

Participate in the Future of Java - How the JCP Works

Heather VanCura

March 2014

<http://jcp.org>

Twitter @heathervc @JCP_org

Agenda

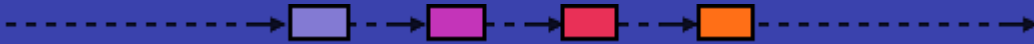


- What is the JCP Program
- How & Why to Participate
- Adopt-a-JSR Program

How the JCP works



For the *community*, by the *community*



- The Java Community Process (JCP) is the open, inclusive process to develop and revise Java technology specifications (JSRs), reference implementations (RI), and technology compatibility kits (TCK).
 - The JCP program now has over 1,000 corporate, individual and Java User Group (JUG) members.
 - More than 350 Java technology specifications are in development in the JCP program; ~two thirds have reached Final Release.
 - Java EE 7 Final Release 2013.
 - Java SE 8 Final Release 2014.
 - Java ME 8 in Proposed Final Draft.

Deliverables



How we do it



- Java Specification Requests (JSRs)
 - A JSR is a single version of a Java specification.
 - JSRs are led by a community member (the *Spec Lead*), with a group of interested members (the *Expert Group*) helping with the day-to-day decisions and work.
 - Any community member can submit and lead a JSR.
- Each Expert Group must deliver:
 - The Specification
 - The Reference Implementation (RI)
 - The Technology Compatibility Kit (TCK)

Compatibility testing



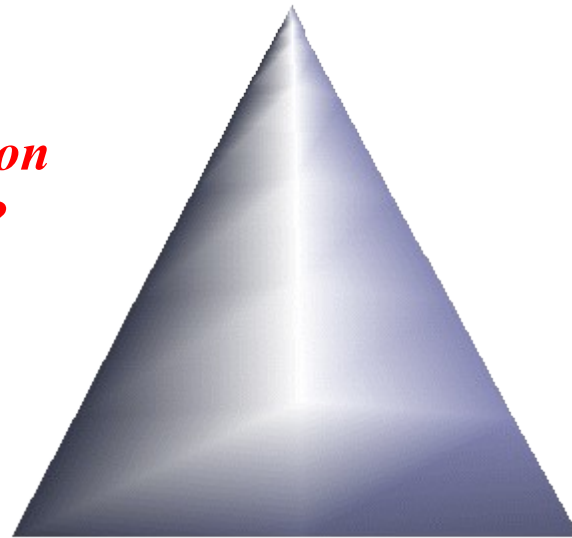
The compatibility triangle



Specification

*Is the specification
unambiguous?*

*Can you build an
implementation?*

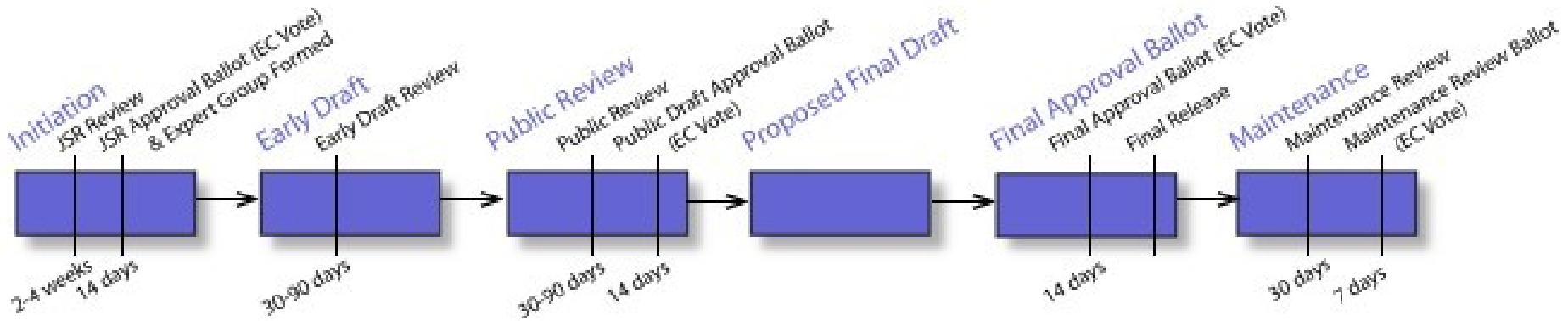


**Technology
Compatibility Kit**

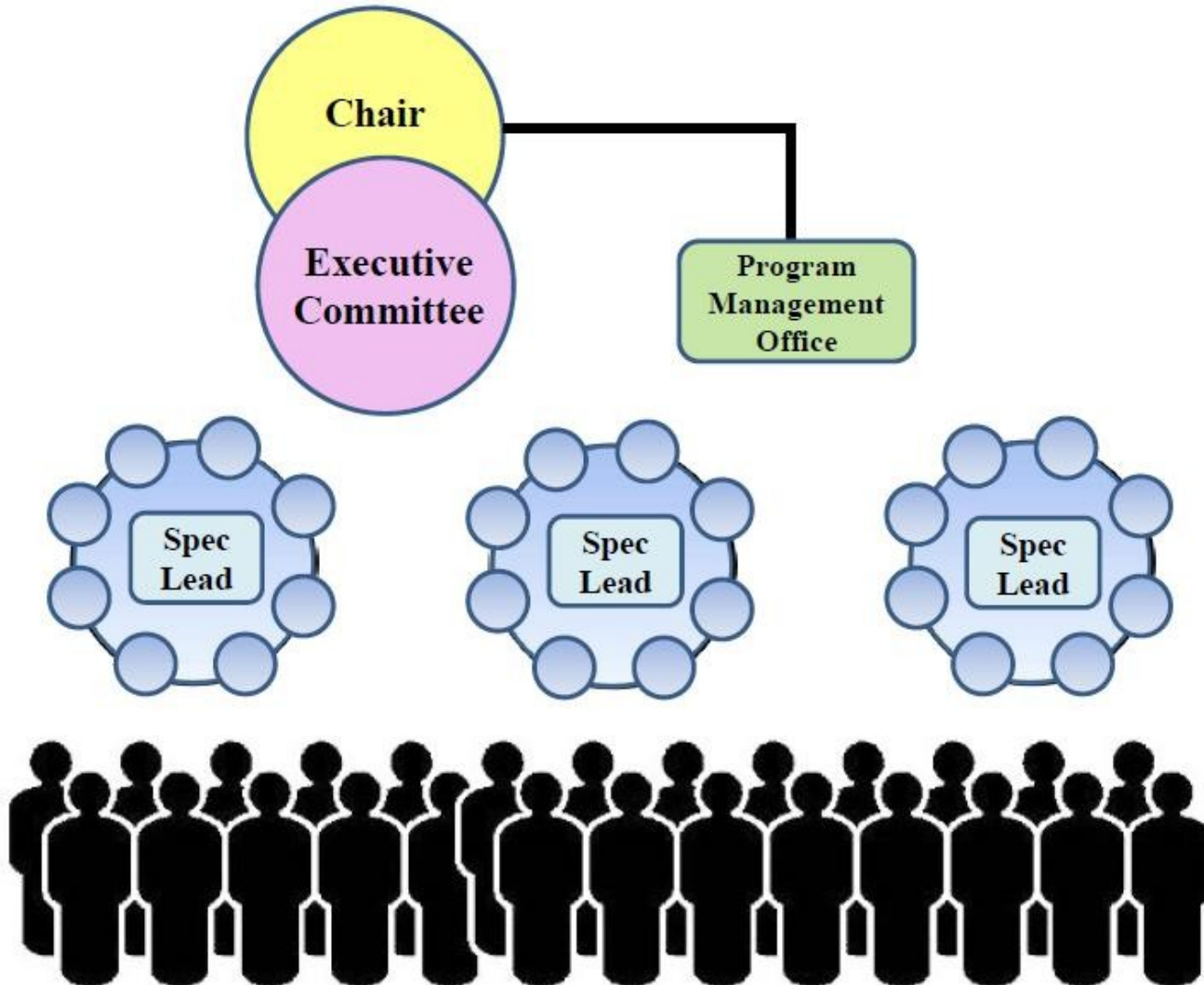
**Reference
Implementation**

*Is the TCK correct?
Does the RI conform?*

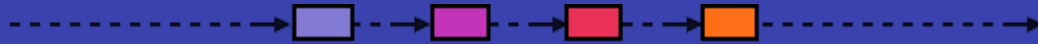
The JSR development cycle



Organization



Roles



- JCP Chair
 - Leads the organization and manages the PMO.
- Program Management Office (PMO)
 - Manages day-to-day operations of the organization.
- Executive Committee
 - Define JCP governance, processes, and contractual terms of membership.
 - Vote on JSRs at defined stages through the process.
- Expert Groups
 - Create JSRs (write the spec, develop the RI and TCK.)
- Members
 - Review specs, may participate in Expert Groups, vote in Executive Committee elections.

The Executive Committee



- Executive Committee (EC):
 - One EC for Java ME, Java SE and Java EE together; EC combined in 2012-2013.
 - Annual elections reconfirm or replace one half of the members.
- EC meets ~monthly by phone, 3 times a year face-to-face.
- EC meeting minutes & materials are public.
 - See http://jcp.org/en/resources/EC_summaries.
- We hold two public teleconferences and one public f2f meeting each year.
- We have a public mailing list for feedback.
 - Sign up at <http://java.net/projects/jcp-ec/lists>.

Who are the members?



- Membership distribution by type:
 - 88% individual.
 - 8% corporate.
 - 4% non-profit.
- Membership distribution by location:
 - 47% North America
 - 34% Europe and the Russian Federation
 - 13% Asia and the Middle East
 - 6% South America

Levels of participation



- Public Observer.
- Members.
 - Review specs, may participate in Expert Groups, vote in Executive Committee elections; flexible time commitment.
- Expert Group members.
 - Create JSRs (write the spec, develop the RI and TCK); expert in field, high time commitment, help drive the direction of the JSR; moderate time commitment.
- Specification Leads.
 - Responsible for leading the JSR & Expert Group (EG), delivery of spec, RI & TCK; a leading expert in area of JSR; considerable time commitment and recognition.
- Executive Committee members.
 - Define JCP governance, processes, and contractual terms of membership. Vote on JSRs at defined stages through the process. Moderate time commitment.

Legal framework and governance



- Java Specification Participation Agreement (JSPA)
 - A legal contract between members and Oracle.
 - Addresses Intellectual Property (IP) grants and the terms under which the spec, RI, and TCK should be licensed.
 - <http://www.jcp.org/aboutJava/communityprocess/JSPA2.pdf>.
- The Process Document
 - Defines the governance of the organization.
 - Defines the processes that are used to submit define, develop, review, approve, and maintain specifications.
 - Defines the obligations to produce an RI and TCK.
 - <http://jcp.org/en/procedures/jcp2>.

Constitutional change (JCP.next)



JCP.next



- A series of three JSRs:
- *JSR 348: Towards a new version of the Java Community Process* completed in October 2011.
- This JSR focused on relatively simple changes that we were able to implement within about six months.
- *JSR 355: Executive Committee Merge* will be complete in October 2013.
- *JSR 358: A major revision of the Java Community Process* is in progress now.
 - This JSR will implement more complex changes and will modify the JSPA as well as the Process Document.

All the lawyers will get involved



Results



- JSR 348 introduced version 2.8 of the Process.
- JSR 355 introduced version 2.9 of the Process, merged the two ECs into one EC.
- It is now easier for JCP members and the public to observe and participate in the work of Expert Groups.
- Transparency is the default mode of operation.
- All new JSRs will be bound by the new rules.
 - Existing JSRs voluntarily adopt them—almost all Active JSRs have.
- Now we need your participation!

That was easy!



JCP is now more open than before



- Public JSR schedule--current, and updated regularly.
- Public read and/or write page/wiki for the JSR.
- Public discussion for the work of the JSR (archived).
- Public feedback mechanism.
- Public issue tracker.
- One EC votes on all JSRs 3x in life of JSR with public results.
- All JSRs have 3 open public review periods (30 days).
 - Release early and often!

Get involved!



Why should you get involved?



- To move the entire Java ecosystem forward.
 - Help define the future of Java – best way to predict it.
- To speed up the standards-development process.
- To become part of the solution as opposed to staying silent.
- To be at the forefront of new technology.
- To ensure that Java standards are practical and relevant for the real world as opposed to the ivory tower.
 - We need expert input from end-users and developers.
- To ensure that Java standards are truly global.
 - We need expert input from non-English speakers.

What do you get?



- As a corporation:
 - Influence over the technologies that drive your market.
 - Savings - it's cheaper than developing everything yourself.
 - Opportunities for your developers.
 - Increase in the size of your market.
 - Competitive advantage (ship new products faster.)
- As an individual:
 - Acquire knowledge: Learn from experts, early access.
 - Better specifications based on real world experience.
 - Experience and skill development.
 - Professional Visibility: Curriculum, articles, presentations.
 - Fame and fortune – grow your reputation and network.

Join the JCP



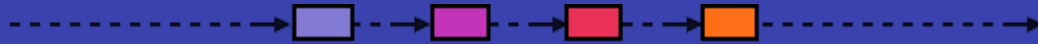
- Join the JCP program as a corporation, non-profit, or individual member.
 - Free for non-profits , Java User Groups, and individuals!
- See **<http://jcp.org/en/participation/membership2>**
- Register for an account on JCP.org to become a public observer participant.
- Download, fill out, and scan, fax or mail the JSPA form to the JCP Program Office to become a JCP Member.

Transparency survey



- Transparency surveys of JSRs hosted on java.net.
 - <http://www.java.net/form/jcp-2829-transparency>
- Provide qualitative and quantitative data to the community, Spec Leads, and the Program Office on compliance with the JCP's transparency and participation requirements.
- A matrix of results will be provided:
 - Once responses are received :-).

Community-provided assessment



- Learn about the JSR and share your input:
 - The quality and utility of the public issue tracker.
 - How the Spec Lead and Expert Group communicate and interact on the JSR's public mailing list. Are issues being resolved and addressed?
 - Whether the Expert Group represents all relevant sectors of the Java community.
 - Whether the Expert Group has published public drafts of the JSR and whether this represents a convergence of competing implementations.

Suggestions for Feedback



- Download and review the latest version of the specification.
 - Available from the JSR page or the corresponding java.net or GitHub project.
 - You don't need to necessarily review the entire specification; can focus on specific sections.
- Download and review the javadocs for clarity and completeness—comment.
- Build sample applications using the implementations.
- Explore integration of one/multiple technologies with your existing applications.
- File bugs on the corresponding spec or RI project.

How will you participate?



- As an individual – OK.
- As part of a team – better.
- Work through your JUG or employer.
 - Help each other.
 - Teach each other.
 - Work with each other.



JUG members ~ 45 JUGs



BeJUG (Belgium)

BreizhJUG (Brittany)

CeJUG (Brazil)

Central Ohio JUG (USA)

Connecticut JUG (USA)

Detroit JUG (USA)

Duchess (Women)

FASOJUG (Burkina Faso)

Green Tea JUG (China)

Houston JUG (USA)

iJUG e.V. (Germany)

IndiJava (India)

Japan JUG

Java Student User Group (Vienna)

Java Web User Group (London)

Jozi JUG (South Africa)

JUG-AFRICA

JUG Chennai (India)

JUG Cologne (Germany)

JUG-EG (Egypt)

JUGF (Germany)

JUG Indonesia

JUG-MK (Macedonia)

JUG-RU (Russia)

JUG-USA

London Java Community (UK)

Malaysia-JUG

MoroccoJUG

Nashville JUG (United States)

Oklahoma City JUG (USA)

Rio JUG (Brazil)

Riviera JUG (France)

Ruhrjug (Germany)

Silicon Valley JavaFX User Group (USA)

SouJava (Brazil)

Utah JUG (USA)

Vancouver Island JUG (Canada)

Adopt a JSR!



Thank you **LJC** and **SOU Java**
sociedade de usuários java

What is it?



- A JUG-lead initiative to improve Java standards.
- Groups of JUG members work on JSRs.
- See <http://java.net/projects/adoptajsr> for full details.

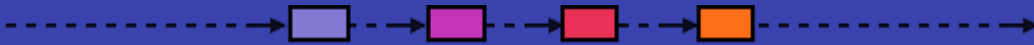


How Will it Help your JUG?



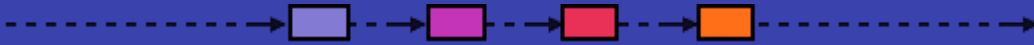
- Receive attention from industry.
 - Big vendors, local companies see the value of all your efforts.
- Become more professional.
 - Participation requires organization and commitment—this helps promote focus in your group.
- Attract people.
 - Developers working together has a global impact, and reaps personal benefits.
- Collaborate with other JUGs.
 - Be part of something larger!

How to get started on Adopt-a-JSR



- Join the JCP as a JUG.
 - Not required, but recommended. You need to do this in order to contribute code, and you get voting rights and other benefits with this membership!
- Join the Adopt-a-JSR project:
 - <http://java.net/projects/adoptajsr/pages/Home>
 - Subscribe to members@adoptajsr.java.net (archive).
- Update the wiki to indicate your interest in a particular JSR and create a wiki page to provide updates on your plans/progress.
- Communicate directly with the Specification Lead/Expert Group (links on jcp.org) or the java.net users alias for the specification.
- Mail your JUG members to gather interest.
- Outline what the JSR is and why the JUG should get involved; identify areas that you want to work on; coordinate this with the Spec Lead
- Share success on the wiki, list, twitter, etc.

How can you help (1)?



- Help to moderate the mailing lists; steer towards helpful conversations
- Evangelize the JSR through social media, blogging or lightning talks.
 - Give a talk at your user group at events, or online.
- Help with documentation.
 - Translate into your native language.
- Help to maintain the FAQ/Wiki.
- Improve project infrastructure and JSR visibility.
 - Help set up JSR hosting (java.net, GitHub, etc.)
 - Help with Search Engine Optimization of website.
 - Ensure that downloads, mailing lists, issue trackers are easy to find.

How else can you help (2)?



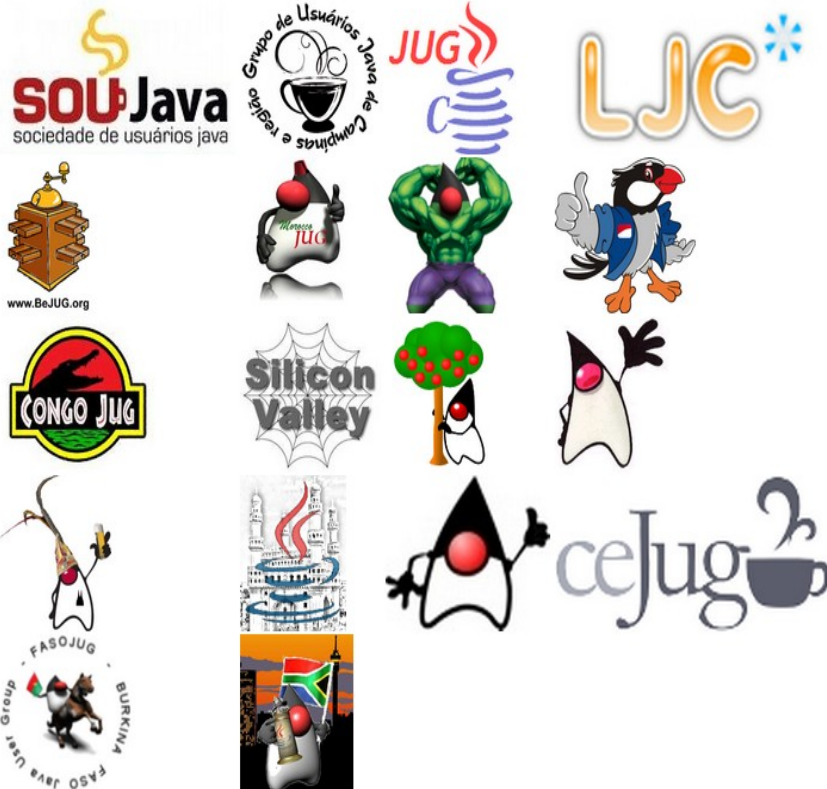
- Organize Hack days.
- Test the early RI builds; use them to find pain points, report bugs, suggest feature enhancements.
 - Help triage issues; reproduce issues, erase/merge duplicates, set priorities/categories etc.
- Give feedback on design; discuss issues and deliver feedback, think about how you would use the JSR as a developer.
- Help build the RI; get coding with the actual implementation of the spec or build sample applications.
- Help build the TCK; all implementations must pass the test suite – this is a great way to gain test experience.

Guidelines for Feedback



- The specification lead and the Expert Group make decision on incorporating the feedback.
- For specifications in Public Drafts, new feature requests may not be considered for this release. For specifications in Early Draft, new features may be accepted if they are within the existing scope. If new features are not accepted for this release, then they may be considered for the next release.
- Multiple JUGs can and should collaborate on a particular JSR. There is always plenty of work and going through the material multiple times only makes it that more better.
- When communicating with the EG on mailing lists, include "Adopt-a-JSR" in the subject line.
- When filing a JIRA issue, use "adoptajsr" tag.
- Use twitter hashtag #adoptajsr to give updates on progress, cc@jcp_org.

JUGS Around the World!



Adopt-a-JSR Adopters



- 26+ JUGS participating:
 - South America: Campinas JUG (Brazil), ceJUG (Brazil), GOJava (Brazil), SouJava (Brazil), Brasilia, DF JUG (Brazil), GUJ Londrina (Brazil) Peru JUG, Guadalajara JUG.
 - North America: Silicon Valley JUG, Houston JUG (USA), Toronto JUG (Canada).
 - Europe: London Java Community (UK), BeJUG (Belgium), Madrid JUG (Spain), JUG Cologne (Germany), Istanbul JUG (Turkey).
 - Asia: Indonesia JUG, JUG JogLoSemar (Indonesia), Hyderabad JUG (India), JUG Chennai (India).
 - Africa & Middle East: EGJUG (Egypt), Morocco JUG, Faso JUG (Africa), Jozi JUG (South Africa), Mbale (Uganda).

Adopt-a-JSR Adopters



- 24 JSRs 'adopted':
- Java SE: JSR 308, 310, 335, 337.
- Java EE: 107, 299, 338, 339, 342, 343, 344, 345, 346, 349, 352, 353, 356.
- JCP.Next: JSR 348, 355, 358.
- Misc.: JSR 321, 331, 347, 354.
- Most popular: JSR 339, Java API for RESTful Web Services 2.0, JSR 344, JavaServer Faces 2.2 & JSR 353, Java API for JSON Processing, being adopted by at least 5 JUGs.

Use Cases of Adopters



- BeJUG: JSR 356, Java API for WebSocket - Hack day in March 2013/developed Tic-Tac-Toe game included in Java EE 7 SDK.
- Chennai JUG: JSR 338, Java Persistence 2.1, JSR 344, JSF 2.2, JSR 352, Batch, JSR 353, JSON, JSR 356, WebSocket - Urban Traveler application; JSR 354, Money & Currency API.
- London Java Community: JSR 310, Date and Time API and JSR 335, Lambda Expressions (Martijn Verburg); myriad hack days, talks and events organized!.
- SouJava: JSRs 310, 339, 344, and other JSRs; talks, books, articles, classes, code dojos:
 - <http://soujava.org.br/servicos/adopt-a-jsr/>

Putting the *community* back into the JCP



- No more barriers to participation.
- All members of the Java community can participate:
 - In the implementation of the platform through OpenJDK and GlassFish.
 - In the evolution of the platform through the JCP and Adopt-a-JSR programs.
- If you care about the future of Java...participate!

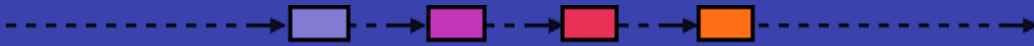
Java EE 7 JSRs - specific requests



- Nine of the Java EE 7 JSRs have specific requests from Adopt-a-JSR participants.
 - <http://glassfish.java.net/adoptajsr/>
- JSR 236, Concurrency Utilities for Java EE.
- JSR 339, Java API for RESTful Web Services.
- JSR 343, Java Message Service 2.0.
- JSR 344, JavaServer Faces 2.2.
- JSR 345, Enterprise JavaBeans 3.2
- JSR 346, Contexts and Dependency Injection for Java EE 1.1.
- JSR 352, Batch Applications for the Java Platform.
- JSR 353, Java API for JSON Processing.
- JSR 356, Java API for WebSocket.



Participate in OpenJDK



- Go to **<http://java.net/projects/adoptopenjdk/>**
- Join the mailing list.
- Join the IRC channel.
- Or contact your local JUG leader.

OpenJDK

Create the future with Java!



Follow the JCP online!

<http://jcp.org>

[Blogs.oracle.com/jcp](http://blogs.oracle.com/jcp)

[@JCP_org](https://twitter.com/JCP_org)

