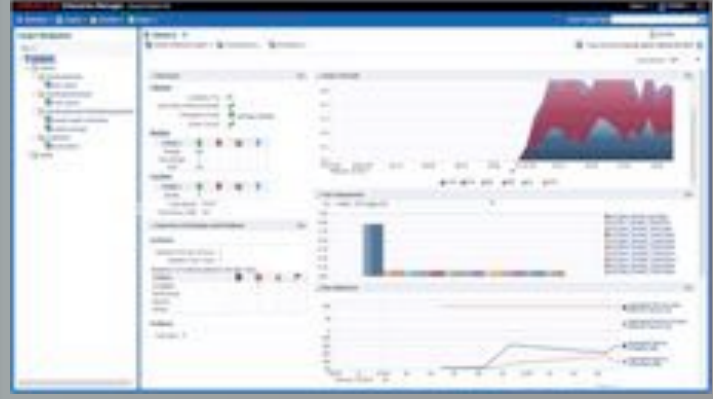
FMW Control

- OOTB administration and monitoring for all FMW
- Dev/QA point-in-time insight into cluster



Coherence Management Pack for OEM

- Complete management and monitoring solution
- Store historical results
- Java diagnostics tooling



Development Standards & Community

Strategic Integration and Participation for Cloud Application Foundation

Standards-Based for
Easy Adoption



HTML 5, Websockets, JCache
GitHub, REST, Maven...

Community Projects
for Ongoing Innovation



Spring, Eclipse, Hibernate,
Java.net, Incubators, More...

Program Agenda

Introduction to Coherence & the 12c Release

 Coherence Roadmap

Customer Use Cases

Oracle Java Cloud Service

Oracle Coherence 12c Roadmap

2013

12.1.2

(July, 2013)

- Managed Coherence Servers
- GoldenGate HotCache
- Live Events
- Configuration Modernization
- Asynchronous Backups
- Improved Backup Management
- Maven Support
- Exalogic optimizations
- Dynamic Proxy Thread Pool Tuning
- REST Improvements
- OUI/Opatch Integration

Community @ Java.Net

- Coherence Incubator 12 (12.1.2)
- Coherence Spring Integration
- Coherence Hibernate 4 L2 Cache

2014

12.1.3

(June, 2014)

- JSR 107/JCache
- Memcached Protocol Support
- VisualVM Plugin
- Exalogic IMB 2.0
- Asynchronous EntryProcessors
- Java 8 Runtime Support

2015

12.2.1

- Multitenancy
- Recoverable Caching
- Federated Caching
- Authorization/Audit Improvements
- Oracle Fusion Middleware Control
- Managed Coherence Servers 2.0
- Elastic Data Improvements
- Java 8 Developer Feature Support
- Generics Support

Coherence 12.2.1

General Availability: 2015

12cR2

Maximum Availability Architecture

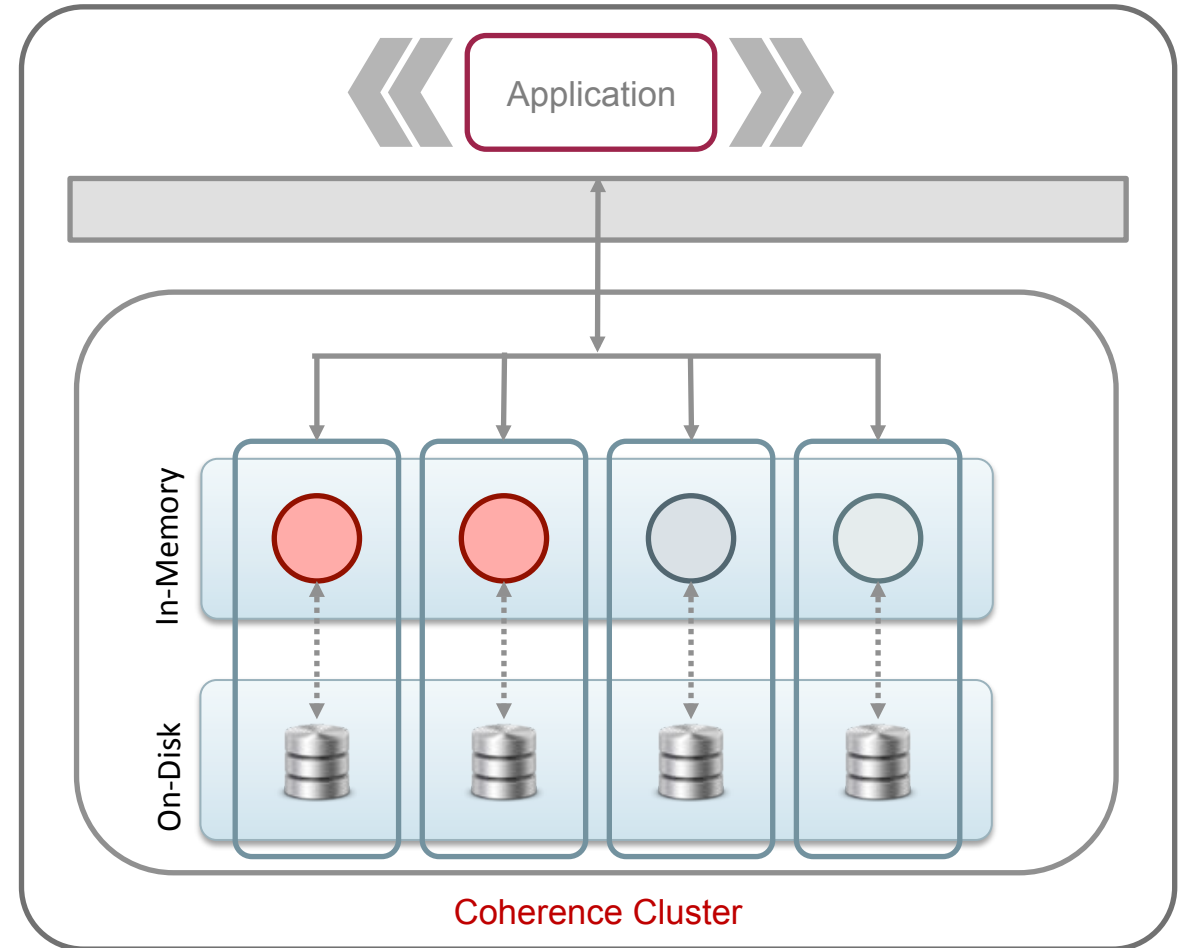
Multitenancy

Major Improvements in Core

Recoverable Caching

Enabling Coherence as Store of Record

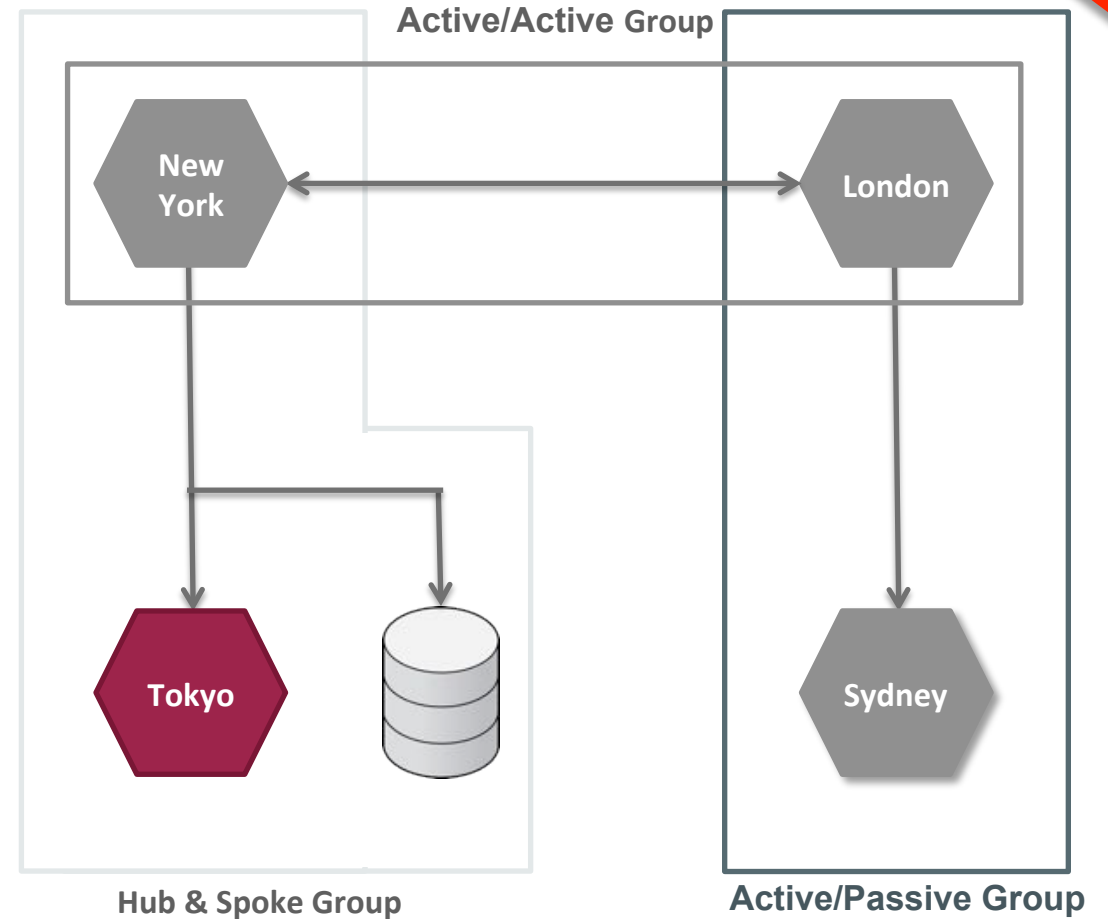
- Recoverable storage of cached data
- Automatic recovery from cluster failure
- Transactional or on-demand durability
- Multiple storage topologies
 - Maximum Scalability with distributed local disks
 - Maximum Availability with shared storage (e.g. SAN)



Federated Caching

Multi-Datacenter Solutions

- Distribute data grid updates
- Span on-premise and cloud cluster
- Multiple distribution strategies
 - Active/Passive
 - Active/Active
 - Hub & Spoke
- Overlay distribution strategies across locations
- Pluggable Conflict Resolution

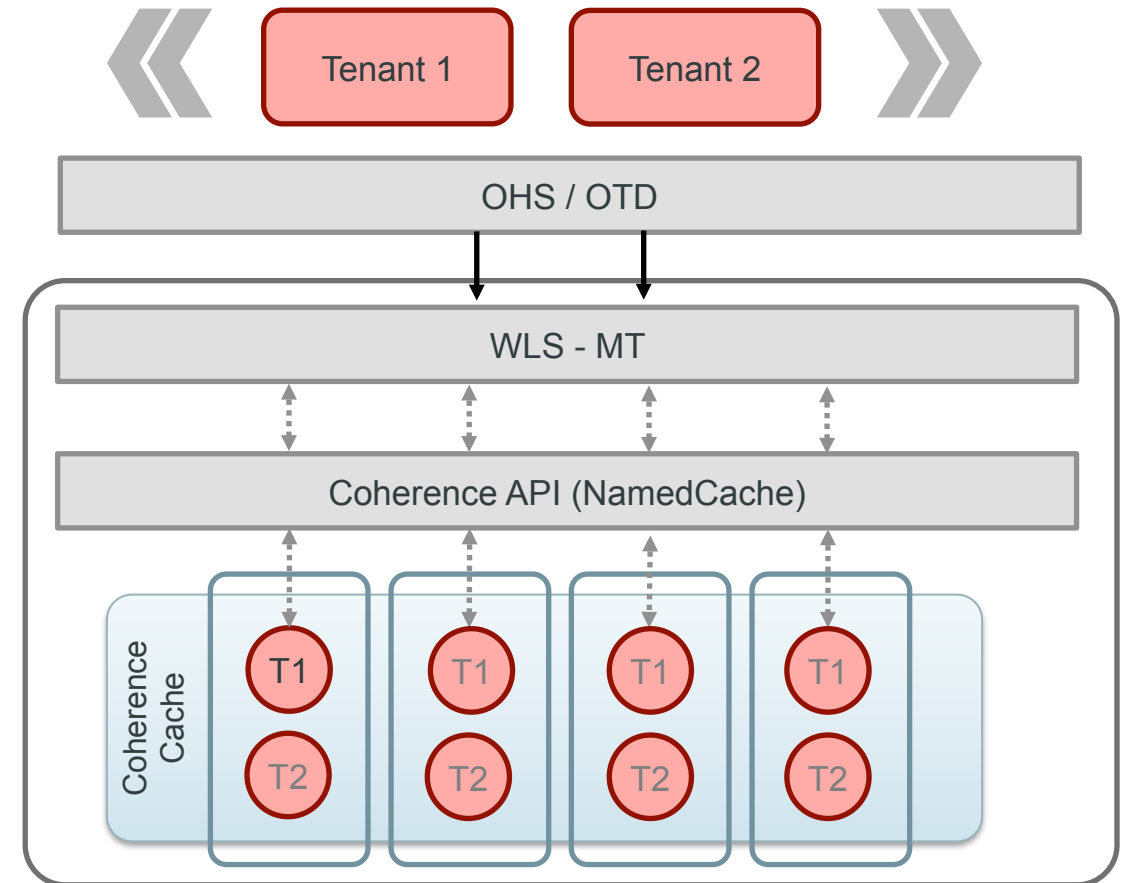


Preliminary Testing: >75% more throughput, up to 7x lower replication time

Multitenancy

Density and Operational Efficiency

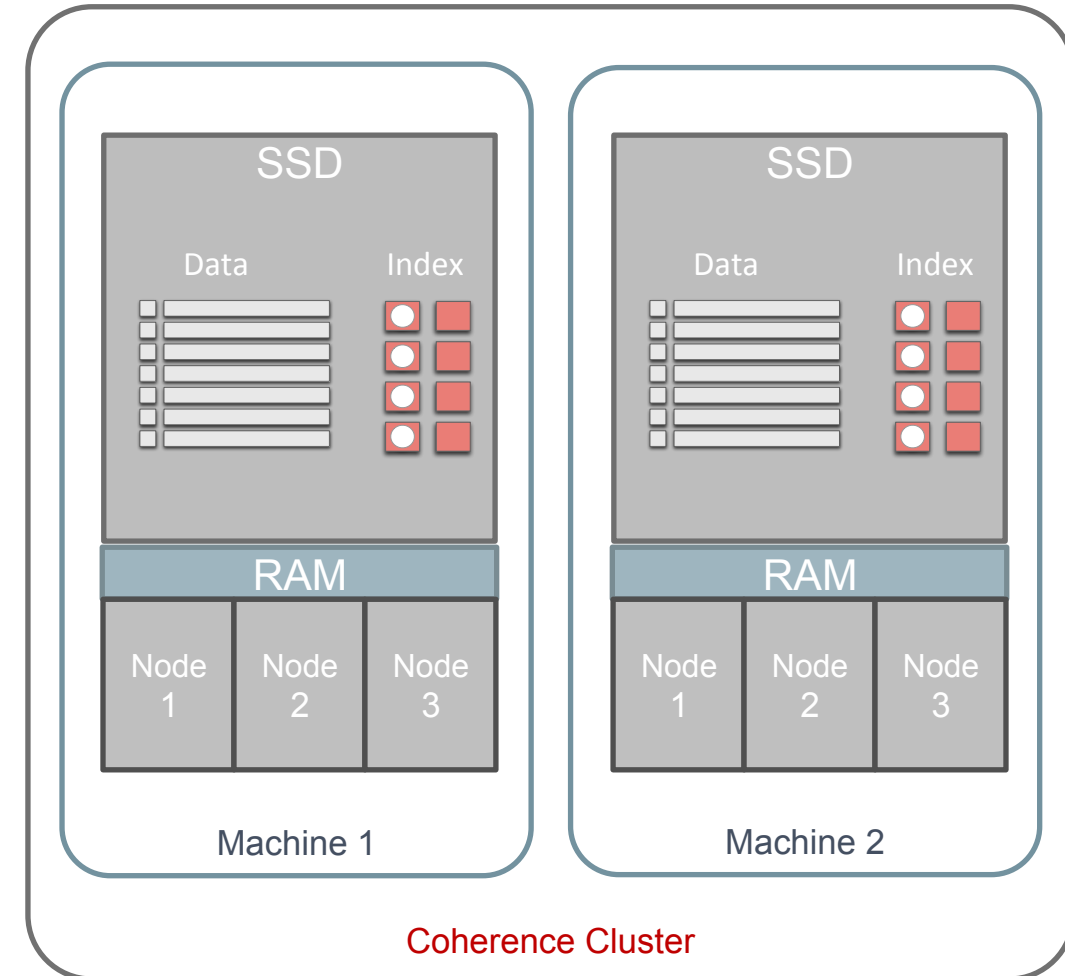
- Host Multitenant Apps in Coherence
 - Share cluster infrastructure
- Flexible cache configuration
 - Tenant-specific caches
 - Shared caches for common data
- Cache and Data Grid Operation Isolation
 - Tenant scoped to Coherence Service
- Resource tuning via tenant configuration
- Common CAF lifecycle tooling



Elastic Data Improvements

Increased Density and Larger Data Grids

- Improved Index Management
 - Reduced Memory Footprint
- Significantly increases cluster densities
 - ~10x over on-heap
- Opens up new “extra-large” use-cases
- Increased density reduces operational costs



Security Improvements

Auditing and Authorization

- New Security SPI for Auditing and Authorization for Entry Access
- Identity of a user will be passed with cross-node requests
- Security SPI's will provide entry point for plug-in points for specific implementations
- Auditing and Authorization SPI's will be able to log and approve all cache entry read/write operations



Managed Coherence Servers 12.2.1 Plans

- Consistent support for MT, Recoverable and Federated caching
- Coherence*Web
 - Improve “ease of use”, add session replication tab, options on deployment, scripting support
- Rolling Restart
 - Integration with Admin Console
- GAR Improvements
 - Shared library references
 - Multiple GAR’s in EAR, for instance for Coherence*Web and direct cache access
- Side by Side deployment support for cache client applications



Java Improvements

Developer and Ops Productivity

- JDK8
 - Support JDK 8 language features
 - Lambdas
 - Default Methods
 - Method References
 - Streams
 - Eliminate need to configure permgen
- Support for Java Generics

Examples

- Method References to add indexes:

```
cache.addIndex(Person::getName, false, null);
```

- Lambda Expressions as Entry Processors

```
positions.invokeAll(  
    equal(Position::getSymbol, "ORCL"),  
    e -> e.setValue(e.getValue().split(2)));
```

- New replaceAll method in Map Interface

```
positions.replaceAll(  
    equal(Position::getSymbol, "AAPL"),  
    (k, v) -> v.split(7));
```

The Big Picture: Coherence As A Service

Convergence of Multiple Investments

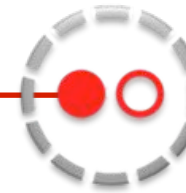
COHERENCE AS A SERVICE

- PUBLIC/PRIVATE/HYBRID CLOUD
- MULTIPLE APPS IN ONE CLUSTER
- DYNAMIC DEPLOYMENT AND SCALING
- MAXIMUM AVAILABILITY ARCHITECTURE



Multitenancy

- Density
- Isolation
- Elasticity



Java Cloud Service w/ Cache Service

- Self- or Oracle-Managed
- Nimbula Integration
- Compute Service



Maximum Availability

- Federated Caching
- Recoverable Caching



Managed Coherence Servers

- Container-based Deployment Model
- Administrative Tooling



Java SE

- Resource Isolation
- Resource Sharing

Program Agenda

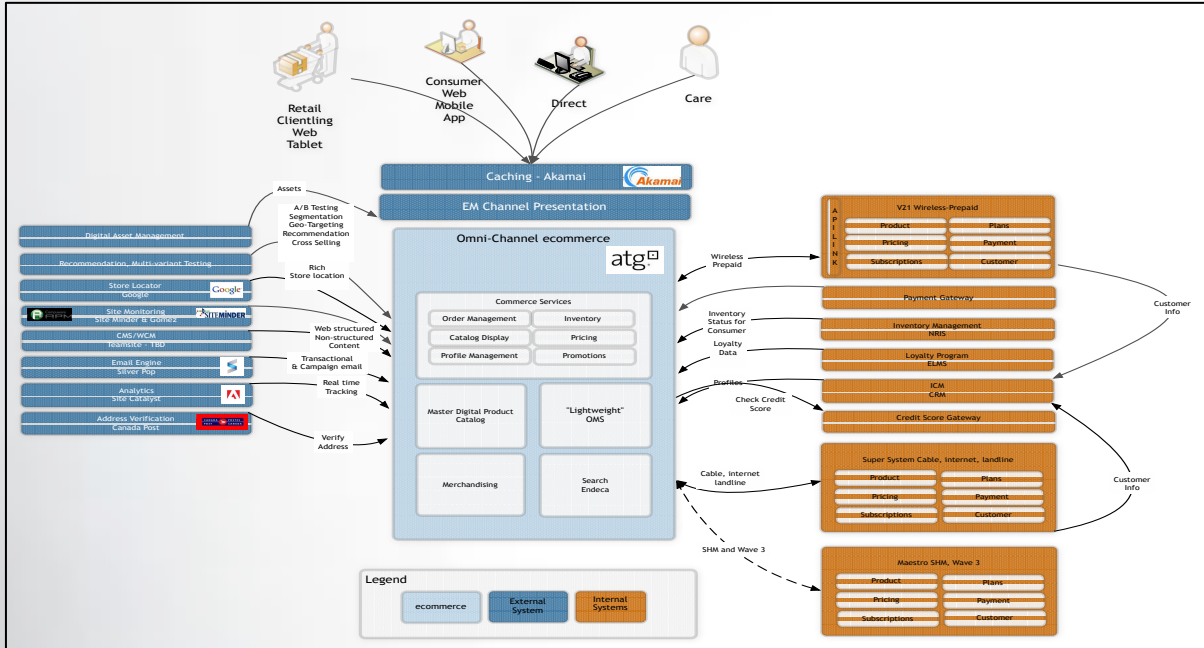
Introduction to Coherence & the 12c Release

Coherence Roadmap

 Customer Use Cases

Oracle Java Cloud Service

Current State – In Transition



Mass proliferation of fragmented, disparate Integration Platforms increasing complexity, risks and total cost of ownership



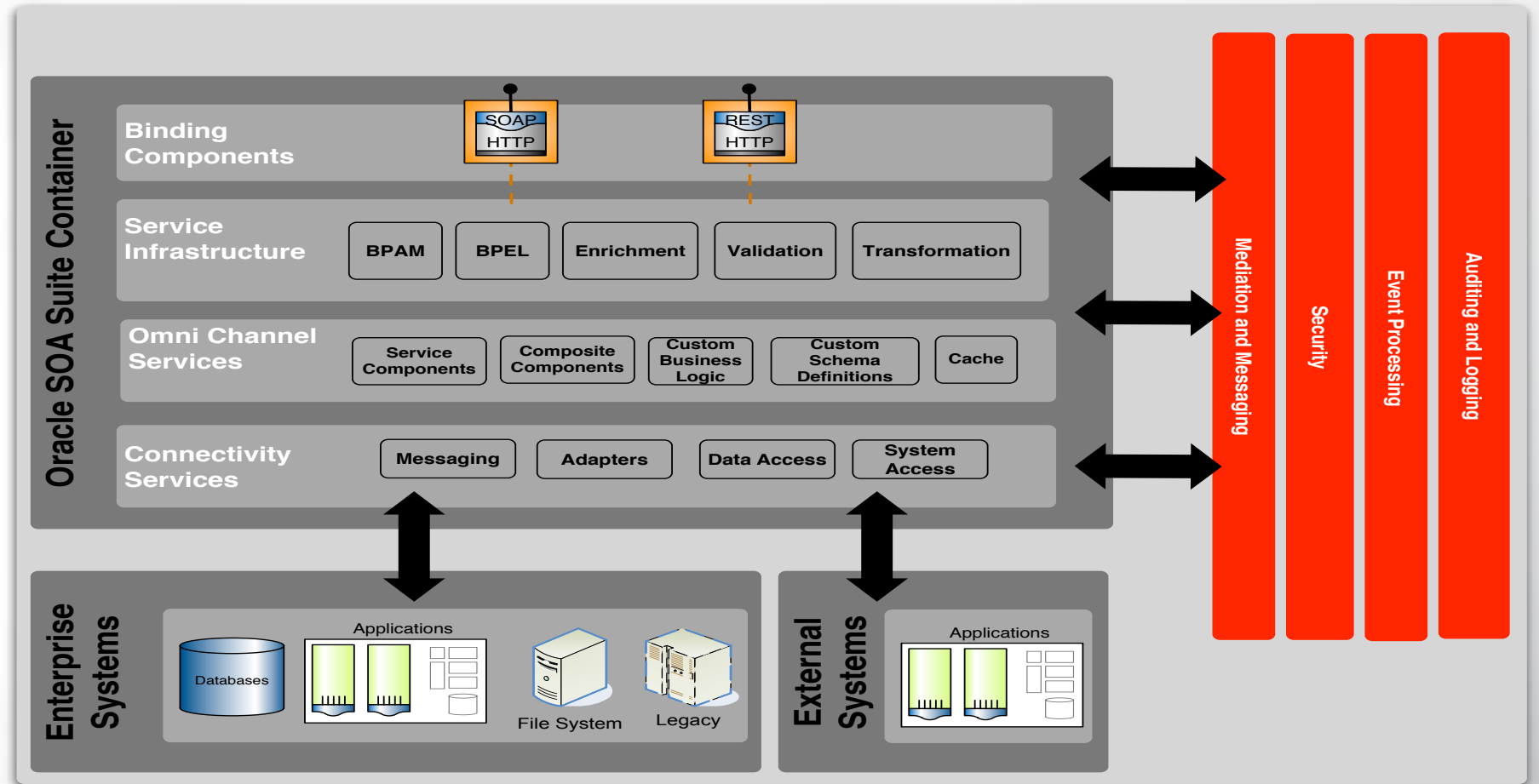
Transformative program in-flight to modernize and consolidate legacy integration into a next generation Service Oriented Architecture foundation leveraging Oracle Fusion Middleware

11/11/2014

Proprietary

WebLogic Service & Coherence Solution

Oracle SOA Suite
Weblogic
Coherence

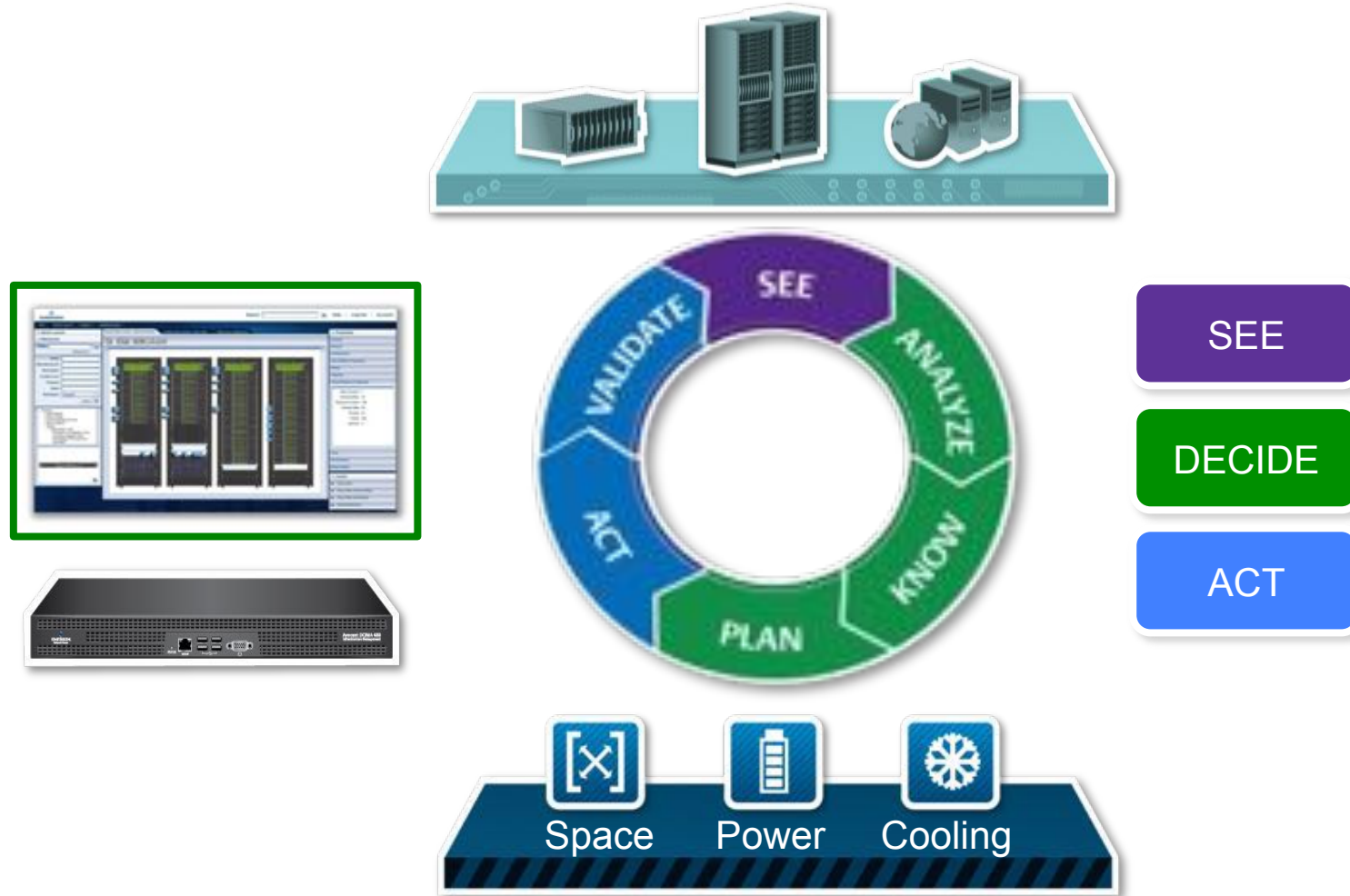


RAC
Oracle Elastic Cloud

Benefits

- Increased Business Agility
- Improved Customer Experience
- Lower Total Cost of Ownership

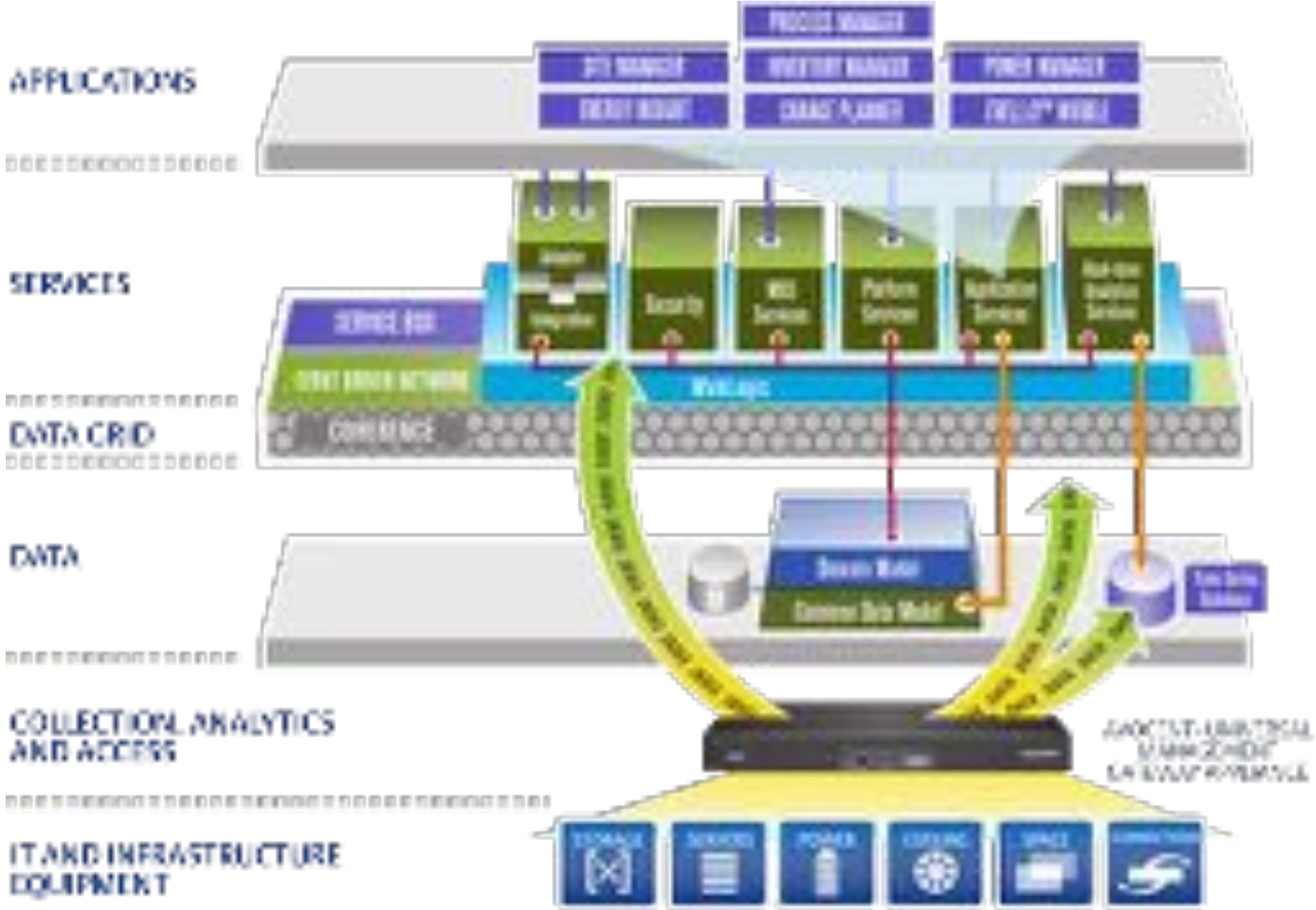
The Trellis™ Platform



- Complete Data Center Infrastructure Management (DCIM) package
- Single management portal for disparate IT and Facilities systems
- Built on industry-leading Oracle® Fusion Middleware
- Distributed architecture, enabling extreme scalability

The *Trellis*™ platform provides real-time, closed-loop Data Center Infrastructure Management

Trellis™ Platform Architecture

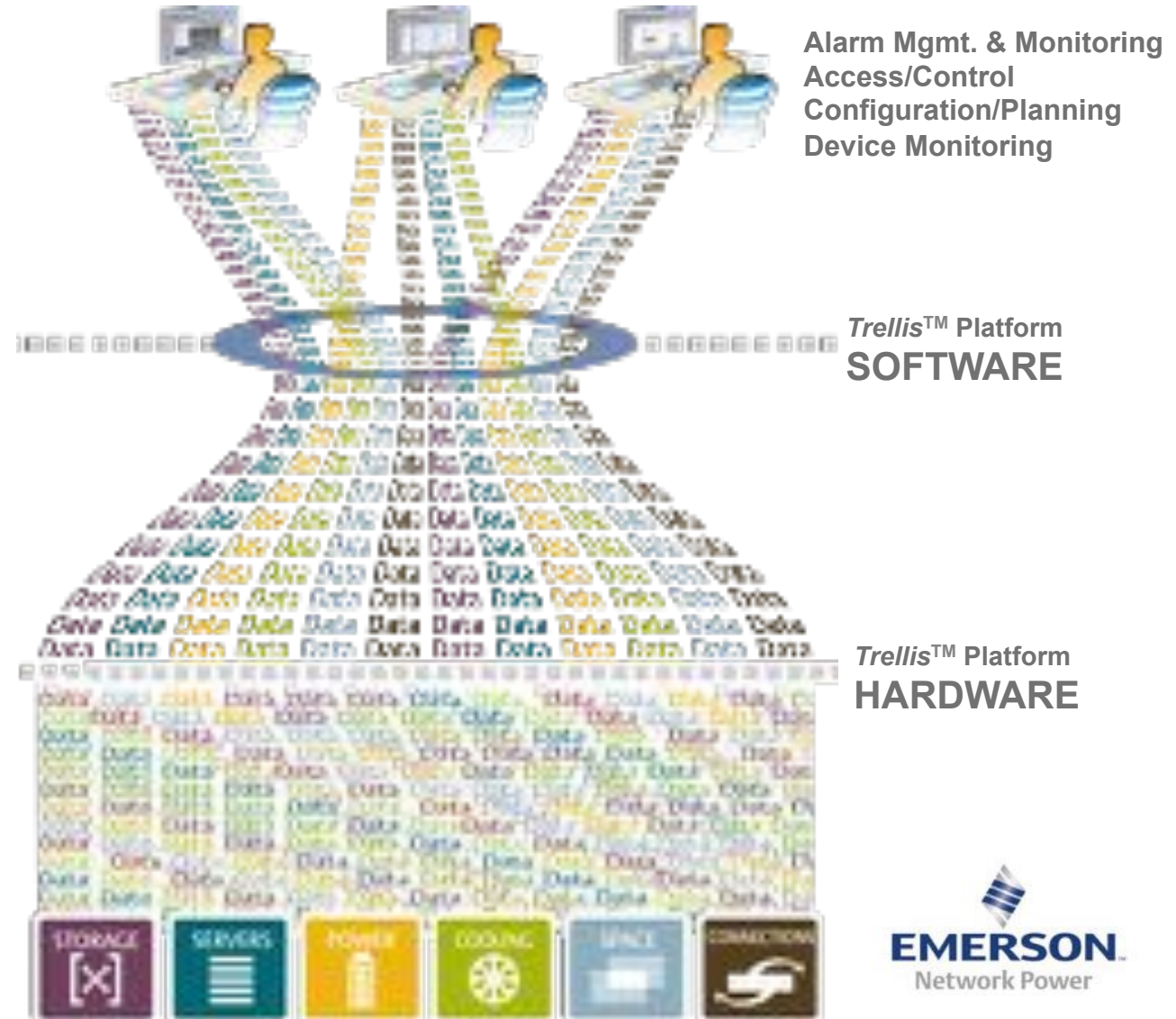


The *Trellis™* Platform relies on Oracle Fusion MiddleWare for key structural components

- WebLogic
- Database
- ADF
- SOA Suite
- Coherence
- OEP

Advantages of Adding a Data Grid Layer to the Trellis™ Platform

- Access to cached data increases application execution speed
- Event-driven evaluation of expressions allows actions to trigger only as changes occur
- Real-time applications have immediate access to current data from disparate sources
 - Avoids continuous calls to the Time Series Database
- On-demand event subscription improves efficiency
- Distributed memory structure supports clustering and high-availability



Program Agenda

Introduction to Coherence & the 12c Release

Coherence Roadmap

Customer Use Cases

 Oracle Java Cloud Service

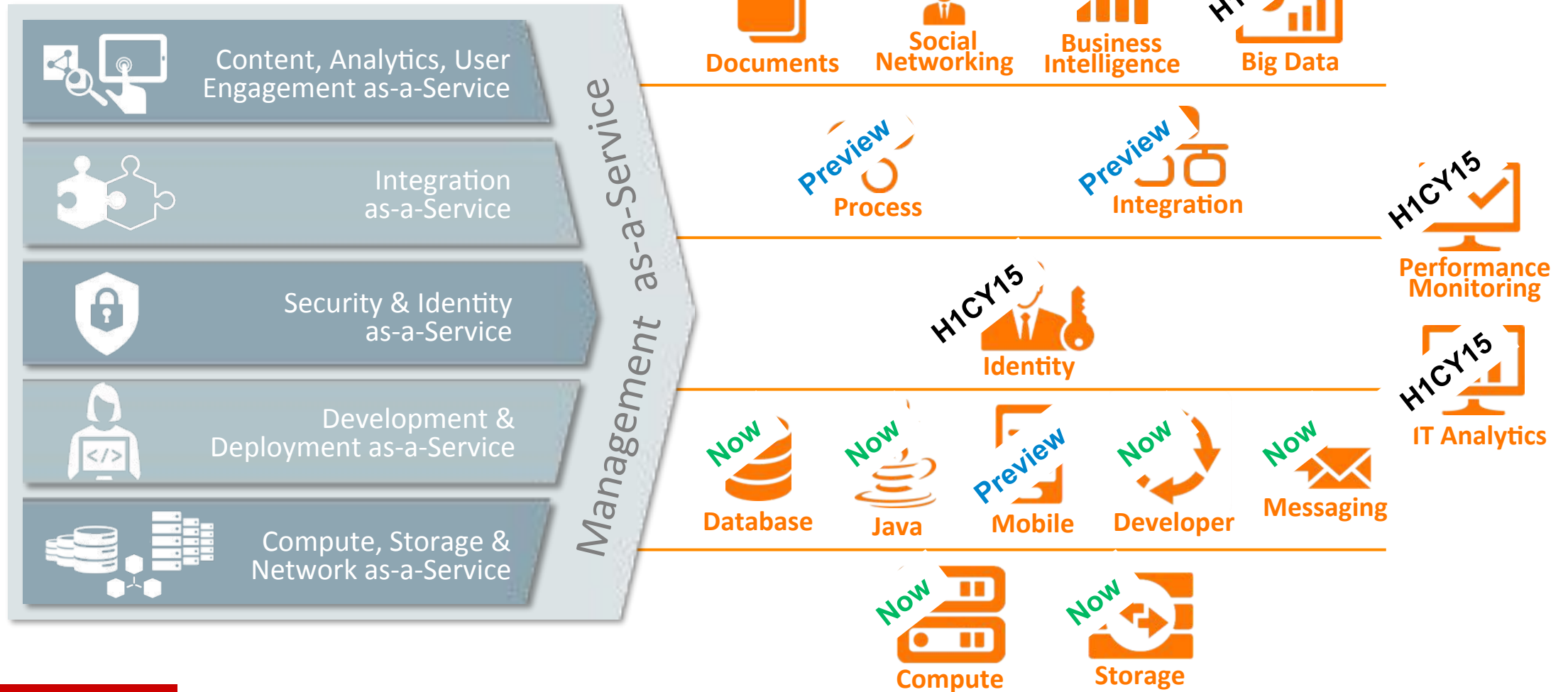
Oracle's Platform & Infrastructure Services

As Oracle Public Cloud Subscriptions



Complete Portfolio Of Cloud Architected Services

To Cross The “Cloud Chasm”



Java Cloud Service

For IT Development, Operations And Line of Business



Java Cloud Service

- Full-Featured: WebLogic 11g or 12c Instance
- Clustering, In-Memory, High Availability, Elastic Load Balancing, Scale Up & Scale Out
- Back Up/Restore, Patching, Application Server Management
- Full portability: On-premise to Cloud

Java Cloud Service – For SaaS Extension

- Pre-packaged, and pre-configured tools and frameworks needed to extend Oracle SaaS applications
- Dedicated, isolated WebLogic instance
- Available in T-shirt sizes - small, medium & large
- Platform lifecycle managed by Oracle, Application & Extension lifecycle managed by Customer

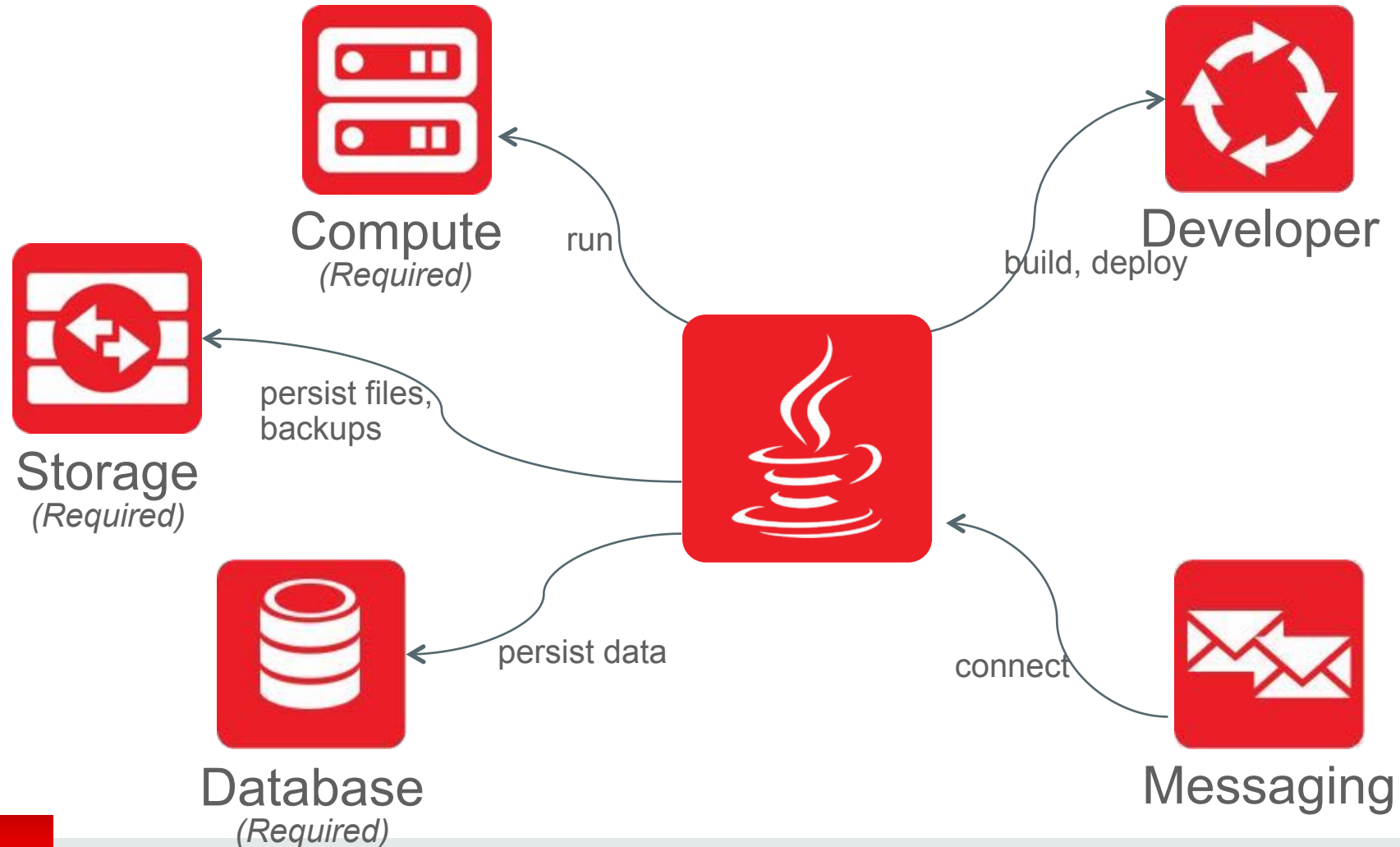
Java Cloud Service

Key Differentiators

- WebLogic/Java with Coherence and Database integration
- Secure, Highly Available with Clustering
- Rapid and fully automated provisioning
- IDE Choice - JDeveloper, Eclipse, NetBeans - and API access
- Simple, automated management

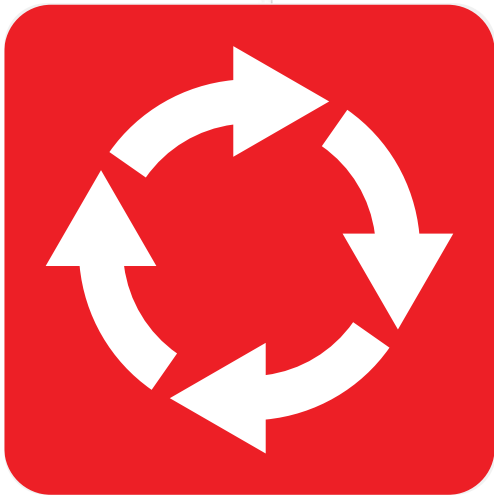
Java Cloud Service

Integration With Other Services



Developer Cloud Service

Key Features



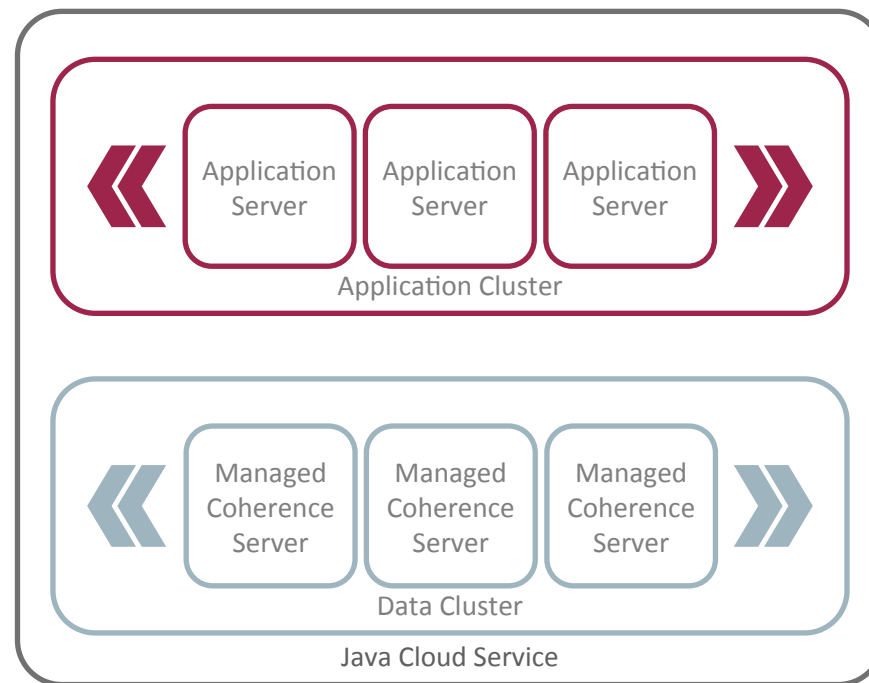
- Secure, Agile, Team Development in Oracle Cloud
- Tightly integrated in Oracle Cloud Ecosystem
- Accelerates Oracle PaaS/SaaS integration and extension
- Integrated IDEs: Eclipse, JDeveloper, Netbeans
- Supports the complete software development lifecycle
- Source control management (GIT), issue tracking, Hudson continuous integration, wiki collaboration



Coherence Cloud Service

Seamlessly Deploy

- Coherence as a feature of Java Cloud Service
- Build on WebLogic/Coherence 12c (12.1.2)
- Leverages Managed Coherence Servers
 - Develop, deploy, manage and monitor your applications via WebLogic Management Framework
- One Coherence cluster per domain
- Coherence (TCMP) cluster spans Java Cloud Service and Coherence Cloud Service managed servers
- Cache storage disabled in the application tier

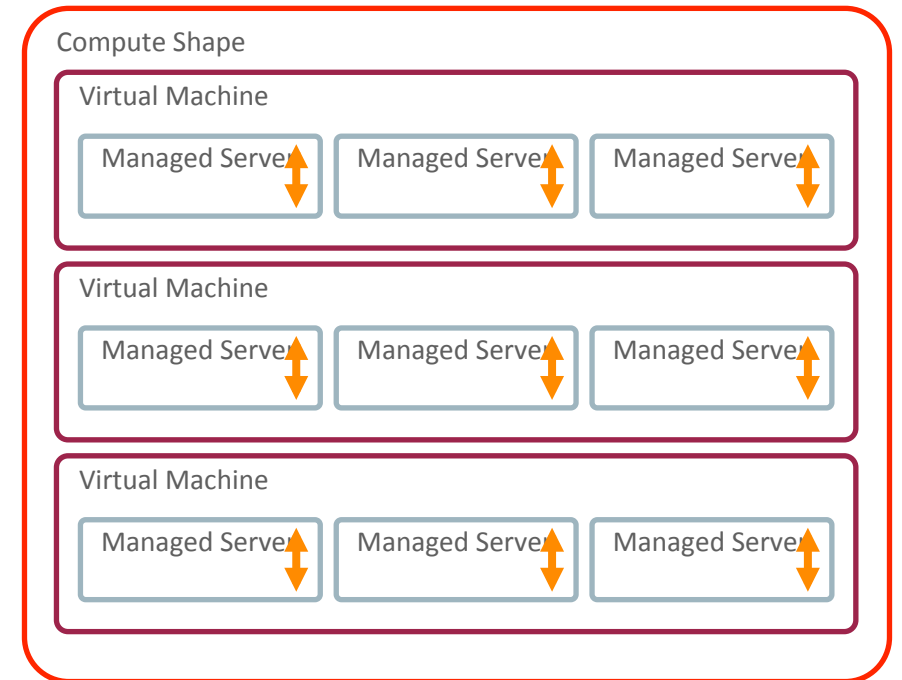


Coherence Cloud Service

Simplify Provisioning and Capacity Planning – Unit of Scale Defines Scale Characteristics

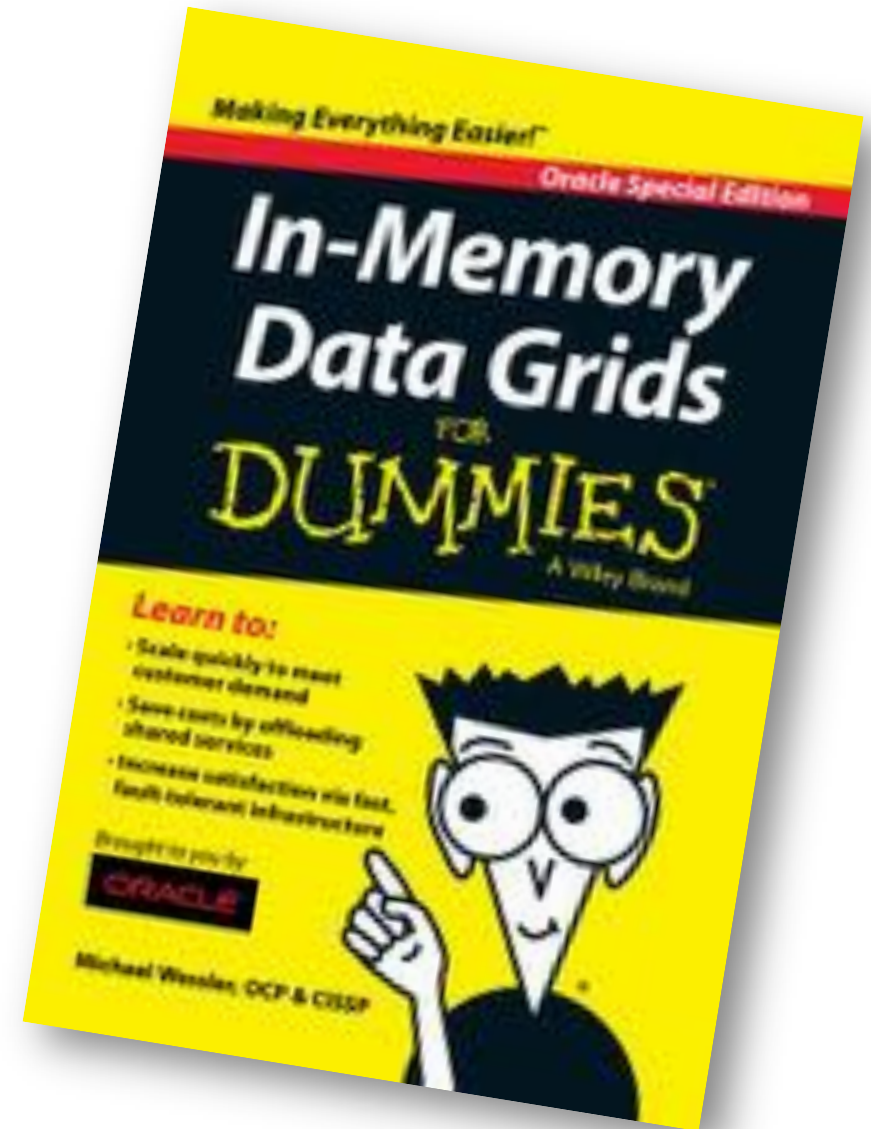
Unit of Scale:

- Compute Shape
- # of Virtual Machines per Compute Shape
- # of Managed Servers to Virtual Machine
- Heap Size per Managed Server



Brand New eBook!

- Covers all topics in greater detail
- Helps you get started
- Tips and Tricks
- FREE!
- Download at oracle.com



Join the Coherence Community



@OracleCoherence



/OracleCoherence



/OracleCoherence



Oracle Coherence
Users



blogs.oracle.com/
OracleCoherence

Visit us at: coherence.oracle.com

ORACLE®