

ORACLE®



Java EE 7 Overview and Status

Peter Doschkinow
Senior Java Architect

MAKE THE
FUTURE
JAVA

ORACLE

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.



Agenda

- Java EE 7 Revised Scope
- HTML5 Technologies
- Productivity Improvements
- Status and Roadmap

Java EE Past, Present, & Future

Productivity
& HTML5



Enterprise Java Platform	Robustness	
J2EE 1.2 Servlet, JSP, EJB, JMS, RMI/IIOP	J2EE 1.3 CMP, Connector Architecture	
Dec 1999 10 specs	Sep 2001 13 specs	
	Web Services	
	J2EE 1.4 Web Services Mgmt, Deployment, Async Connector	
	Nov 2003 20 specs	
	Ease of Development	
	Java EE 5 Ease of Development, Annotations, EJB 3.0, JPA, JSF, JAXB, JAX-WS, StAX, SAAJ	
	May 2006 23 specs	
	Lightweight	
	Java EE 6 Pruning, Extensibility Ease of Dev, CDI, JAX-RS	
	Web Profile Servlet 3.0, EJB 3.1 Lite	
	Dec 2009 28 specs	
	Java EE 7 JMS 2.0, Batch, Caching, TX Interceptor, WebSocket, JSON JAX-RPC, CMP/BMP, JSR-88	
	Web Profile JAX-RS 2.0	
	Q2 2013 32+ specs	

Java EE 7 Revised Scope

- PaaS theme postponed for Java EE 8
- HTML5 Support
 - WebSocket, JSON
 - HTML5 forms and markup
- Higher Productivity
 - Less Boilerplate
 - Richer Functionality
 - More Defaults



Java EE 7 – Candidate JSRs



unmodified

modified

substantially modified

new, to be included

PaaS Theme Postponed for Java EE 8

- Reasons
 - Not enough experience in tenants management, provisioning, deployment and elasticity implementation in Cloud environments
 - Not enough consensus and substance as of August 2012
 - Only roles, some service metadata and tenant-id location in JNDI specified
- CAMP initiative to by standardized by OASIS
 - Language-neutral standard self-service interface for PaaS environments
 - Supported by Oracle, Red Hat, CloudBees, Rackspace, Software AG, ...

Java EE 7 Community Feedback

Java EE is **already 90% cloud ready** due to its focus on clear packaging, deployment and portability ... When it comes to the last 10% of unmet cloud needs, we are **clearly in a time of experimentation, not yet a time of standardization...** Vendors innovate, collectively we standardize. We are not done innovating in this space.



David Blevins, Apache TomEE, Geronimo

...it is a **brave and honest decision** of the EG to realign Java EE 7 to the community needs and make a second try on Cloud and PaaS with the Java EE 8 Platform release scheduled for the spring of 2015...



*Markus Eisele, Java EE Consultant
<http://blog.eisele.net/>*



Java EE 7 Cloud features likely to be delayed until 2015 – your view?

Right call. Cloud standardization is a bad move with the market changing so much



52%

Speaking as a Java EE implementer, we (Red Hat) are **very much in support of this**. We've long advocated that we, the Java EE community **are not ready to standardize cloud yet**, and feel this is proven by OpenShift, our Java EE cloud offering, which is working well with Java EE 6.



redhat. Pete Muir, Redhat

WebSocket

- TCP based, bi-directional, full-duplex messaging
- Originally proposed as part of HTML5
- IETF-defined **Protocol**: RFC 6455
 - Handshake, data transfer
- W3C defined **JavaScript API**
 - Candidate Recommendation
- Supported by most browser already
<http://caniuse.com/websockets>



Web Sockets - Working Draft												
Bidirectional communication technology for web apps												
Resources: Wikipedia	Details on newer protocol			WebSockets information								
	IE	Firefox	Chrome	Safari	Opera	iOS Safari	Opera Mini	Android Browser	BlackBerry Browser	Opera Mobile	Chrome for Android	Firefox for Android
13 versions back		4.0	5.0									
18 versions back			6.0									
17 versions back			6.0									
16 versions back		2.0	7.0									
15 versions back		3.0	8.0									
14 versions back		3.5	9.0									
13 versions back		3.6	10.0									
12 versions back		4.0	11.0									
11 versions back		5.0	12.0									
10 versions back		6.0	13.0		9.0							
9 versions back		7.0	14.0		9.5-9.6							
8 versions back		8.0	15.0		10.0-10.1							
7 versions back		9.0	16.0		10.5							
6 versions back		10.0	17.0		10.6							
5 versions back		11.0	18.0	3.1	11.0							
4 versions back	5.5	12.0	19.0	3.2	11.1	3.2		2.1				
3 versions back	6.0	13.0	20.0	4.0	11.5	4.0-4.1		2.2		10.0		
2 versions back	7.0	14.0	21.0	5.0	11.6	4.2-4.3		2.3		11.0		
Previous version	8.0	15.0	22.0	5.1	12.0	5.0-5.1		3.0		11.1		
Current	9.0	16.0	23.0	6.0	12.1	6.0	5.0-7.0	4.1	7.0	12.0	18.0	15.0
Next future	10.0	17.0	24.0		12.5				10.0	12.1		



WebSocket Handshake and Communication

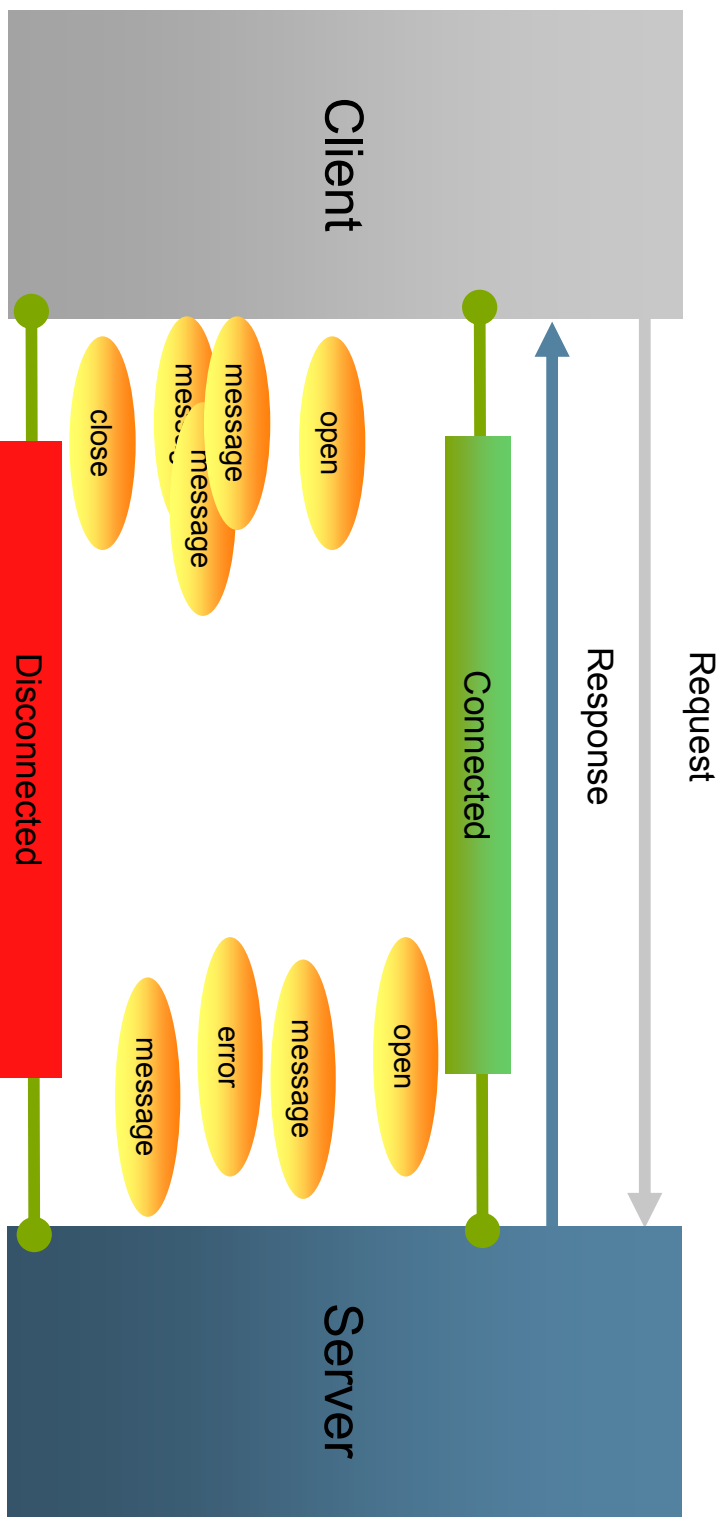
Handshake request

```
GET /chat HTTP/1.1
Host: server.example.com
Upgrade: websocket
Connection: Upgrade
Sec-WebSocket-Key: gBSZSbub25jZQ==
Origin: http://example.com
Sec-WebSocket-Protocol: chat, superchat
Sec-WebSocket-Version: 13
```

Handshake response

```
HTTP/1.1 101 Switching Protocols
Upgrade: websocket
Connection: Upgrade
Sec-WebSocket-Accept: s3pRbK+xOo=
Sec-WebSocket-Protocol: chat
```

WebSocket Handshake and Communication





Java API for WebSocket (JSR 356)

- Standard API for creating WebSocket Applications
- Specification
 - Early Draft Review
 - Part of Java EE 7
 - Under discussion: Client API in Java SE
- Reference implementation
 - <http://java.net/projects/tyrus>
 - Bundled in latest GlassFish 4.0 builds

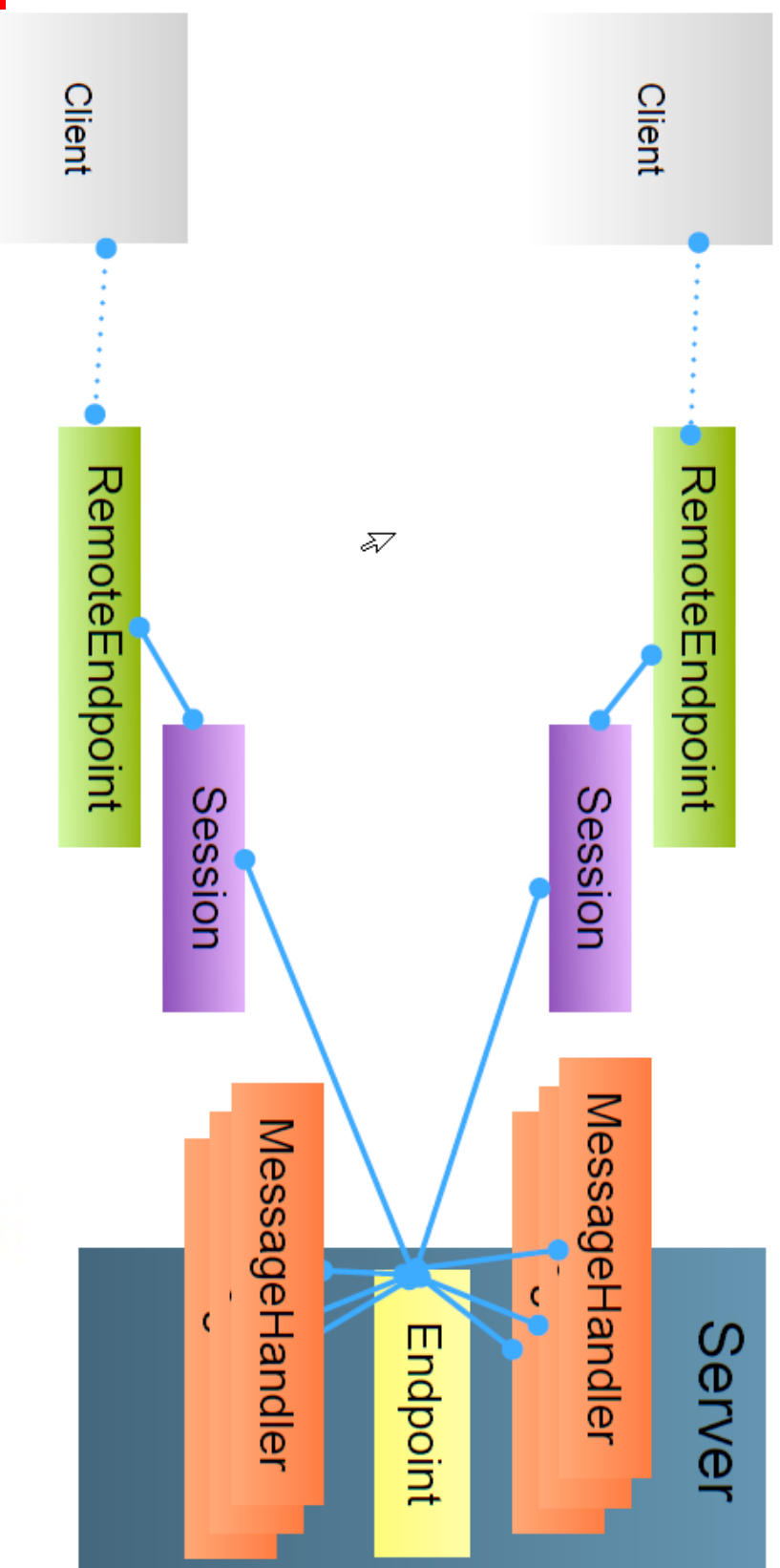


Java API for WebSocket Features

- Create WebSocket Endpoints
 - Annotation-driven (`@WebSocketEndpoint`)
 - Interface-driven (`Endpoint`)
- SPI for extensions and data frames
- Integration with Java EE Web container
- Client and server APIs

WebSocket

Server Side Object Model



WebSocket Chat Sample

```
@WebSocketEndpoint(path="/chat")
public class ChatBean {
    Set<Session> peers = Collections.synchronizedSet(...);
    @WebSocketOpen
    public void onOpen(Session peer) {
        peers.add(peer);
    }
    @WebSocketClose
    public void onClose(Session peer) {
        peers.remove(peer);
    }
    ...
    @WebSocketMessage
    public void message(String message, Session client) {
        for (Session peer : peers) {
            peer.getRemote().sendObject(message);
        }
    }
}
```




WebSocket Drawing Demo

- Cooperative painting with HTML5
- Used technologies
 - WebSocket, Server-Side Events, JSON, JavaScript (Angular.js)
- To participate
 - Connect to Peter7Links
 - <http://192.168.0.2:8080/drawingboard/>



Java API for JSON Processing

JSON 1.0

- API to parse, generate, transform, query, etc. JSON
- Object Model and Streaming API
 - Similar to DOM and StAX
- Binding JSON to Java objects forthcoming

Java API for JSON Processing

Build a JSON Object

```
"phoneNumber": [  
  {  
    "type": "home",  
    "number": "408-123-4567"  
  },  
  {  
    "type": "work",  
    "number": "408-987-6543"  
  }  
]
```

```
JsonObject jsonObject = new JsonBuilder()  
    .beginArray("phoneNumber")  
    .beginObject()  
    .add("type", "home")  
    .add("number", "408-123-4567")  
    .endObject()  
    .beginObject()  
    .add("type", "work")  
    .add("number", "408-987-6543")  
    .endObject()  
    .endArray()  
    .build();
```



Java API for RESTful Web Services

JAX-RS 2.0

- Client API
- Filter and interceptor framework
- Asynchronous processing – server and client
- Hypermedia support
- Integration with Bean Validation

Java API for RESTful Web Services

Client API

```
// get Client  
Client client = ClientFactory.newClient();
```

```
// get balance for a specific card with a PIN  
String balance = client.target("/atm/{cardId}/balance")  
    .pathParam("cardId", "123456")  
    .queryParam("pin", "3711")  
    .request("text/plain")  
    .get(String.class);
```



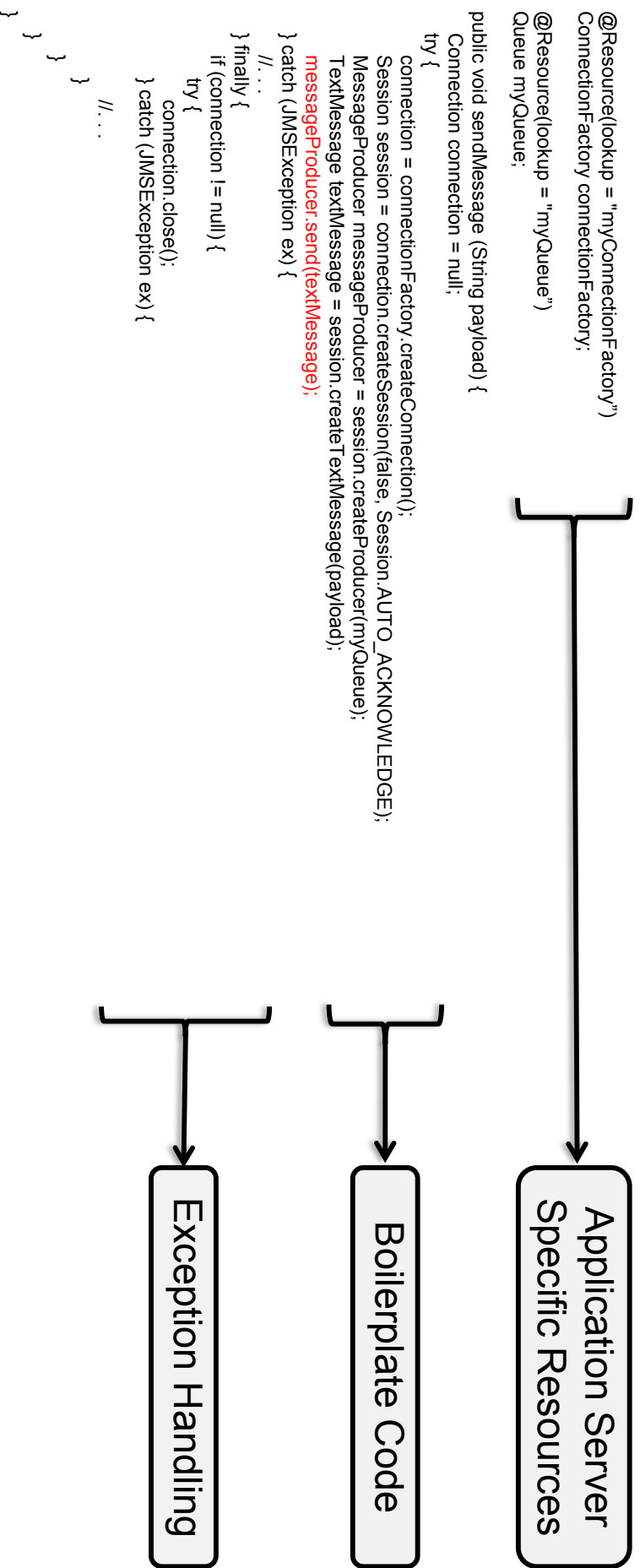
Java Message Service 2.0

Simplify the existing API

- Reduce boilerplate code
- API modernization using dependency injection
- JCA adapter for Java EE container required
- Connection, Session, and other objects are *AutoCloseable*

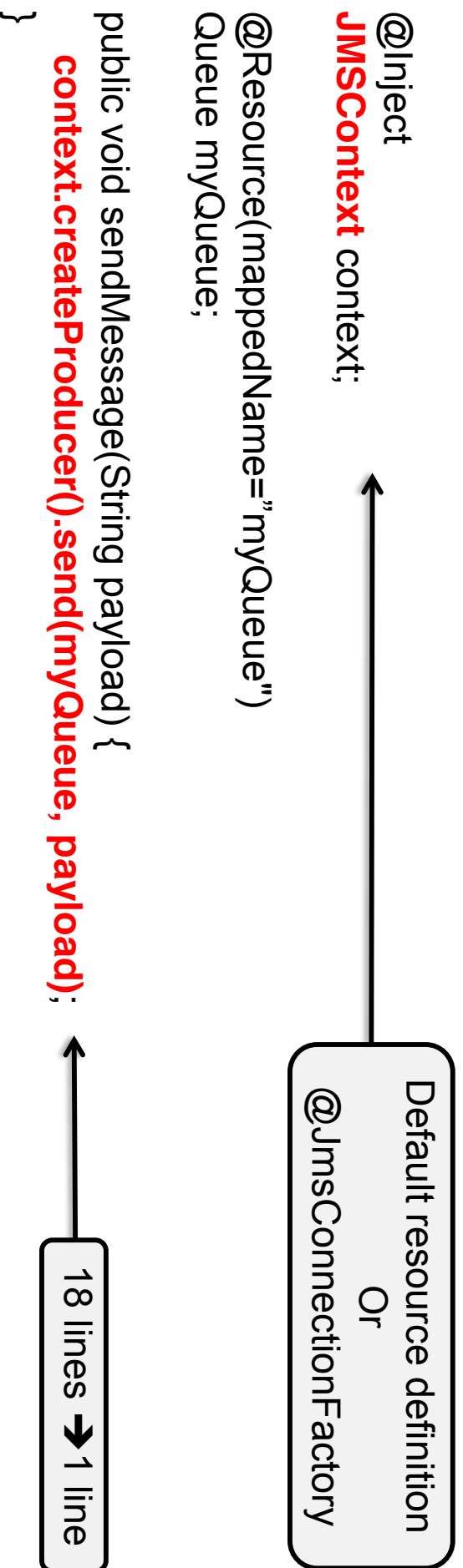
Java Message Service 2.0

Sending a Message using JMS 1.1



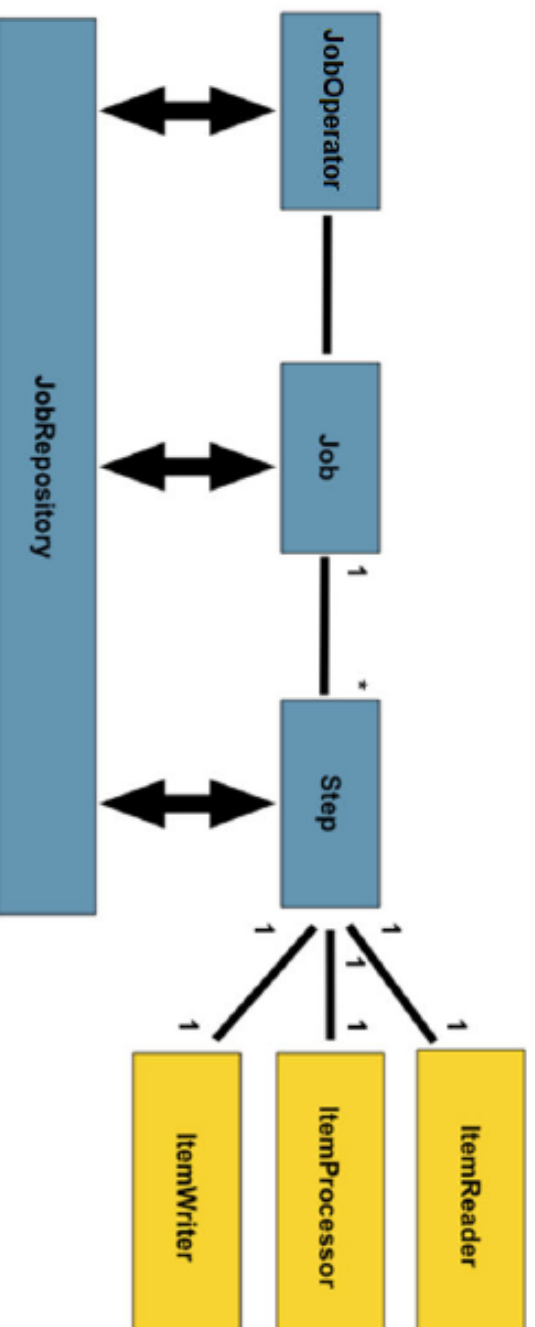
Java Message Service 2.0

Sending a Message using JMS 2.0 Simplified API



Batch Applications for the Java Platform 1.0

- API for robust batch processing targeted to Java EE



Batch Applications for the Java Platform 1.0

```
<step id="sendStatements">
  <chunk reader="AccountReader"
    writer="EmailWriter"
    processor="AccountProcessor"
    chunk-size="10" />
</step>
```

@ReadItem

```
public Account readAccount() {
    // read account using JPA
}
```

@ProcessItem

```
public Account processAccount(Account account) {
    // calculate balance
}
```

@WriteItems

```
public void sendEmail(List<Account> accounts) {
    // use JavaMail to send email
}
```

Java Temporary Caching API

JCache 1.0

- API and semantics for temporary, in-memory caching of Java objects
- Basic API:

```
CacheManager cacheManager =  
CacheManagerFactory.getCacheManager();  
Cache<String, Object> cache = cacheManager.getCache("testCache");  
...  
cache.put(key1, value1);  
...  
Object value2 = cache.get(key2);  
...  
cache.remove(key3);
```

Java Temporary Caching API

Caching Annotations

```
public class CustomerDao {  
    @CachePut(cacheName="customers")  
    public void addCustomer(@CacheKeyParam int cid,  
                            @CacheValue Customer customer) {  
        ...  
    }  
}
```



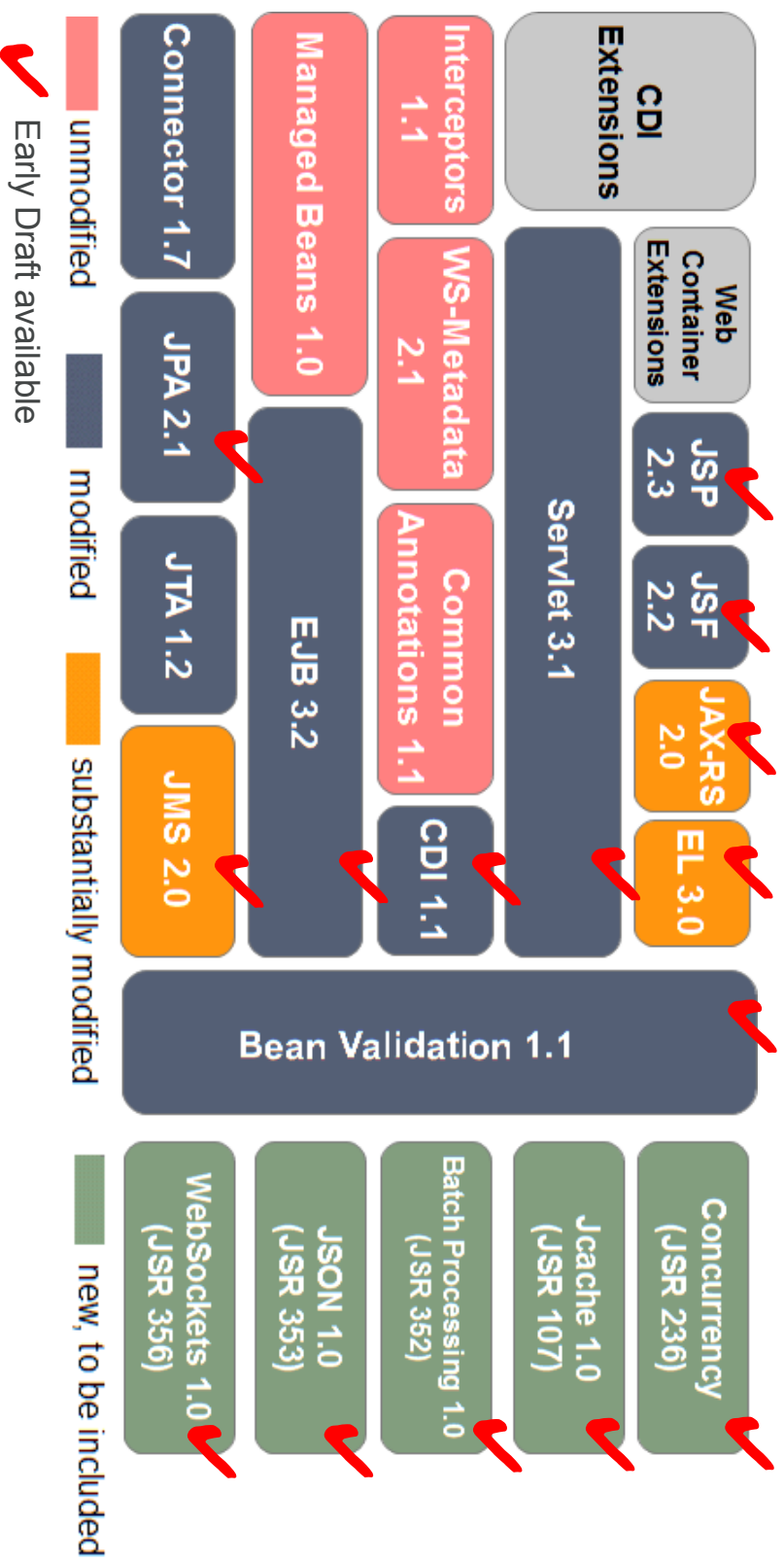
Many other Improvements

- **JPA 2.1**: Schema generation, unsynchronized PC, converters, ...
- **EJB 3.2**: Optional CMP/BMP, Ease-of-use, Refactoring, ...
- **JSF 2.2**: Faces Flows, HTML5 markup support
- **Servlet 3.1**: Non-blocking IO, Upgrade to WebSocket, ...
- **EL 3.0**: Lambda expressions, Collection, Operators, ...
- **JTA 1.2**: Transactional interceptors
- **CDI 1.1**: Ordering of interceptors, Servlet events, ...



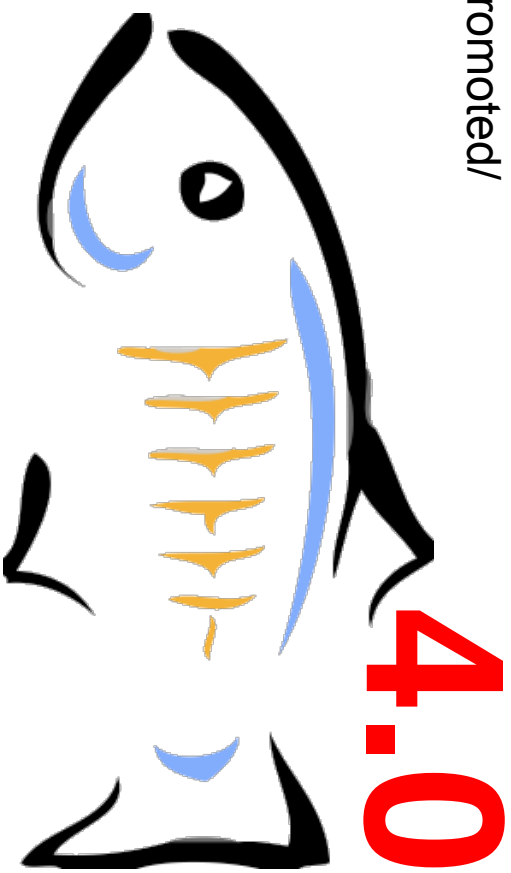
Java EE 7 – Specification Status







https://blogs.oracle.com/arungupta/entry/java_ee_7_status_update



Java EE 7 – Implementation Status

<http://download.java.net/glassfish/4.0/promoted/>



	glassfish-4.0-b62-ml.zip	07-Nov-2012 14:49	96M
	glassfish-4.0-b62-unix-ml.sh	07-Nov-2012 14:53	65M
	glassfish-4.0-b62-unix.sh	07-Nov-2012 14:33	62M
	glassfish-4.0-b62-windows-ml.exe	07-Nov-2012 14:57	65M
	glassfish-4.0-b62-windows.exe	07-Nov-2012 14:37	62M
	glassfish-4.0-b62.zip	07-Nov-2012 14:29	92M

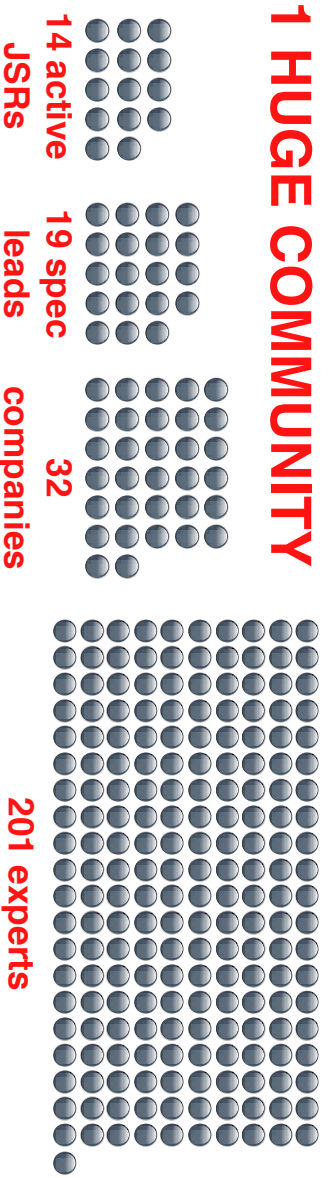
Java EE 7 Community



Java
Community
Process

Java EE 7

1 HUGE COMMUNITY



java.net

Java EE 7



23 active projects, active
and transparent mailing lists



GlassFish



2 Releases
shipped



62 promoted builds



Java EE 7 Opinion Survey

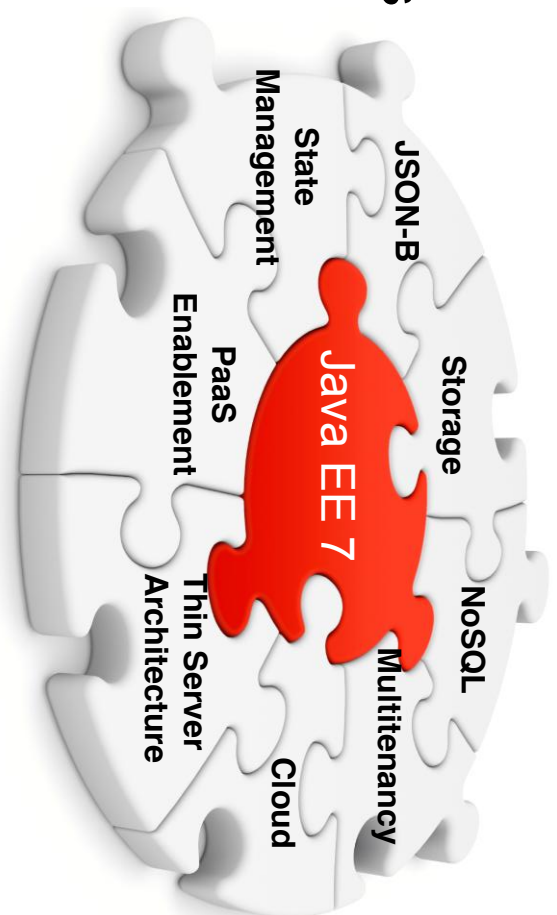
<https://www.surveymonkey.com/s/javaee7>

- Which of the new API to include in Java EE 7 Full / Web Profile
- CDI support
 - Enabled by default for all jar files?
 - Should all Java EE API stick to CDI @Inject instead of using their own injection mechanisms?
 - Expanding the use of @Stereotype to non-CDI annotations?
 - Using interceptors on all Java EE components?

Java EE 8 and Beyond

Standards-based cloud programming model

- Deliver cloud architecture
- Multi tenancy for SaaS applications
- Incremental delivery of JSRs
- Modularity based on Jigsaw





Project Avatar

- Applications based on Thin Server Architecture
- Incubator for HTML5/JavaScript clients with Java EE
 - Learn by experimenting
 - Identify targets for standardization, e.g. WebSocket (now), Server-Sent Event (Future)
- End-to-end framework: Client and Server
 - A-la-carte, not all or nothing
- Tooling: Project Easel



Call to Action

- Java EE & GlassFish @ JavaOne 2012
 - glassfish.org/javaone2012
- Java EE 7 transparent Expert Group
 - javaee-spec.java.net
- Java EE 7 Reference Implementation
 - glassfish.org
- The Aquarium
 - blogs.oracle.com/theaquarium

Hardware and Software

The Oracle logo consists of a solid red horizontal bar. Centered within this bar is the word "ORACLE" in a white, uppercase, sans-serif font. A small registered trademark symbol (®) is located at the end of the word.

ORACLE®

Engineered to Work Together

ORACLE®