InnerSource 101 and The Apache Way

Jim Jagielski
About Me

→ Apache Software Foundation
  → Co-founder, Director, Member and Developer
→ Director
  → Outercurve, MARSEC-XL, OSSI, OSI (ex)...
→ Developer
  → Mega FOSS projects
→ O’Reillyph Open Source Award: 2013
→ European Commission: Luminary Award
→ Sr. Director: Tech Fellows: Capital One
What is “InnerSource”

- InnerSource is the methodology of taking the “lessons learned” from successful FOSS projects and using them internally in Enterprise IT development environments.
  - “running your internal development as if it was an open source project”
- **Principles** and Techniques
Why InnerSource?

- Gain efficiencies by leveraging resident talent to its fullest
- Develop better software faster through collaboration
- Reduce costs through technology standardization and reuse
- Bring products to market faster
- Increase developer engagement and innovation through social creativity
- Attract and retain higher quality talent
What is “The Apache Way”

- “The Apache Way” relates to how the ASF (and its projects) work and operate
- Basically, the least common denominators on how PMCs operate (or are expected to operate)
- Basic governance principles

This work is licensed under a Creative Commons Attribution 3.0 Unported License. - Jim Jagielski - @jimjag
So why focus on Apache?

- The ASF is a 501(c)3, non-profit foundation
- Membership-based
- Large number of success stories (and some “failures”)
- Recognized and acknowledged by peers and press.
Origins of “The Apache Way”

- History of The Apache Group and the ASF
- Focus on community
- Focus on the un-aligned volunteer contributor
Basic Memes

- Meritocracy
- Peer-based
- Consensus decision making
- Collaborative development
- Responsible oversight
- Individual Participation
Meritocracy

➡ “Govern by Merit”
➡ Merit is based on what you do
➡ Merit never expires
➡ Those with more merit, get more responsibility
➡ Provides incentive to Do More
➡ And you CAN Do More (go up the ladder)
➡ It’s NOT a dirty word
➡ aka “do-acracy”
Peer-based

- Developers represent themselves - individuals
- Mutual trust and respect
- All votes hold the same weight (no BDFL)
- Community created code
  - Healthy communities create healthy code
  - Poisonous communities don’t
- Feel as part of a community
Why Community -> Code

- Avoid for new projects, what boot-strapped the Apache Group in the 1st place
- Since we are all volunteers, people’s time and interests change
- A healthy community is “warm and inviting” and encourages a continued influx of developers
- Poisonous people/communities turn people off, and the project will die
- Diversity ensures survivability
- End result - better code, long-term code
Consensus decision making

- Key is the idea of voting
  - +1 - yes
  - +0 - no real comment
  - -1 - veto

- Sometimes you’ll also see stuff like -0, -0.5, etc...

- Used to gauge consensus, not really “majority rule”
Collaborative Development

- Transparent and Public
- Code is developed by the community
- Voting ensures at least 3 active developers
- Development done online and on-list
  - If it didn’t happen on-list, it didn’t happen
Responsible Oversight

- Ensure license compliance
- Track IP
- Quality code
- Quality community
- Community ownership
“If you want to build a ship, don’t drum up the men to gather wood, divide the work, and give orders. Instead, teach them to yearn for the vast and endless sea.”

— Antoine de Saint-Exupéry
Principles of InnerSource

- Culture
- Communication
- Transparency
- Collaboration
- Community
- Meritocracy
Principles: Culture

- Inner-sourcing is a cultural mind-shift
  - Create an expected set of behaviors
  - Must be truly engaged
  - Must be truly accepted

- Techniques:
  - Be the model and guide
  - Embrace the model

- Culture ➔ Communication
Principles: Communication

- Is core and foundational
  - Everything builds on this
- Open and asynchronous
  - Doesn’t disenfranchise anyone
- Archivable
  - Maintains history and allows ebb/flow
  - Document tribal knowledge
- Techniques:
  - Mailing lists; reduce reliance on F2F; invest in infra
- Communication ➔ Transparency
Principles: Transparency

- Reinforces and Enables Public and Open mindset
  - Inclusion
- Reuse
  - You can only reuse what you can see
- Quality/Security
  - More eyeballs mean better quality
- Measurement/Metric
  - You can only measure what you can see

- Transparency ➔ Collaboration
Principles: Collaboration

- Common Vision
- Common Goal
- See connections
- Consensus:
  - Encourages contribution and improves leverage
  - Encourages feedback and dialogue
- Techniques:
  - Build and Test Tools
  - “Newbie” issues
- Collaboration ➔ Community

This work is licensed under a Creative Commons Attribution 3.0 Unported License. - Jim Jagielski - @jimjag
Principles: Community

- Loyalty
  - Community breeds loyalty

- Durability
  - Communities can create durable assets, processes and culture

- Health

- Feedback and Dialogue

- Not-just the “regular, expected players”

- Community ➔ Meritocracy
Principles: Meritocracy

- Technical decisions made by technical experts
  - Better informed decisions
- Role models
  - Merit provides examples
- Earned authority
  - “Natural” leadership
- Known path and “rewards”
- Influence and Direction are possible (! (Us vs Them))
- Meritocracy ⇒ Communication
Final Thoughts

- Community is not the same as team
  - self organizing
  - self identifying
- Contribution is work
- Community requires investment
- Transparency is not a threat
- Collaboration means compromise
- Driving results means driving consensus
Thanks

Twitter: @jimjag

Emails:
  jim@jaguNET.com
  jim@apache.org
  jim.jagielski@capitalone.com

http://www.slideshare.net/jimjag/