Adopt-a-JSR Panel for VJUG
Introduction of Speakers

- Ed Burns
- Arun Gupta
- Heather VanCura
- Martijn Verburg
• Celebrating 15 years!
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 – release early and often!
Adopt a JSR!

Thank you LJC and SOUJava
What is it?

• A JUG-lead initiative to improve Java standards (2012).
• Groups of JUG members work on JSRs.
• See http://java.net/projects/adoptajsr for full details.
JUGS Around the World Adopt JSRs!
Major Java Releases – 2013, 2014

Revolutionary Java 8 release in 2014 – Java SE 8 & Java ME 8 & Embedded Profile!
Java EE 7 release in 2013!
– Significant community contributions to these!
2 Way Street
• Experience with Adopt-a-JSR
Spec lead experience: Ed Burns

• Before Adopt-a-JSR: 2011
• During Adopt-a-JSR: 2012
• Lessons learned: 2013
• Expectations going forward
Before Adopt-a-JSR: 2011

• We did Adopt-a-JSR before there was such a thing
• Use my https://www.java.net/blogs/edburns/ to announce contests and drives
  • Top Five Issues Voting drive

Vote for Your Top Five Issues for JSF 2.2

Posted by edburns on April 14, 2011 at 8:18 AM PDT

Yesterday I facilitated the JSF 2.2 BOF at the CON_FESS 2011 and decided to upload the slides there, at http://www.slideshare give pointers for how to get involved at various levels of interest.

After the meeting we held the kick off meeting for the JSF 2.2

• JSF Logo contest
  • Winner received free book
  • 31 entries!

• https://java.net/jira/browse/JAVASERVERFACES_SPEC_PUBLIC-980
During Adopt-a-JSR: 2012

- Tried to use Adopt-a-JSR for a specific “nice-to-have” technical task
  - reverse engineer style from Java SE 7 Javadoc and apply it to other parts of JSF specification
    - TLDDoc
    - JSDoc
    - RenderKitDoc
  - Proved to be too difficult and ended up not being done
Lessons Learned

• Spec Leads: you get out of it what you put into it
• Tasks must be small and self contained
• Very little context required to do the task
  • Look at Human Intelligence Tasks from Amazon Mechanical Turk: http://www.mturk.com/
• Remember: Adopt-a-JSR is for people who want to contribute but who don’t have time to be a full Expert Group member
Expectations Going Forward

- Kinds of tasks worth considering for Adopt-a-JSR during Java EE 8
  - Given a collection of JIRA issues
    - Ensure they have the right metadata
    - Ensure there is a SSCCE if applicable
  - Define use cases for potential new features
  - Proofread portions of the specification
  - Advocate for “non-supported” platforms
Java EE 8 Community Survey

https://blogs.oracle.com/ldemichiel/entry/results_from_the_java_ee
https://java.net/downloads/javaee-spec/JavaEE8_Community_Survey_Results.pdf
Java EE 8 Possibilities

• Web Standards/HTML5 Alignment
  – HTTP2, SSE, JSON-B, action-oriented web framework

• Cloud
  – Simple security providers, multitenancy, REST management/monitoring

• CDI Alignment
  – CDI 2, EJB services outside EJB, security interceptors, EJB pruning

• Enterprise
  – JCache, Configuration, JMS

• Java SE 8 alignment
Java EE/GlassFish Roadmap

- Expert groups up and running (JavaOne)
- Public Review Draft
- Proposed Final Draft
- Final Drafts Submitted


- GlassFish 4.1 Open Source Edition

- JSR Submissions
- Early Drafts

- 2014
- 2015
- 2016
- 2017
Java EE 8 JSR Status

### Already Here

- Java EE 8 (JSR 366)
- CDI 2 (JSR 365)
- JSON-B (JSR 367)
- JMS 2.1 (JSR 368)
- Servlet 4 (JSR 369)
- JAX-RS 2.1 (JSR 370)
- MVC (JSR 371)
- JSF 2.3  (JSR 372)

### Forthcoming

- Security
- REST Management
- JSON-P 1.1
- WebSocket 1.1
- JTA 1.3
- Batch 1.1
- JPA 2.2
- Concurrency 1.1
Adopt-a-JSR for Java EE 8

• Make Java EE 8 the most community driven platform to-date!
• Share ideas and feedback, possibly by entering issues in public issue trackers.
• Follow expert group discussions and chime in if needed.
• Read early versions of specifications and Javadocs.
• Try writing applications using early builds of the reference implementation.
• Write or speak about the technology.
• Encourage others to participate.
• A number of JUGs have already signed up!

http://glassfish.org/adoptajsr/
Resources: find out more

• Adopt-a-JSR online meeting-materials and recordings available: http://jcp.org/en/resources/multimedia

• Follow the JCP online: http://JCP.org
  • Twitter @jcp_org
  • blogs.oracle.com/jcp
  • Facebook: Java Community Process

• Adopt-a-JSR online: http://adoptajsr.java.net.
  • Twitter @adoptajsr
  • Facebook: AdoptaJSR
  • members@adoptajsr.java.net - the global mailing list
  • IRC: #adoptajsr channel on irc.freenode.net
  • GitHub: https://github.com/Adopt-a-JSR