Java Community Process

Community Participation in the JCP Program: a winning combination

heather@jcp.org
http://jcp.org

November 2012

Java Community Process

JCP.next.1 (JSR 348)

- JCP 2.8, implemented in October 2011
- Three themes aimed at:
- Transparency
- Participation
- Agility

Java Community Process

Transparency

Java Community Process

EG transparency – trust and security

- The Expert Group (EG) must do all substantive business on a public mailing list.
- The EG must track issues in a public issue tracker.
- Members of the public must be able to comment on the EG's work.
- The EG must publicly respond to all comments.

Java Community Process

Executive Committee transparency

- Must hold semi-annual teleconferences and an annual open meeting at JavaOne.
  - All JCP members are free to attend these meetings.
  - Agenda will be chosen from topics suggested by members.
- Must create a public alias with archive for members to provide feedback to the ECs.
- Private and non-normative EC policies and procedures made public and normative in new EC Standing Rules.
- Escalation and Appeal process defined.
  - Spec Leads, EG members and JCP members can appeal to EC for help in resolving disputes.

Java Community Process

Participation

Java Community Process

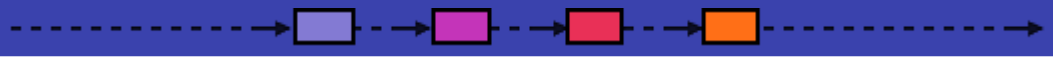
Participation – hope and action

- Requests to join EGs, the Spec Lead's responses, and decisions to remove or replace EG members, must be reported on the EG's public alias.
- Better processes for dealing with uncooperative, unresponsive, or disruptive EG members and Spec Leads.
- EC members who miss two consecutive meetings lose their voting privileges until they have attended two.
- EC members who miss 5 meetings in a row or 2/3 of the meetings in a 12 month period lose their seat.

Agility

Java Community Process

Agility – measure in weeks & months



- Time-outs for inactive JSRs.
  - Must reach Early Draft within 9 months, Public Draft within 12 months of EDR, Final Release within 12 months of PR, else can be up for Renewal Ballot (withdrawn by EC vote).
- Simplifies the Maintenance Release process.
- Clarifies the Final Release and Maintenance processes to ensure that completed/updated Spec, RI, and TCK are posted promptly.
- Ensures that links to RI and TCK are maintained.
  - If broken and not fixed, JSR must revert to “incomplete” stage.

Java Community Process

Results...

- It is now easier for JCP members and the public to observe and participate in the work of Expert Groups.
- Transparency is the expected and required mode of operation.
- All new JSRs will be bound by the new rules.
  - Existing JSRs will be encouraged to voluntarily adopt them.
- Further changes will be introduced in follow-on JSRs.
- Now we need your participation!

Java Community Process

JCP.next.2 (JSR 355)

Java
Community
Process

JCP.next.3

Java Community Process

JCP.next.3 (JSR 358)

- The last of the three JCP.next JSRs, JSR 358, is in process now.
- This will tackle more complex issues such as licensing, IP flow, and governance, and will modify the JSPA.
- This is **a big deal** (because all the lawyers will get involved.)
  - It will take a long time.

Java Community Process

JCP 2.8 JSRs

- JCP version 2.8 was introduced in October 2011
  - Merged EC, JCP 2.9, will take effect following the 2012 EC Elections in November 2012.
  - JSR Deadlines introduced with JCP 2.8: http://jcp.org/en/procedures/jcp2#1.3
    - If a JSR does not begin Early Draft Review within 9 months of completing its JSR Approval Ballot, or does not begin Public Review within 12 months of first submitting an Early Draft, or does not reach Final Release within 12 months of commencing Public Review, then the EC should initiate a JSR Renewal Ballot unless it is agreed that there are extraordinary circumstances that justify the delay. The PMO shall inform the Spec Lead and Expert Group of this decision and will request the Spec Lead and Expert Group to prepare a public statement to the EC. The JSR Renewal Ballot shall start 30 days after the request. If the JSR Renewal Ballot is approved by the EC, then another renewal ballot cannot be initiated for that JSR for an additional year.
  - For migrated JSRs: JSR Deadlines take effect (clock starts) when JSR migrates

JCP 2.8 Stats

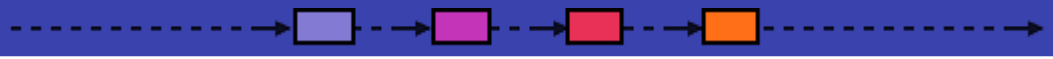
- JCP has 30 Active/In Progress JSRs
- 22 Active JSRs operating under version 2.8
  - No Renewal Ballots yet
- 13 JSRs have migrated to JCP 2.8
- 10 JSRs initiated under JCP 2.8 (1 rejected)
- 1 JSR in process of migration to JCP 2.8
- 8 in-flight Active JSRs have not yet migrated
  - Excludes those that have already posted FR or MR
  - The PMO encourages Spec Leads to migrate at milestone postings

List of new 2.8 JSRs

- JSR 352, Batch Applications for the Java Platform (IBM) – PR, November 2012
- JSR 353, Java API for JSON Processing (Oracle) – EDR, September 2012
- JSR 354, Money and Currency API (Credit Suisse) – JSR Review, February 2012
- JSR 355, EC Merge (Oracle) – Final Release, August 2012
- JSR 356, Java API for WebSocket (Oracle) – EDR, September 2012
- JSR 358, A major revision of the Java Community Process (Oracle) – JSR Review, July 2012
- JSR 359, SIP Servlet 2.0 (Oracle) – JSR Review, July 2012

Java Community Process

List of new 2.8 JSRs



- JSR 360, Connected Limited Device Configuration 8 (Oracle) – JSR Review, October 2012
- JSR 361, Java ME Embedded Profile (Oracle) – JSR Review, October 2012

Java Community Process

List of migrated 2.8 JSRs

- JSR 172, J2ME Web Services Specification (Oracle) – Maintenance Release, October 2011
- JSR 236, Concurrency Utilities for Java EE (Oracle) – EG Formation/2.8 move April 2012
- JSR 338, Java Persistence 2.1 (Oracle) – EDR2, November 2012
- JSR 339, JAX-RS 2.0: The Java API for RESTful Web Services (Oracle) – PR, October 2012
- JSR 340, Java Servlet 3.1 Specification (Oracle) – EDR, August 2012
- JSR 341, Expression Language 3.0 (Oracle) – Public Review, August 2012
- JSR 343, Java Message Service 2.0 (Oracle) – EDR, March 2012
- JSR 344, JavaServer Faces 2.2 (Oracle) – EDR, December 2011
- JSR 345, Enterprise JavaBeans 3.2 (Oracle) – EDR, February 2012
- JSR 346, Contexts and Dependency Injection for Java EE 1.1 (RedHat) – PR, November 2012

Java Community Process

List of migrated 2.8 JSRs

- JSR 335, Lambda Expressions for the JavaProgramming Language (Oracle) – EDR 2, migrated November 2012
- JSR 337, Java SE 8 Release Contents (Oracle) – EG Formation, migrated September 2012
- JSR 308, Annotations on Java Types (M Ernst, Oracle) – EDR 2, migrated September 2012

Java Community Process

JSRs in process of migrating

- JSR 342, JavaPlatform, Enterprise Edition 7 (Java EE 7) Specification (Oracle) – EDR2, November 2012, migration in process

Java Community Process

List of JSRs not yet migrated to JCP 2.8

- JSR 107, JCACHE - Java Temporary Caching API (G Luck, Oracle) – EDR, JCP 2.7
- JSR 302, Safety Critical Java Technology (Open Group) – EDR, January 2011, JCP 2.7
- JSR 310, Date and Time API (S Colebourne, M Santos, Oracle) – EDR 2 September 2012, JCP 2.7
- JSR 333, Content Repository API for Java Technology 2.1 (Adobe) – EDR, September 2011, JCP 2.7
- JSR 347, Data Grids for the Java Platform (RedHat) – JSR Review, April 2011, JCP 2.7
- JSR 349, Bean Validation 1.1 (RedHat) – PR, October 2012, JCP 2.7
- JSR 350, Java State Management (Oracle) – JSR Review, August 2011, JCP 2.7
- JSR 351, Java Identity API (Oracle) – JSR Review, September 2011, JCP 2.7

Java Community Process

Next Steps

- Contact list of JSR Spec Leads of in-flight Active JSRs that have not yet migrated; encourage them to migrate to JCP 2.8.
- Assist JSR Spec Leads in the process of migrating to JCP 2.8.
- Volunteers (through Adopt-a-JSR program, JUGs, or individuals) to help the PMO do transparency checks on JSRs.

Get community involved!

Java Community Process

Why should we involve community?

- To move the entire Java ecosystem forward.
- To speed up the standards-development process.
- To create better specifications and 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 must enable non-English speakers to participate and to provide their input.

Java Community Process

Adopt a JSR!

Thank you

LJC and SOUJava sociedade de usuários java

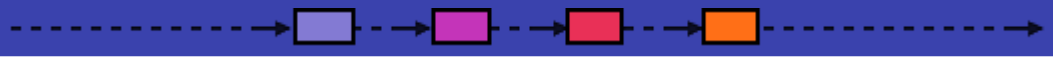
Java Community Process

What is it?

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

Java Community Process

How can Adopt-a-JSR help you?



- Review the specs and provide feedback.
- Help to develop the RI and TCK.
- Respond to messages on mailing lists.
- Triage issues.
- Perform testing.
- Help with documentation.
- File bug reports.
- Complete Transparency check or audit of JSRs.
- Evangelize the project.
- And more...

Java Community Process