

# Contributing to Open Source Projects

# Contributing Agenda

---

- Learning about a Project
- Using a Project
- Joining Project's Community
- Becoming Contributor
- Q/A

# Learning about a Project

- WebSite
  - > <http://www.netbeans.org/community>
- Getting the sources
  - > `hg clone http://hg.netbeans.org/main-golden/`
- Reading documentation
  - > <http://platform.netbeans.org>
- Playing with the project
  - > download NetBeans IDE

# Using a Project

---

- Build sources
  - > ant build
- Running
  - > ant tryme -Ddebug.port=1234
- Patching
  - > enough to build the affected module
  - > cd core.windows; ant clean netbeans
- Use support from some IDE

# Joining Project's Community

- Subscribe to mailing lists
  - > <http://www.netbeans.org/community>
  - > answer questions
- Report bugs and patches
  - > find issue tracking systems
  - > learn about release cycles
  - > insist on bugs being fixed
- Become known in the community
  - > asking questions & showing progress

# Becoming Contributor

---

- Writing Tutorials
- Fixing on enhancing wiki
  - > <http://wiki.netbeans.org>
- Answering on mailing list
- Spreading the Word
  - > blogging
- Engineers Contribute Code

# Engineers Contribute Code

- Building Applications
  - > converting apps to NetBeans
  - > building on top of Platform or IDE
  - > no quality criteria
  - > no release cycles
  - > sandbox at <http://contrib.netbeans.org>
- Publishing via Autoupdate
  - > Plugin Portal: <http://plugins.netbeans.org>
  - > publish from contrib



# Engineers Contribute Code II

- Fixing Bugs
  - > there is always enough bugs
  - > fastest way to learn about a project
  - > fixing the right module
  - > quality criteria
    - > review in issuezilla
    - > hunting module owner to apply the fix
  - > not too many release cycle issues
  - > continuous builds
    - > <http://deadlock.netbeans.org>
  - > join NetFIX team
    - > <http://wiki.netbeans.org/NetFIX>



# Engineers Contribute Code III

- Architecture Changes
  - > open API Review process
    - > <http://openide.netbeans.org/tutorial/reviews/>
  - > fast vs. standard review
  - > need for a test case
  - > proper documentation
    - > <http://openide.netbeans.org/tutorial/api.html>
    - > updating module versions
  - > heavily influenced by schedule

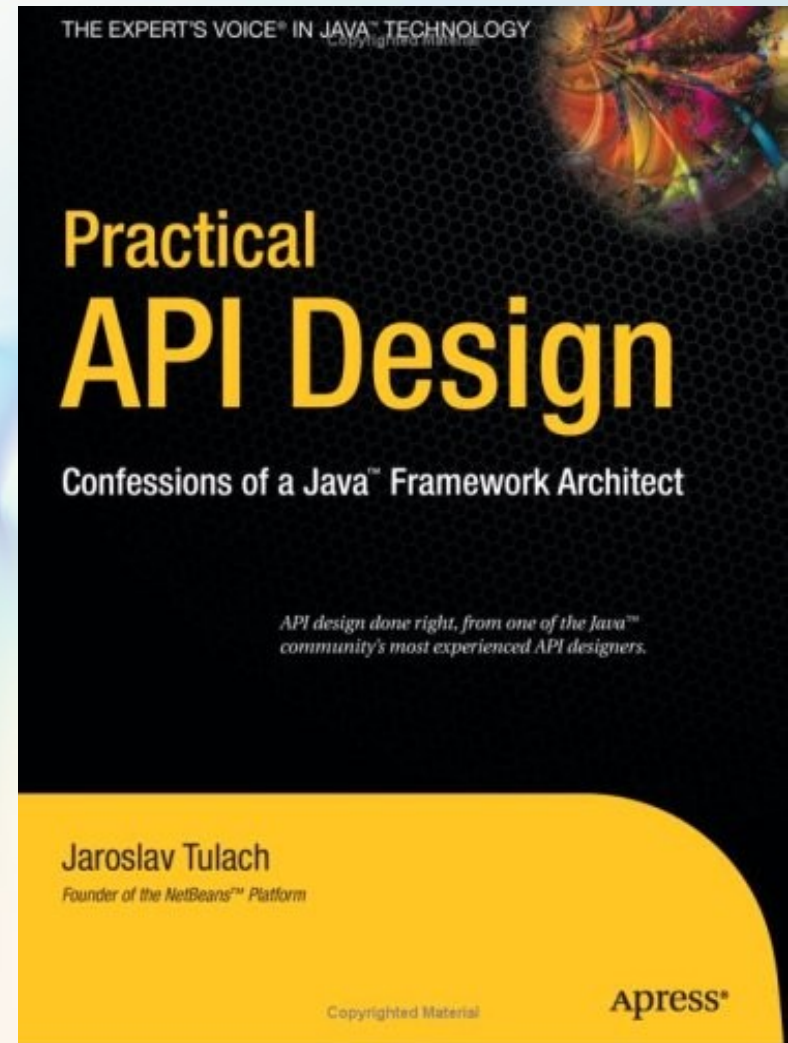
# What is an API?

---

- Public vs. implementation packages
- Extension points
  - > lookup registrations and discovery
  - > Layer registrations and reading
- Properties and Env variables
- Localization messages
- Files being read
- Open sockets

# What is an API?

- <http://apidesign.org>
- theory
- practice
- process



# Conclusion

---

- Know your project
- Engineers write code
  - > on top
  - > patches
  - > architecture changes
- Deal with the community
- Select your Exam Project