

# Adopt a JSR



Constantin Pârțac

@quosco

[gplus.to/quosco](https://plus.google.com/u/0/117180540241270412041)

# Acronym soup

- JUG - Java User Group
- JCP - Java Community Process
- JSR - Java Specification Request
- RI - Reference Implementation
- TCK - Technology Compatibility Kit



# What is the Adopt a JSR program?

- A JUG lead initiative to improve standards in the Java ecosystem
- Groups of JUG members work on Java Specification Requests



# Why should I get involved?

- To move the entire Java ecosystem forward
  - To become part of the solution as opposed to staying silent
  - Help define the future
- To ensure that ivory tower standards do not occur
- To make sure the useful libraries and APIs get built
  - Ones that are developer friendly



# Why should I get involved? 2

- To be at the forefront of new technology
- To gain competitive advantage
  - Be the early experts in a new standard
- Boost your career
  - Gain new technical and community skills
- Helps build long lasting activities for the JUG
- **It's fun!**



# How does it help the Standards?

- Standards get earlier feedback
  - Leading to more developer friendly APIs
  - Standards get 'end user/developer' expert input
  
- Standards get developed more quickly, JUGs can help build
  - Reference Implementations (RI)
  - Technical Compatibility Kits (TCK)
  
- JUGs can also help with management of the JSR
  - Managing mailing lists
  - Triaging issues
  - Testing
  - Promote
  - + more!



# What difference have they really made?

- SouJava helped push through JSR-348 and JSR-355
  - Reforming the Java Community Process (JCP)
  - Open mailing lists
  - Public Issue trackers
  - Standards reviewed early and often
  - Smaller EC
  
- The LJC is helping make sure JSR-310 gets through
  - Date and Time for Java 8
  - Building the Technology Compatibility Kit (TCK)
  - Building the RI
  
- Houston JUG is working on Data Grids
  - Vital technology for the cloud space
  - Avoids vendor lock-in



# Who's involved already?

- The London Java Community (UK)
- SouJava (Brazil)
- GoJava (Brazil)
- Houston JUG (USA)
- Chennai JUG (India)
  
- + Several more starting out
  - Nashville JUG (USA)





# Starter Levels to get involved

- Test the early RI builds
  - Use them to find pain points (*“It’s just too laborious to construct X”*)
  - Report bugs (*“Arggh NPE!!”*)
  - Suggest feature enhancements (*“A convertX method would help”*)
- Help triage issues
  - Reproduce issues
  - Erase/merge duplicates
  - Set priorities/categories etc
- Give feedback on design
  - Discuss issues with your JUG and deliver feedback
  - Think about how you would use the JSR as a day to day developer



# Starter Levels to get involved 2

- Help moderate the mailing lists
  - Help the community self police towards helpful conversations
  - Pour water on flame wars etc
  
- Help promote the JSR
  - Social media (Twitter, G+, Facebook)
  - Blogging (write a post about the JSR)
  - Lightning talks (give a talk at Transylvania JUG)
  
- Improve project infrastructure and JSR visibility
  - Help setup canned hosting (java.net, GitHub etc)
  - Help with SEO of website
  - Ensure downloads, mailing lists, issue trackers are easy to find
  - Help maintain their FAQ/Wiki



# Intermediate Levels to get involved

- Help get the JSR to JSR-348 compliance
  - Is there a public issue tracker?
  - Does the EG communicate on a public mailing list?
  - Is the EG balanced?
  - Is the Std a coming together of competing implementations?
  
- Help build the RI
  - Get coding with the actual implementation of the spec!
  
- Help build the TCK
  - All implementations must pass this crucial test suite
  - Great way to gain real TDD/Unit/Integration test experience



# Advanced Levels to get involved

- Join the Expert Group (EG)
  - You need to be an expert in this technology
  - EG members are central to pushing the JSR forwards
  - High time commitment
  - Lots of personal, community and career benefits
  
- Become the Spec Lead for a JSR
  - You need to be a leading expert in a particular technology
  - Considerable time commitment
  - International recognition for your work
  
- Join the Executive Committee
  - High time commitment, influence all standards



# What's next

- Form a group of interested people
- Choose a JSR
- Contact the SL of the JSR
- Hackathon

# How do I join

- Complete the form
  - <http://goo.gl/nc03m>
  
- Or
  
- Contact me
  - @quosco
  - gplus.to/quosco
  - CPARTAC @ .com



# Open JSRs

- ❑ JSR-236 (Concurrency utilities for Java EE)
- ❑ JSR-308 (Type Annotations)
- ❑ JSR-310 (Date and Time)
- ❑ JSR-335 (Lambda Expressions)
- ❑ JSR-339 (JAX-RS 2.0)
- ❑ JSR-341 (EL 3.0)
- ❑ JSR-343 (JMS 2.0)
- ❑ JSR-344 (JSF 2.2)
- ❑ JSR-345 (EJB 3.2)
- ❑ JSR 346 (CDI 1.1)
- ❑ JSR-347 (Data Grids)



# Open JSRs 2

- JSR-349 (Bean Validation 1.1)
- JSR-350 (Session State management)
- JSR-351 (Identity Management)
- JSR-352 (Batch processing)
- JSR-353 (JSON API)
- JSR-354 (Currency)
- JSR-356 (Websockets)
- JSR-358 (JCP.next.3)
- JSR-359 (SIP Servlet 2.0)





# Questions



# Thank you

