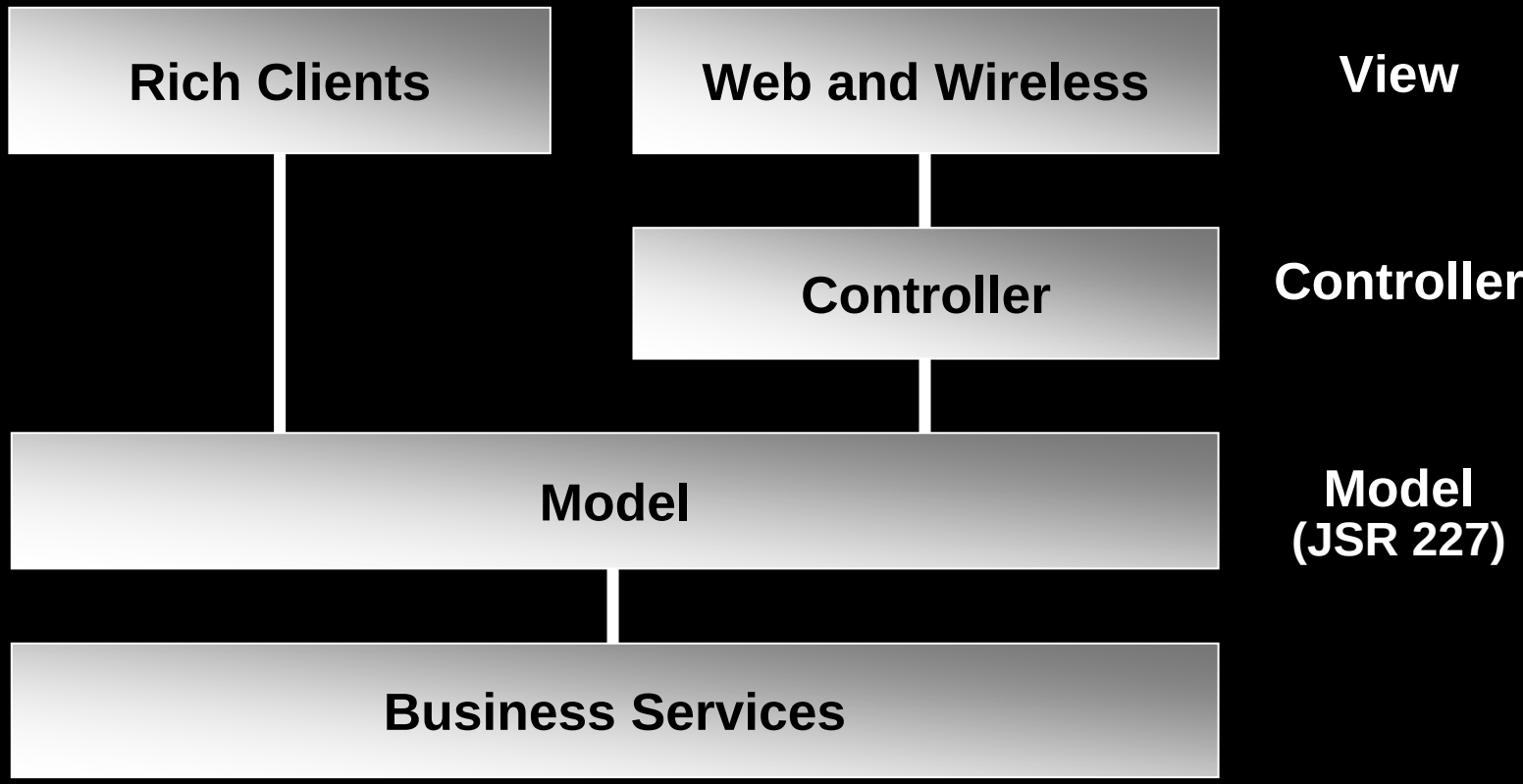
Oracle Toplink ili ADF BC?

Rudolf Jovanović
Principal Senior Consultant
Oracle Hrvatska

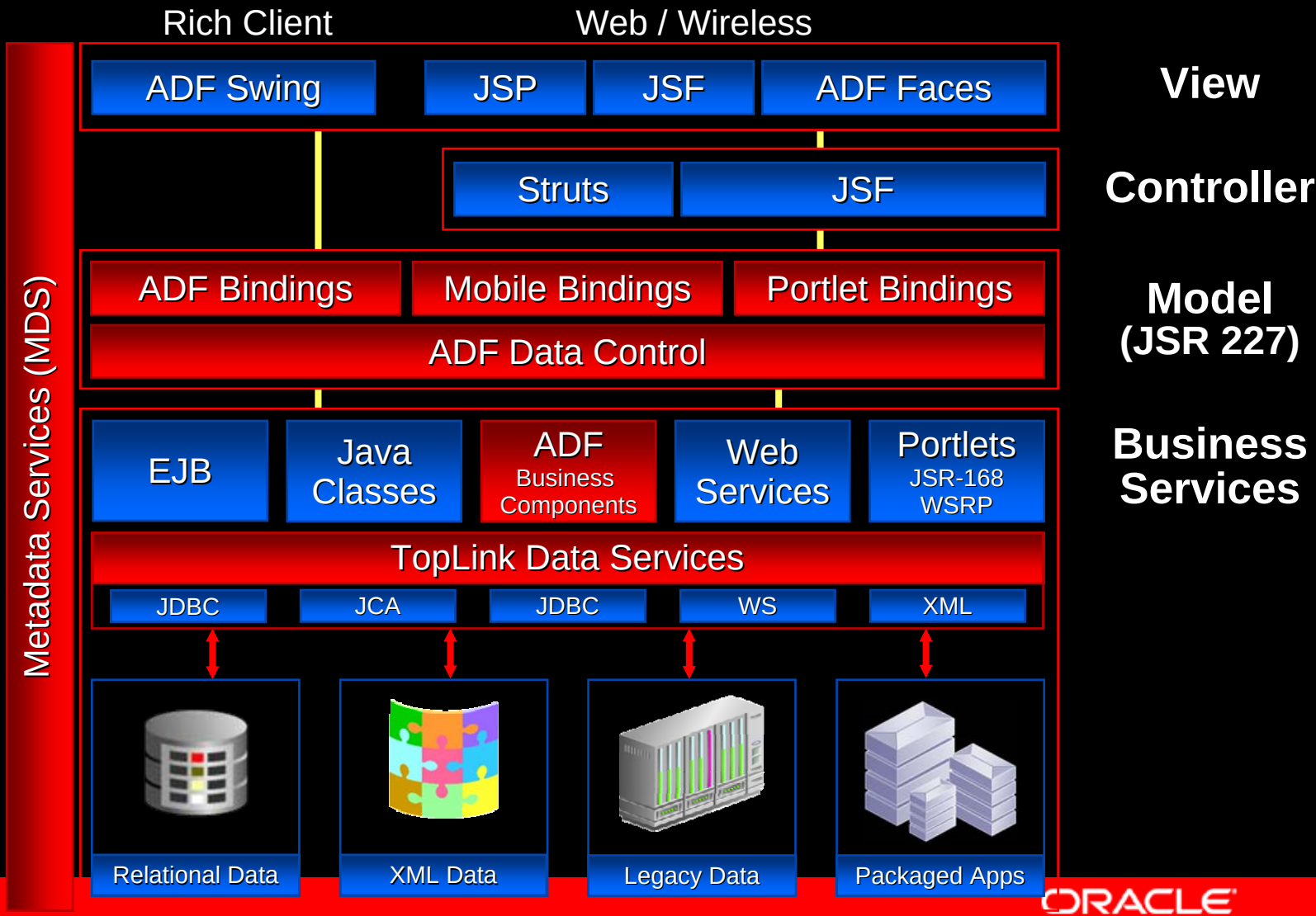
ADF Architecture



Business Services Choices

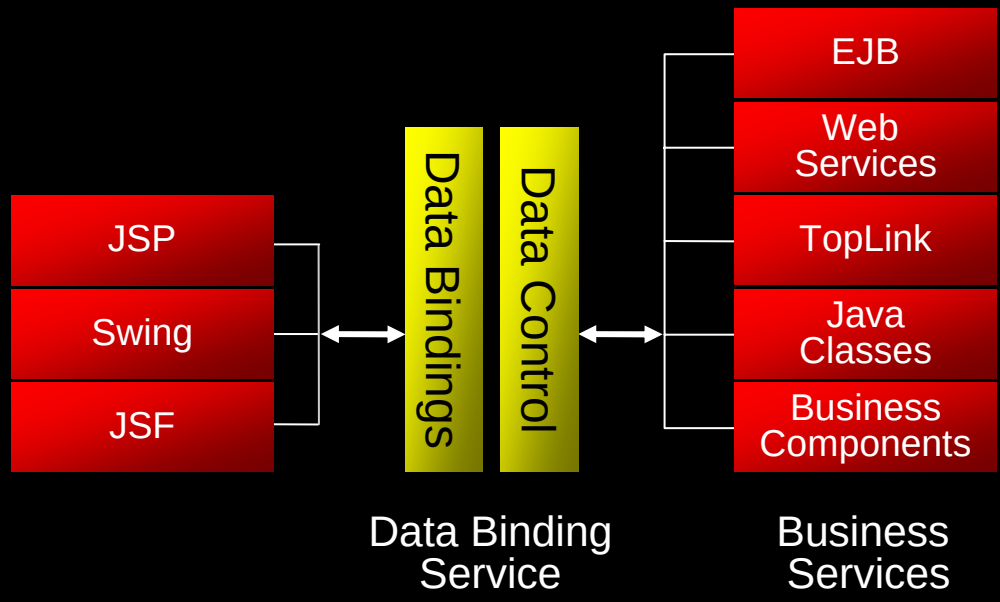
- TopLink
  - O/R Mapping and Persistence
  - For POJO and EJB CMP
- EJB
- Web Services
- JavaBeans
- ADF Business Components
  - SQL Driven POJO
- Build Your Own

ADF Architecture



JSR 227 – J2EE Data Binding

- Data Controls
  - Unified description of any Business Service
    - Collections
    - Attributes
    - Operations
- Data Bindings
  - Usages



ADF Business Components

- Declarative “SQL based” persistence
  - Declarative, familiar approach for 4gl developers
- Wizards, Editors, and UML Modelers
  - Visually design and modify declarative component settings
- XML-Configured, Lightweight JavaBeans
- *ApplicationModule* - Business Service Component
  - Use as JavaBean, EJB, Web Service
- *ViewObject* - Data Access Component
  - Performs SQL queries and coordinates with entity objects
- *EntityObject* - Business Domain Component
  - Encapsulates business domain data and validation

ADF Business Components Types

- The Business Components framework comprises two groups of components:
  - Business Domain components
    - Enforce business rules
    - Entity Objects, Entity Associations, and Custom Object types
  - Data Model components
    - Provide data access to client applications
    - View Objects, View Links, and Application Modules

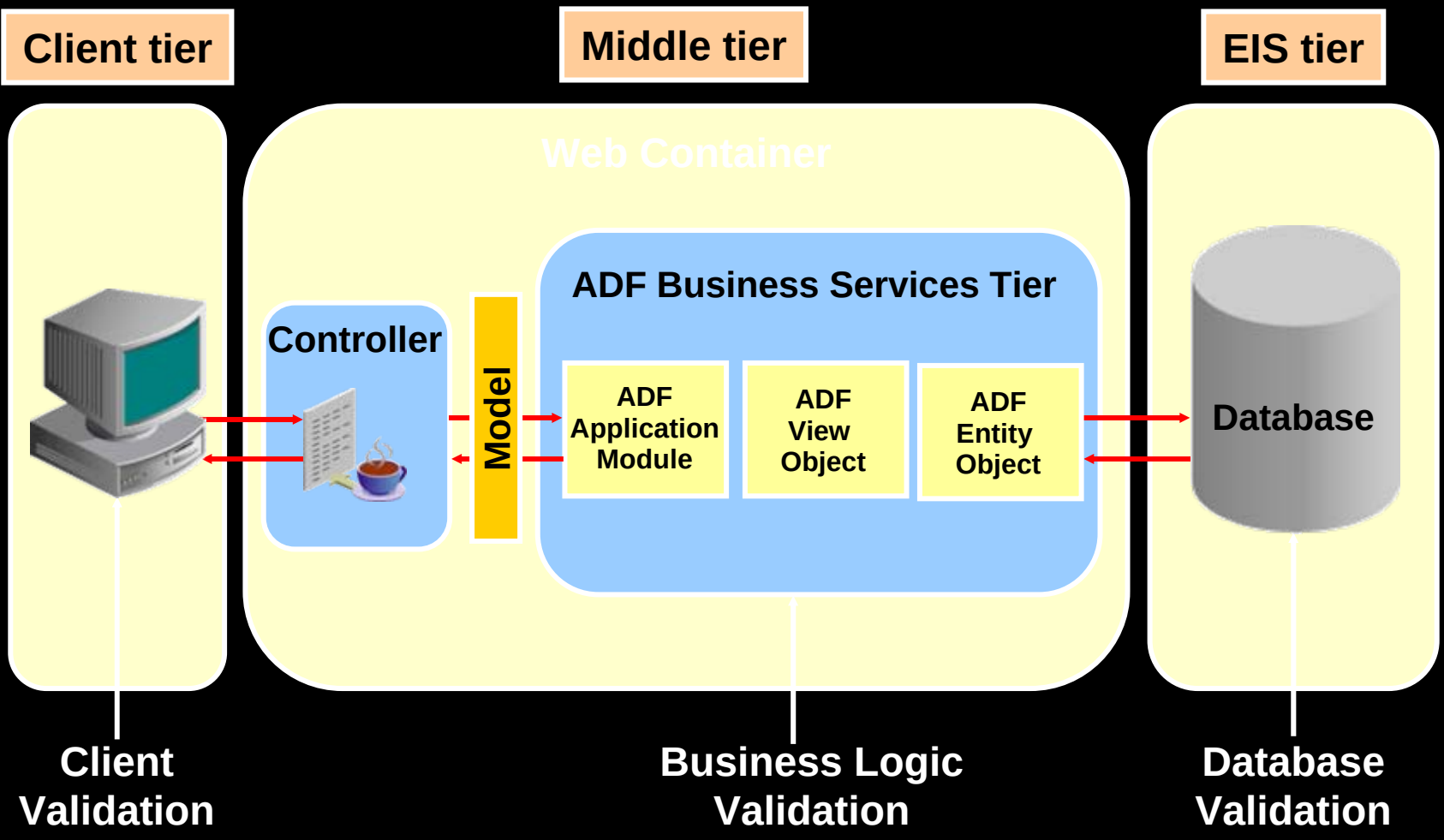
Business Domain Components

- Entity Objects
  - Enforce business rules and data validation
  - Persistence, caching
  - Data manipulation
- Entity Associations
  - Enforce relational rules between entity objects
- Custom Object Types (Domains)
  - Custom objects, such as AddressType
  - Object validation, such as EmailAddress

Data Model Business Components

- Application Modules
  - Manage transactions
  - Define data and methods that a client can see
- View Objects
  - Provide client access to data
  - Are collections of data for client access
- View Links
  - Link View Objects for master-detail coordination

Business Logic Validation



What is TopLink?

- A scalable enterprise Java persistence solution
- Addressing multiple data formats/storage
  - Relational
  - XML
  - Non-Relational
- Standards based for interoperability
  - Implements EJB 3.0 JPA and JAXB
  - Integrates through EJB, JCA, JTA, JDBC

Value of TopLink

- Performance and Scalability
  - Flexible query for optimized data access
  - Minimize transaction costs
  - Configurable caching with clustered coordination
- Developer Productivity
  - Simplified mapping using graphical editors
  - Generation and auto-map wizards
  - Error detection and warning during development
- Flexibility
  - Support of leading databases and Java containers
  - Java EE and Java SE

Proven Solution

Persistence Goes Prime Time at the U.S. Department of Defense

“I'd say we saved 20% to 30% in development time on the server side thanks to TopLink..”

“the Oracle9iAS/TopLink combo performed beyond initial expectations…”

Jim Scott, Northrop Grumman

Award winning Persistence

- JDJ Reader’s Choice
- JavaPro Reader’s Choice
- JavaWorld Editor’s Choice

CuraGen Speeds Search for Breakthrough Drugs Using Oracle9iAS

“"We handle a massive amount of data at CuraGen, tracking and analyzing hundreds of thousands of experiments each day. Oracle9iAS TopLink has been a critical link in accelerating our genomic research efforts." –

Steve Gold, Director of Bioinformatics, CuraGen Corporation

Over 12 Year History

Where does TopLink fit?

Java EE

Java SE

ADF

Spring Framework

TopLink Mapping Workbench

JPA | EJB 2.1 CMP/BMP | JAXB | EIS (JCA)

Object-Relational Mapping | Object-XML Mapping

Oracle TopLink

Packaged Apps

Databases

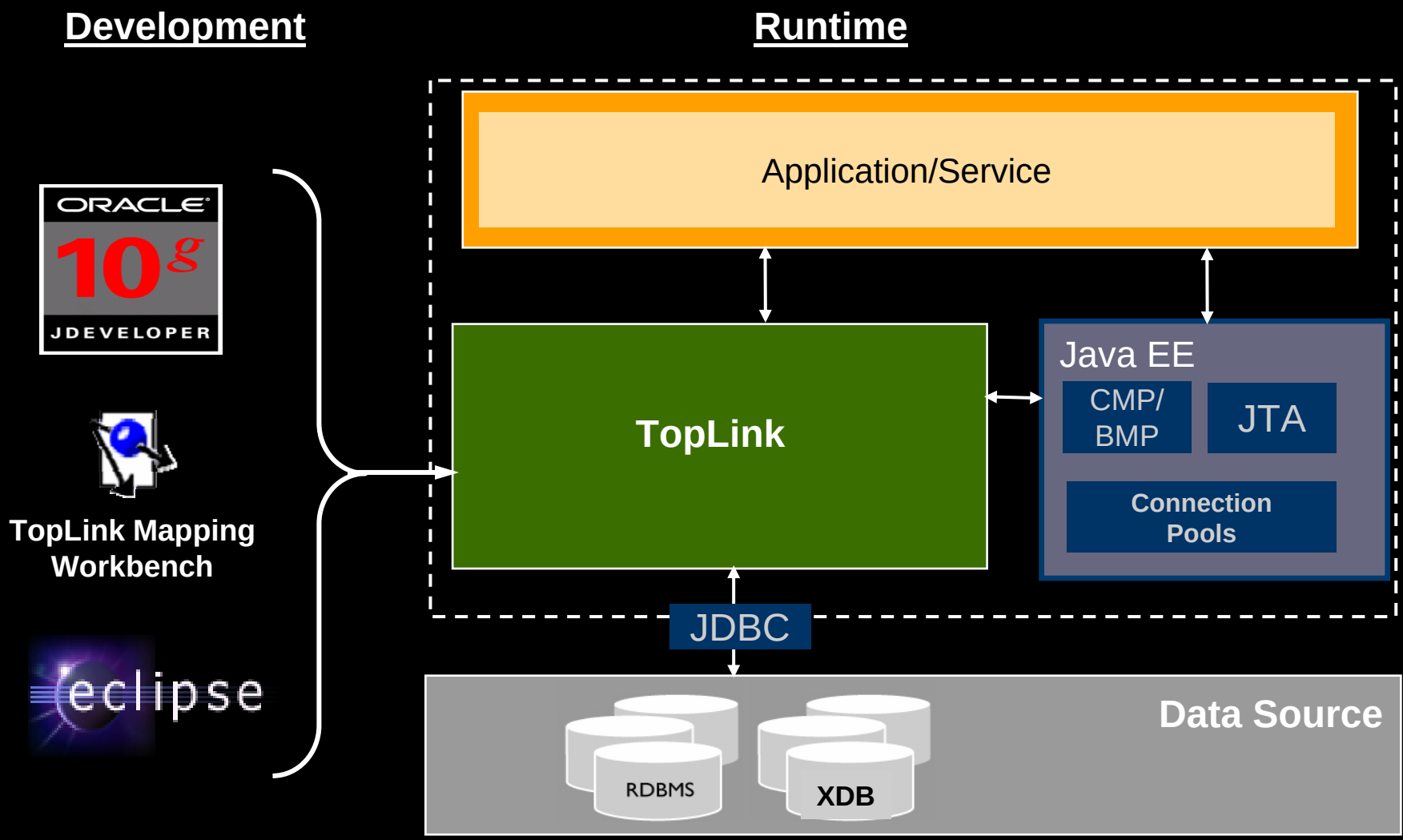
XML Data

Legacy Systems

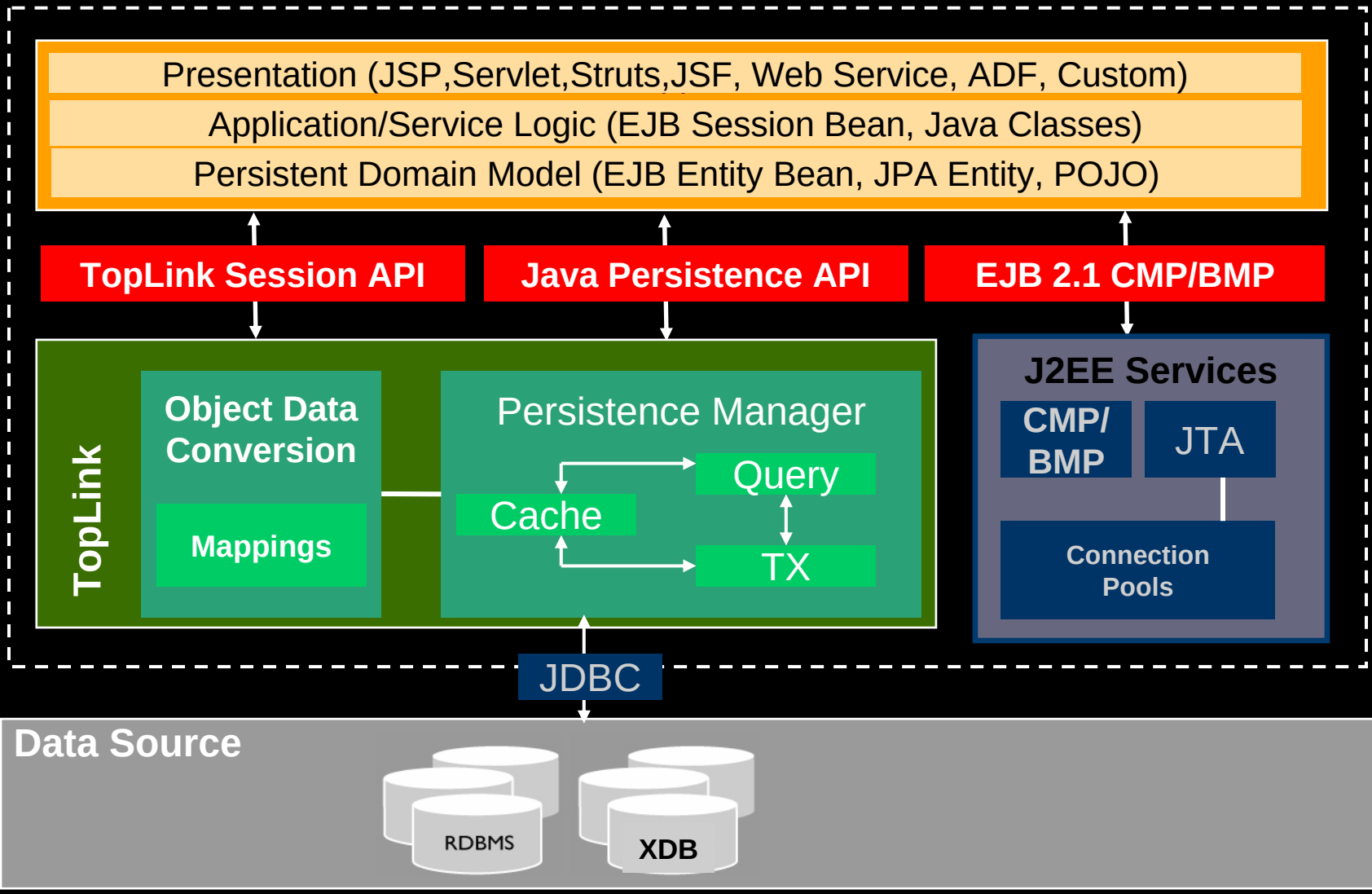
TopLink Capabilities

- Object-Relational Mapping
  - JPA: Java Persistence API (EJB 3.0/SE)
  - POJO in any Java container/architecture (EE/SE)
  - EJB 2.1 CMP and BMP (OC4J)
- Object-XML
  - JAXB
  - Non-Intrusive (meet in the middle) mapping
- EIS mapping using JCA Resource Adapters
  - XML mapping leveraging OXM and CCI mapping
  - Support MQSeries, OracleAQ, Sun JCA, XML Files, …

TopLink Object-Relational (ORM)



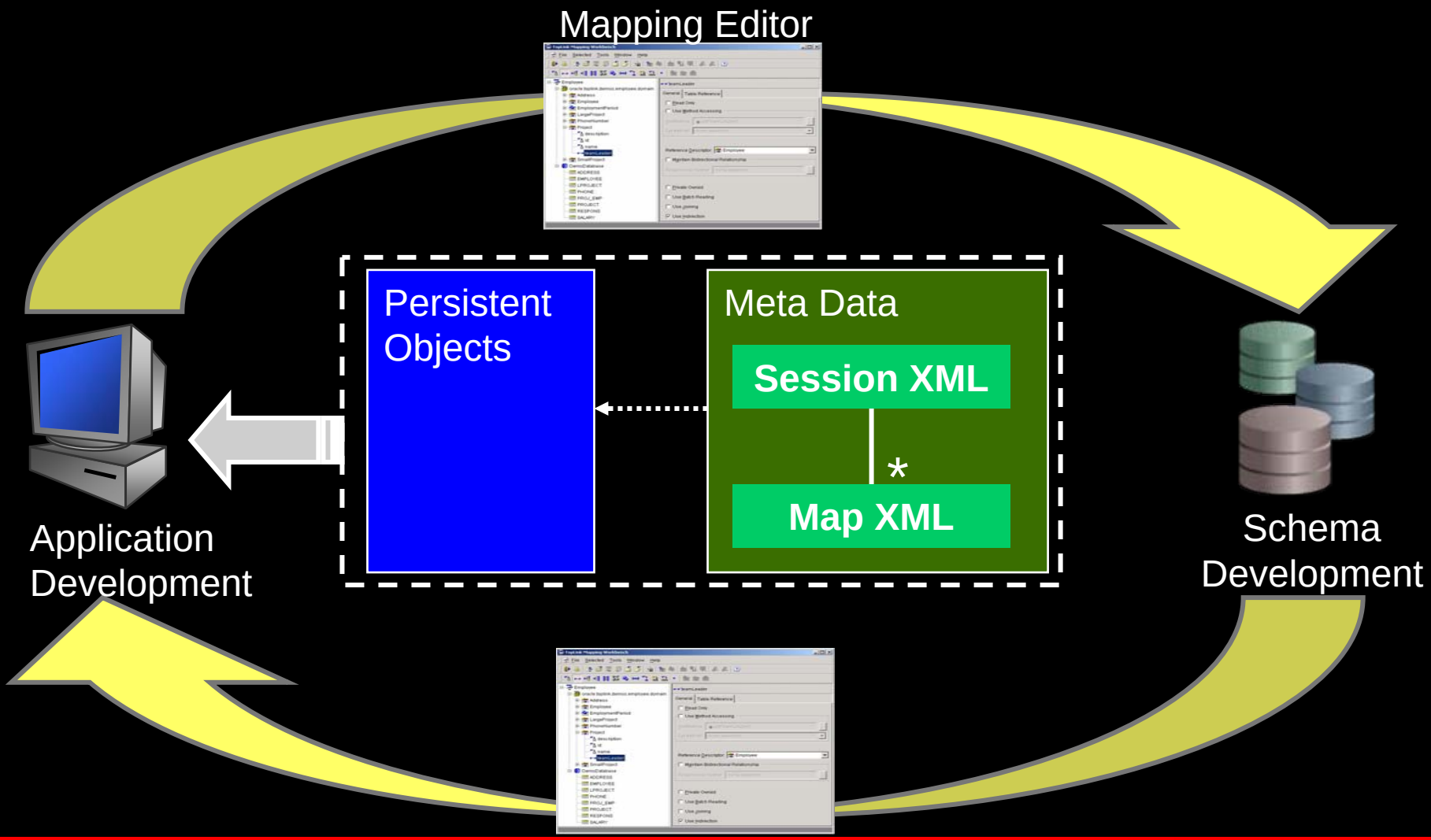
TopLink ORM in an Application



Key Features

- Metadata architecture
- Comprehensive visual mapping editors
- Advanced mapping support and flexibility
- Query flexibility
- Caching
- Concurrency protection
- Transaction support and integration
- Performance tuning options
- Application server integration

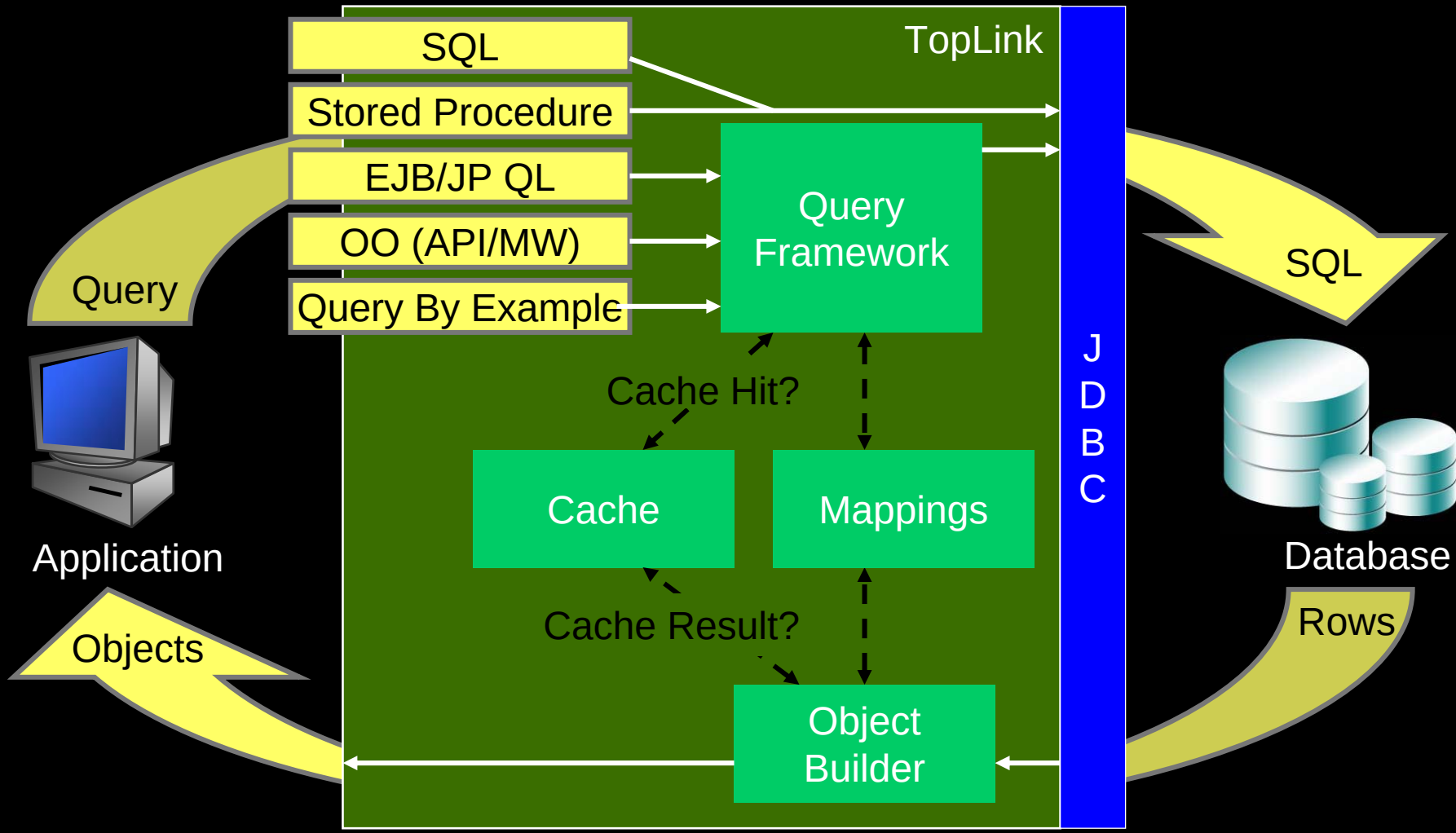
TopLink Design Time



TopLink ORM Runtime

- **Metadata** (Map/Project, Descriptors with Mappings)
  - Typically read from XML
  - Platforms: Database & Server
- **Session**: Primary runtime interface
  - Accessed through SessionFactory/SessionManager using sessions configuration (sessions.xml)
- **Query**: Requests to read or modify persistent objects
- **Cache**: Performance and Scalability optimization
- **UnitOfWork**: Transactional API for writing changes to objects

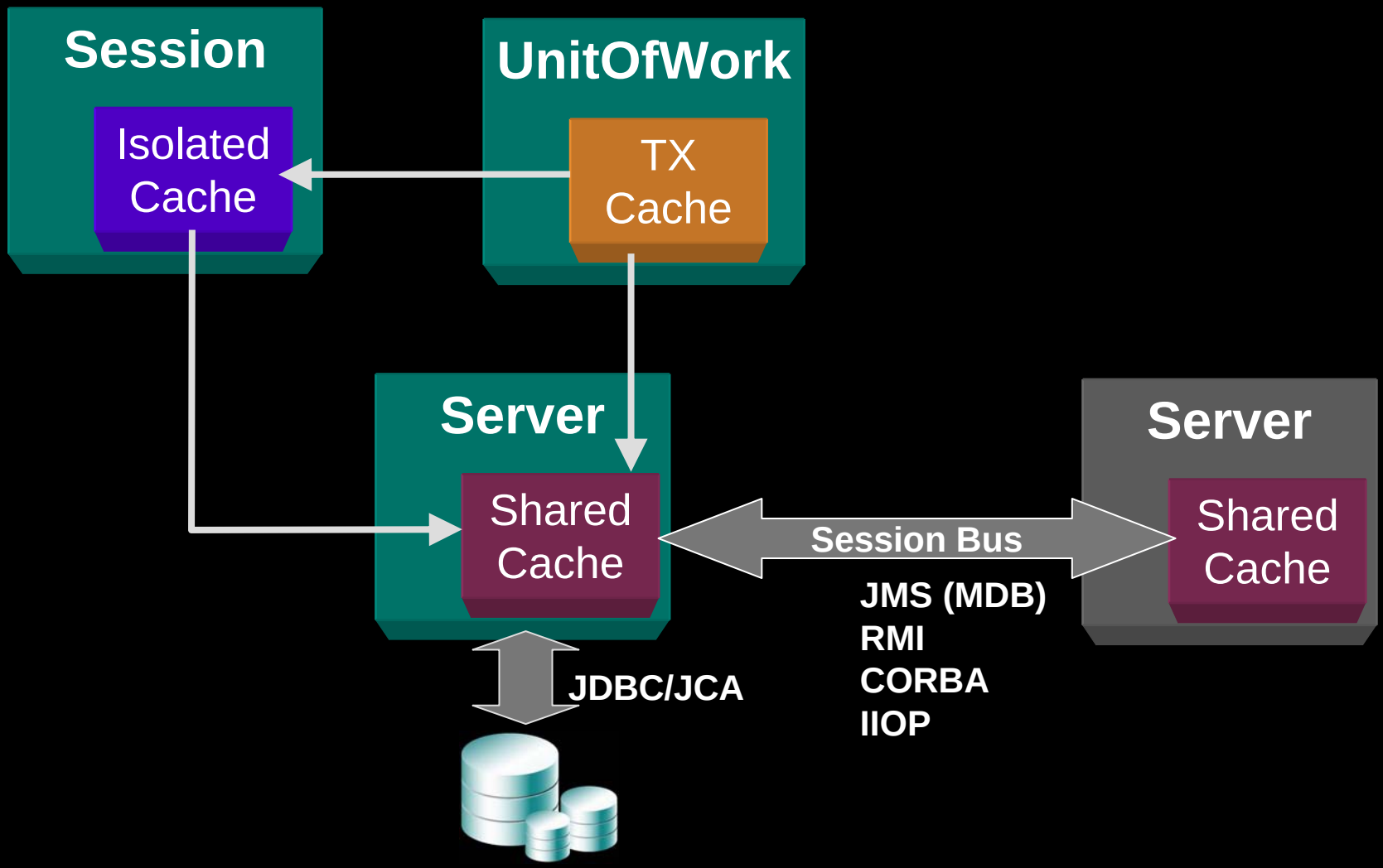
TopLink Query Execution



TopLink Caching

- Persistent objects cached by “identity”
  - Identity == Primary Key field(s)
- Benefits:
  - Avoid unnecessary database trips
  - Avoid re-building objects from data
  - Enables in-memory query processing
  - Can be coordinated in clustered deployments
- Developer Tasks:
  - Application specific cache configuration to optimize performance and minimize stale data
  - Leverage locking to avoid data corruption

Caching Architecture



Descriptor Caching Options

- SHARED or ISOLATED
- Type
  - SOFT-WEAK, HARD-WEAK: Shared with limited quantity
  - WEAK: Only held while used by the application
  - FULL: Reference data held for application's life
  - NONE: Read-only without relationships
- Invalidation/Expiration
  - Time to live (ms), Fixed Time of Day, API
- Coordination
  - Messaging: JMS, RMI, IIOP, CORBA, …
  - Modes: SYNC, SYNC+NEW, INVALIDATE, NONE
- Refresh Always
- Disable Cache Hits
- Only refresh if Newer (leverages optimistic locking field)

Configuring the Cache

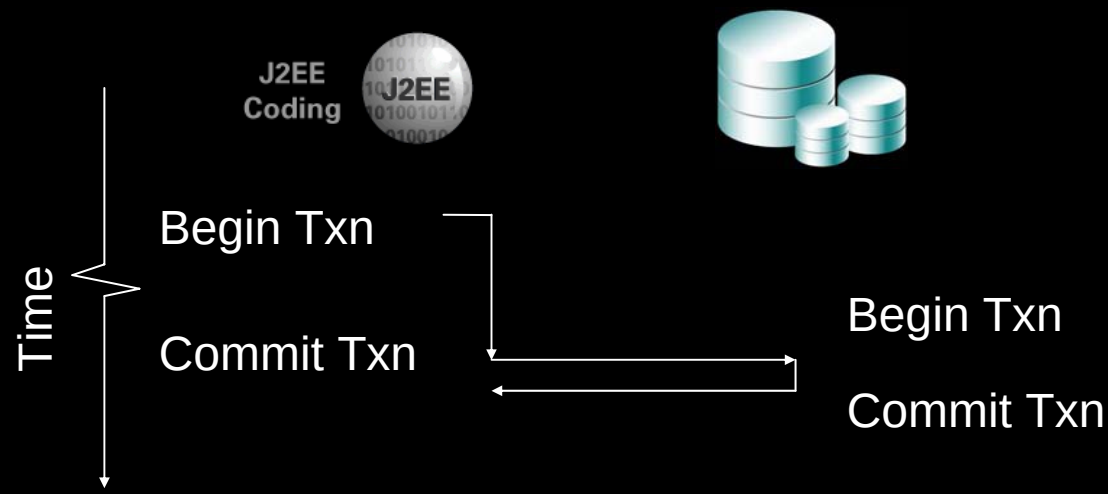
- Default: objects read are cached and trusted
- Configuration by entity type important
  - Volatility of data within application and through other applications
  - Shared nature of data between application clients
- Configuration Parameters
  - Cache isolation, type and size
  - Refreshing
    - By query (use-case) or descriptor (always)
    - Expiry/Invalidation
  - Cache Coordination
- Locking is the only way to avoid potential data corruption in concurrent write scenarios

Concurrency Protection: Locking

- **Prevent data corruption !!!**
- Java Developers think of locking at the object level
- Databases may need to manage locking across many applications
- TopLink is able to respect and participate in locks at database level
  - Optimistic: Numeric, Timestamp, All fields, Selected fields, Changed field
  - Pessimistic

Transactions

- Java apps typically support many clients sharing small number of db connections
- Ideally would like to minimize length of transaction on database



Transaction Features and Support

- UnitOfWork provides Java abstraction
- Minimizes database interactions
  - Calculates the minimal change set at commit time (deferred write)
  - Only the minimal DML issued
- Respect database integrity
  - Orders INSERT, UPDATE and DELETE statements
- “Unit Of Work” fully supports JTA

Transactions and the Cache

- Client
  - Acquire UnitOfWork
  - Make Changes (Read, Update, Delete)
  - Commit (direct or JTA)
- UnitOfWork
  - Write Changes
    1. Calculate minimal changes
    2. Order SQL statements
    3. Execute statements
  - Cache Merge: Post TX commit
    1. Merge in Shared/Isolated cache
    2. Send change-set to other nodes if using coordination

Performance and Tuning

- TopLink focuses on performance and scalability
- Highly configurable and tunable
  - Guiding principle: minimize and optimize database calls
  - Enable application specific tuning
  - No two applications are the same, TopLink allows for decisions on what specific behavior needs to be configurable depending on situation
- Flexibility of TopLink allows efficient business models and relational schemas to be used
- Leverages underlying performance tuning features
  - Java, JDBC and the underlying database technology

Performance and Tuning Options

- Minimal Writes, Updates
- Batch Reading, Writing
- SQL ordering
- Transformation support
- Existence checks
- Stored procedures
- Statement Caching
- Scrolling cursors
- Projection Queries
- Partial Attribute Queries
- Bulk Update Queries
- “Just in Time” reading
- Automatic change detection
- Caching policies and sizes
- Parameterized SQL (binding)
- Pre-allocation of sequence numbers
- Cache Coordination
- Optimistic, Pessimistic locking
- Joining object retrieval optimization
- In memory querying
- Dynamic queries
- Optimized Change Tracking

AND MUCH MORE!

OracleAS (OC4J) Integration

- TopLink CMP support in OC4J
  - CTS 1.4 compliant EJB CMP solution
  - Many value-added features beyond specification
- Enterprise Manager Support (JMX)
- Diagnostics
  - Monitoring support through DMS
  - Integrated logging
- Security Policies

Oracle DB Features

- Oracle native SQL and custom operators
- Isolated session cache and connections for use with Oracle DB's VPD/OLS
- Proxy Authentication
- Support for XDB-XMLType and SQLX
- Stored Procedure & Function
- TIMESTAMP & TIMESTAMPTZ (oracle.sql)
- Configurable value return from write
- Spatial, Object-Relational, ADTs
- Historical Flashback…

Historical Support

- Generic support for any RDBMS
- Custom support for Oracle 10g+ Flashback
- Configure and use change management of objects over time
  - Versioning
  - Auditing
- Point in time querying
- Historical Session

EJB 3.0 Java Persistence API

- JSR 220: Enterprise Java Beans 3.0
  - Component model improvements: SB, MDB, Timer, …
  - Java Persistence API (JPA)
- Goals:
  - Simplify EJB — make it easier to use
    - Simplified set of APIs
    - Eliminate requirement for deployment descriptors
    - Facilitate test-driven development
    - Improve developer productivity
    - Persistence based on proven solution
  - Capture broader range of developers
    - Make it simpler for average developer
    - Increase developer base, target more corporate developers

Primary Features

- POJO-based persistence model
  - Simple Java classes—not components
- Support for enriched domain modelling
  - Inheritance, polymorphism, etc.
- Expanded query language (JP QL)
- Standardized object/relational mapping
  - Using annotations and/or XML
- Usable in Java EE and Java SE environments
- Support for pluggable persistence providers

Entity == POJO

```
public class Employee implements Serializable {

        private long id;



        private String name;

        public long getId() { return id; }
        public void setId(long id) { this.id = id; }

        public String getName() { return name; }
        public void setName(String name) { this.name = name; }
}
```

Entity

```
@Entity
@Table(name="EMPLOYEE")
public class Employee implements Serializable {
        @Id
        @Column(name="ID")
        private long id;

        @Basic
        @Column(name="ENAME")
        private String name;

        public long getId() { return id; }
        public void setId(long id) { this.id = id; }

        public String getName() { return name; }
        public void setName(String name) { this.name = name; }
}
```

Configuration by Exception

```
@Entity

public class Employee implements Serializable {
        @Id

        private long id;


        @Column(name="ENAME")
        private String name;

        public long getId() { return id; }
        public void setId(long id) { this.id = id; }

        public String getName() { return name; }
        public void setName(String name) { this.name = name; }
}
```

ORM Config using XML

- Annotations and/or XML can be used

```
<entity class="model.Employee">
  <attributes>
    <id name="id"/>
    <basic name="name">
        <column name="ENAME"/>
    </basic>
  </attributes>
</entity>
```

TopLink Essentials: JPA RI

- **Open Source Reference Implementation** of JPA
- Derived from proven Oracle TopLink product
- Available through Sun's OS Glass Fish Project

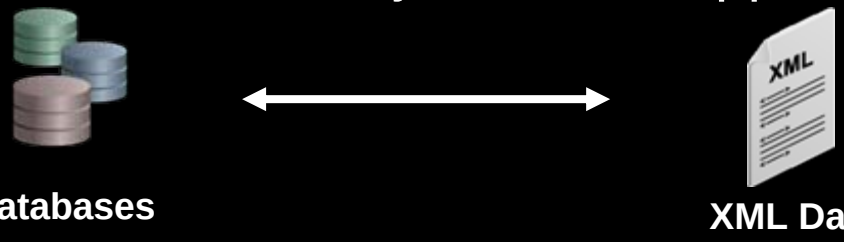
  https://glassfish.dev.java.net/
- Easy upgrade to full TopLink
  - Enhanced caching – performance & scalability
  - Advanced ORM capabilities
    - Including leveraging of advanced DB features
  - Object-XML (JAXB) and EIS Support
- Shipped as default JPA provider in OracleAS 10.1.3.1

JPA Tooling

- JDeveloper 10.1.3
  - Entities from tables
  - Session Bean generation
    - EJB 3.0 Entities or TopLink POJOs
  - ADF Integration
- Eclipse
  - Dali Project – www.eclipse.org/dali
  - JPA development support within WTP
  - Lead by Oracle with contributions from others
  - Available for download

TopLink JAXB

- Provides complete Object-XML mapping capabilities
  - Allows developers to work with XML as objects
  - GUI tool for mapping
  - Efficiently produce and process SOAP messages
- Supports Object-XML standard - JAXB
  - Provides additional flexibility to allow complete control on how objects are mapped



TopLink JAXB Benefits

- Rich set of mappings providing complete control and flexibility to map objects to any XSD
  - Direct, composite object, composite collection, inheritance, positional, path, transformation ….
- Visual tool to define mappings
- Leverages object-relational mapping features and infrastructure
- Embraces standards
  - JAXB and JAXP compliant
  - On JAXB EG, pushing advanced features into JAXB 2.0

Combining ORM and JAXB

- TopLink's metadata based approach allows the same POJO model to be mapped to both XML and Relational.
  - Supports construction of Web Services using persistent POJOs
  - Persistent POJOs can be mapped to any number of XML schemas—support multiple web services
  - Mapping is bidirectional:
    - Unmarshall XML to objects and then persist
    - Marshall persistent objects to XML

TopLink EIS

- Provide persistence support for non-relational data stores using J2CA/JCA
- Combined with TopLink JAXB support allowing XML interactions
- Mapping support for CCI
- TopLink Workbench mapping and configuration support
- Out of the box support for:
  - MQSeries, OracleAQ, Sun JCA, XML Files

Oracle ADF and TopLink

- JDeveloper productivity tooling
  - Generate TopLink JPA entities or TopLink POJOs from tables
  - Generate service façade with queries and entity lifecycle methods
    - EJB 3.0 Session Beans Facade
    - Java POJO Façade
  - Generate test client
- Support for data control generation from service façade enabling rapid and declarative presentation development

Spring Framework and TopLink

- Integrated DAO support for TopLink in Spring 1.2
- Integrated support for TopLink Essentials in Spring 2.0
  - Traditional DAO template based approach supported
  - Spring 2.0 supports JPA container contract which enables advanced configuration and ease of use
  - Spring 2.0 ships with TopLink Essentials as default JPA provider
  - Certified within OracleAS (OC4J)

TopLink Summary

- TopLink ORM
  - *Any* JDBC compliant data sources
  - *Any* Java EE compliant application server
  - *Any* IDE and toolset
  - JPA Support and Reference Implementation
- TopLink OXM
  - *Any* JAXP compliant parser
  - JAXB compliant API
- TopLink support in leading frameworks
  - Oracle ADF
  - Spring Framework

TopLink and ADF BC

- TopLink

  "Oracle9*i*AS TopLink provides an advanced Java persistence architecture for rapid development, deployment and execution of enterprise Java applications with relational databases."

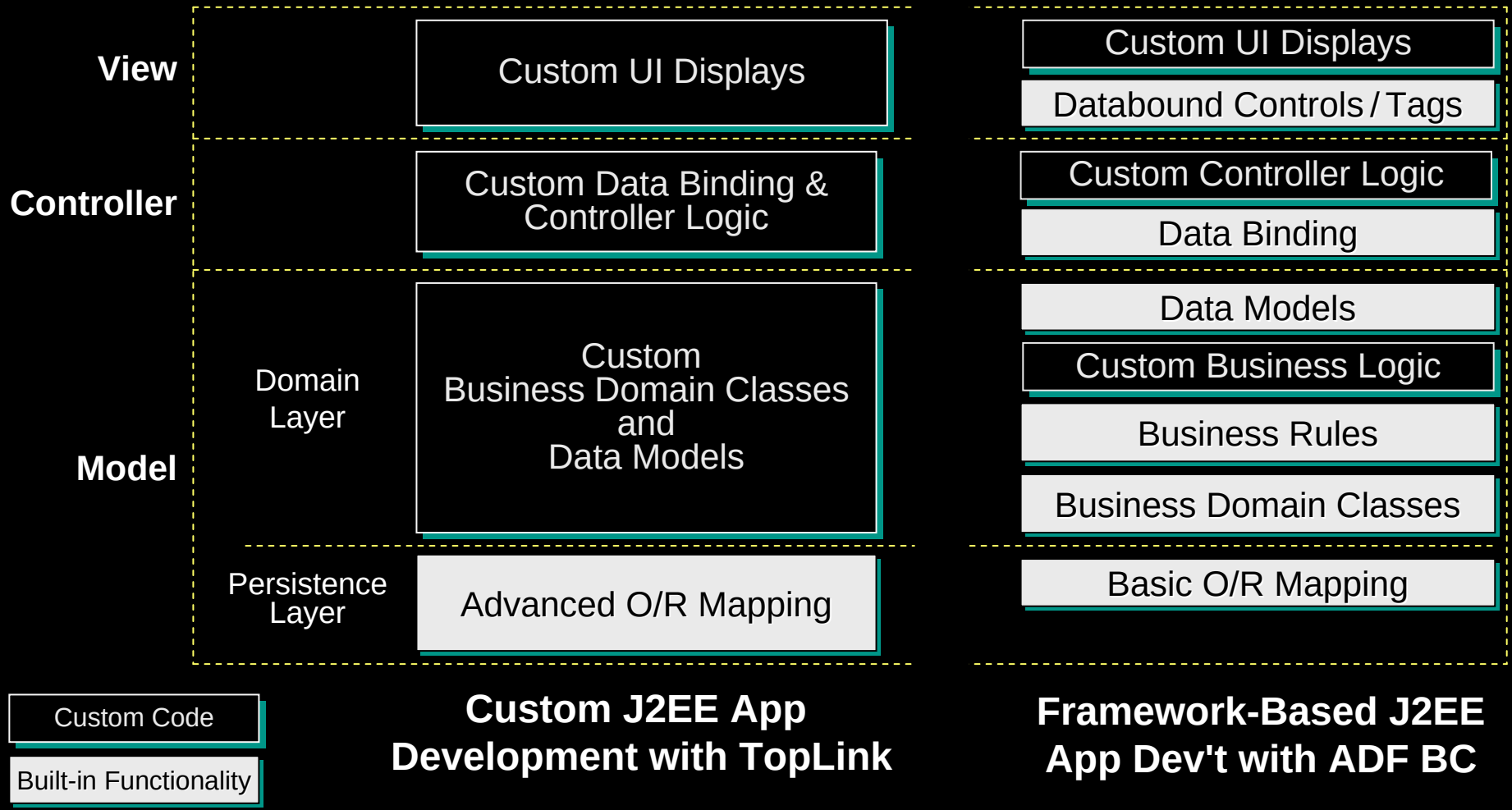
- ADF Business Components (BC4J)

  "ADF BC is an application development framework that simplifies delivering enterprise applications by generating functional business components that implement J2EE design patterns."

TopLink and ADF BC

- TopLink
  - Persistence Architecture
  - Maps Any Business Model to Any Schema
  - Simplifies Development of Persistence Infrastructure
  - Provides Advanced O/R Mapping Support
- ADF BC
  - J2EE Application Development Framework
  - Generates Functional Business Domain Classes & Data Models
  - Simplifies Building Web/Swing User Interfaces & Web Services
  - Provides Basic O/R Mapping Support

TopLink and ADF BC: Solutions For Any Development Approach



TopLink and ADF BC

- Both Products help developers build J2EE applications using relational databases
- Consider TopLink:
  - When you need a persistence architecture for your own application framework
  - Your team is familiar with OO/UML techniques
- Consider ADF BC :
  - When you need a complete application framework to avoid writing your own
  - Your team has Oracle Forms/Designer experience

Hvala na pažnji!
Za pitanja i odgovre nemamo više vremena.