



MAY 16-18, 2017

MIAMI, FL

# Navigating the Incubator Trenches

John D. Ament, Sparta Systems

# Who am I?

<http://twitter.com/johnament> | <http://github.com/johnament>

Apache DeltaSpike PMC - it's how I got started!

Active in the Incubator - how I learned The Apache Way

Other ASF projects - ActiveMQ, Usergrid, Juneau, Tamaya, Aria  
Tosca, Pony Mail

Outside the ASF - Arquillian, Hammock

# How to get these slides?

[https://s.apache.org/jda\\_acna\\_2017](https://s.apache.org/jda_acna_2017)

# What is the Incubator?

- Where you (as a community) go to learn The Apache Way
- The main entry point for new projects
- The ASF **does** accept new Straight-to-TLP projects
  - Sometimes..
- This is for everything else

# Why Come to the ASF?

- Open development community
- Move away from vendor driven development
- Strengthen an existing community

# Why Incubate?

- You're unsure of the ASF's policies and need to learn them more
- You're looking to learn The Apache Way
- You need to prove out your community growth
- You're unsure of the actual licensing stance of your project

# How to Incubate

- Find a Champion, someone who will answer your questions on a personal basis
- Draft a Proposal, giving good details about the project and why you think its suitable
  - Your Champion will help you draft the proposal
- Find Mentors, who will help guide the podling into maturing
  - Hopefully your Champion is your first Mentor

# Discuss & Vote

- Your discussion thread will lead to finding more potential contributors and mentors
  - Be selective
- Hold your first vote!
  - Someone from the incoming project should start the vote
  - Sometimes the Champion does it



# Acceptance

- Once the incubator has approved you to join, its time to get started at Apache
  - This is the time when you can start calling yourselves “Apache Foo”
- The Champion’s role is done
  - We hope they’re a mentor still
- Mentors should start helping with Podling Bootstrap

# Migrating Development Over

- E-mail, e-mail, e-mail!
- Use our JIRA, or BugZilla
- GitHub is OK, with prior approval
  - If you're using GitHub, may make sense to use GitHub Issues

# Keeping Your Source Code Clean

- Start with a grant, ideally
- If you're an existing open source project, may not be needed
- Ideally, we want a consistent code base with appropriately licensed code

# Grants & IP Clearance

- Either provide a software grant for the ASF to use the code
- Or go through IP Clearance to vet the code

# Watch What you Import

- Always credit the original author
- Prefer transitive dependencies

# Build a Community

- Open discussions get more people participating
- E-mail avoids “the loudest voice”
- It also helps with asynchronous processing - ensuring that your diverse community can be located anywhere in the world

# Dependencies

- Take a close look at what licenses are approved, and what are banned
- If you don't see what you're using, don't assume, ask!

# Getting a Release Out the Door

- Get in the habit of frequent releases
- Releases should get better each go around, fixing common license and notice content issues



# Release Contents

- Apache produces Source Code Releases
- Convenience binaries can be published in addition to the source code

# Naming Your Release

- The most common is "apache"- "podling"- "version"- "incubating"
- Incubating **isn't** part of the version

# What goes in it?

- All of your source code
- Everything required to build it
- Don't include binaries in the source release

**Stick Around for Justin's talk in a few minutes about Incubator Releases!**



APACHECON<sup>TM</sup>  
North America