



THE
APACHE™
SOFTWARE FOUNDATION

Apache Explained

Presented by
Craig Russell
Justin Mclean
Willem Jiang

Including original work of Bertrand Delacretaz, Roman Shaposhnik and other amazing ASF contributors

Who is Craig Russell?

- Software Architect
 - Object Data Management Group
 - Java Data Objects (JDO)
 - Database Jones (MySQL)
- Apache Committer - from 2005
- Apache Member - from 2007
- Apache Secretary - 2010 - 2019
- Member, Incubator Project Management Committee
- Chairman, Apache Board of Directors

Apache Software Foundation: The World's Largest Open Source Foundation

The ASF's Mission

The Apache Software Foundation (ASF) is a US 501(c)(3) charitable organization. Its mission is to provide Open Source software for the public good. We do this by providing services and support for many like-minded software project communities of individuals who choose to join the ASF.

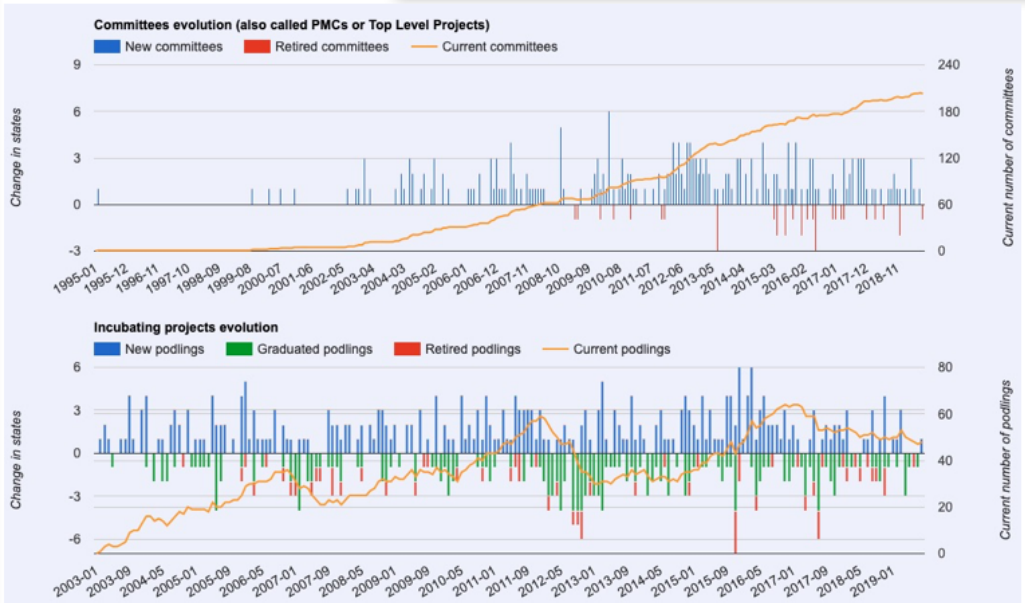
A **neutral space** where projects which are **independent** from any corporate influence can prosper and create **Open Source software** for the **public good**. Under the business-friendly Apache License 2.0

**Established as a public charity in
1999**

**Mission: To provide software at no
charge for the public good.**



7036 Committers
 765 Members
 203 PMCs
 10 nonPMCs
 48 Podlings



First Project

APACHE
HTTP SERVER



World's Most Popular Server

2019 24th Anniversary

(20 years at the ASF)

Apache Project Domains

- Artificial Intelligence
- Internet of Things
- Big Data
- Financial Management
- Climate Change

Not Apache Project Domains [Yet]

- Robot Apocalypse
- Autonomous Vehicles
- Interplanetary Exploration
- ~~Blockchain~~
- Your Project Here

ASF provides project governance

The Apache Way

“The Apache Way” core principles

- Rough consensus and working code
- Small reversible steps
- Any constructive contribution DOES earn merit... while corporate affiliation and seniority DOESN'T
- Merit doesn't expire
- Merit doesn't give you authority; it gives you access
- No BDFLs, no vetoes - relying on building consensus instead

“The Apache Way”
AKA
“Community over code”

Why the ASF Works: The Apache Way

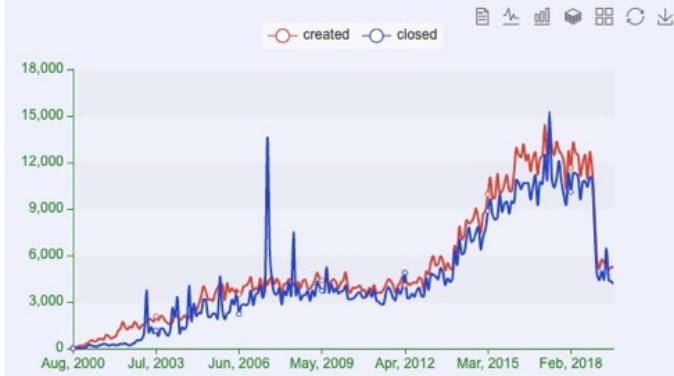
- **Meritocracy** — Individuals earn merit by what they do and how.
- **Transparency** — Our projects are developed in the open, so everyone who wants to participate is able to.
- **Community** — We place community health over all else

And yet it is NOT the right fit for every
project

The Apache Way Ensures:

- Vendor Neutrality
- Diversity
- Trust
- Safety

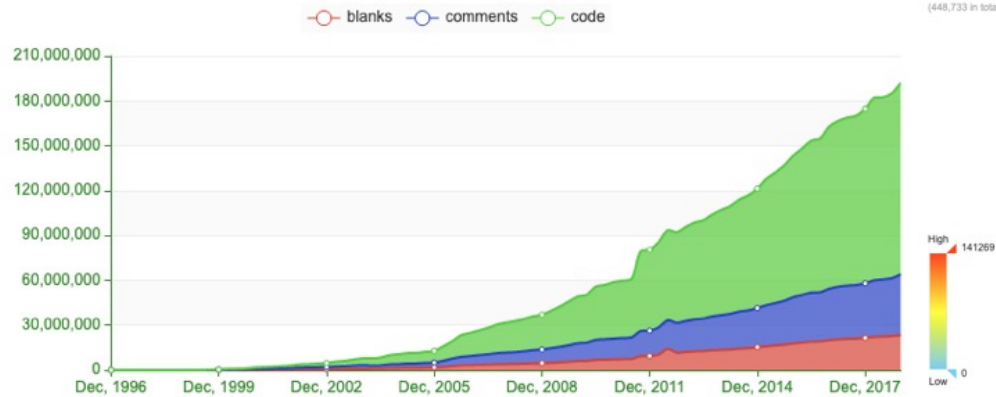
Number of issues opened/closed, all time:



Email, topics and email authors, past year:

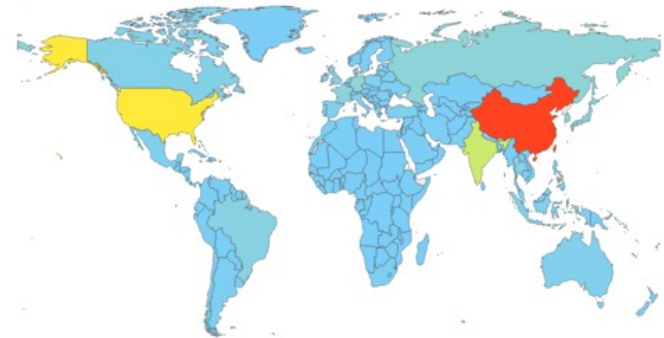


Evolution of codebase over time:



Worldwide download mirror activity, past 30 days

Worldwide distribution by country
(448,733 in total from 203 countries)



PLATINUM SPONSORS:



TARGETED PLATINUM SPONSORS:



GOLD SPONSORS:



TARGETED GOLD SPONSORS:



SILVER SPONSORS:



TARGETED SILVER SPONSORS:



And also a number of Bronze sponsors, full up-to-date list at

<http://www.apache.org/foundation/thanks.html>

Why existing projects come to Apache

- **Corporate Citizenship** — Give to charity
- **Higher Quality** — More developers with different perspectives
- **Collaboration vs Competition** — It's easier to work with competitors on open source projects that benefit everyone
- **“Free” Training** — The community can train your new developers easier than you can
- **Legal Protection** — The ASF protects contributors

The Apache Incubator

- The official entry path for projects to enter The Apache Software Foundation.
- Mentorship helps train communities in "The Apache Way".



Podlings Undergoing Development



Who is Justin Mclean?

- Freelance developer for 25+ years
 - IoT Meetup Sydney
- Apache Committer - from 2012
- Apache Member - from 2014
- Member, Incubator Project Management Committee
- VP Apache Incubator, VP Apache Mynewt
- Mentor for IoTDB, Dubbo, RocketMQ and Apex

The Apache Incubator

- Learn The Apache Way
- Growing the community
- Establish Licensing
- Release



The Apache Incubator

- Takes 1-2 years sometimes longer
- Graduate to a Top Level Project
- Not all projects graduate



Community Building

- **Politeness** — Be nice
- **Respect** — Everyone's point of view has value
- **Trust** — Assume the best intentions
- **Humility** — Others might have better ideas

Code of conduct: <https://www.apache.org/foundation/policies/conduct.html>

Licensing is a key aspect

- **Provenance** — Establish where every line of code originated
- **Trademarks** — Must not conflict with existing names
- **Software Grants** — The owner of the intellectual property of existing code/test/documentation must grant rights to Apache
- **Contributor Licenses** — Contributions while at Apache are governed by contributor licenses:
 - Copyright, Patent, Distribution/Modification

Apache License

- Universal donor
- Pragmatic
- Free for commercial and non-commercial use
- Compatible with other Open Source licenses

Licensing

- **Category A** - can depend on and include in release
 - MIT, 2 and 3 clause BSD
- **Category B** - can depend and may be able to include
 - Common Development and Distribution License (CDDL), Eclipse Public License (EPL), Mozilla Public License (MPL), Creative Common Attribution (CC-A)
- **Category X** - can't depend on or include in release
 - GPL, LGPL, non commercial licenses, JSON

Release distributions

- Legally an act of the Foundation
 - “Does NOT need to work”
 - “Does need proper licensing”
- Authorized by the Project Management Committee
 - Vote required
 - Digital Signature(s) required
- Distributed via extensive mirror system

Releases

- Are signed
- Need an incubating disclaimer
- Have LICENSE and NOTICE files
- Follows license terms of ALv2 and any included 3rd party software
- Source files have ASF headers
- Consist of source with no compiled code

Binary releases

- Are not required
- Are not official releases
- Also need to follow policy

Graduation

- Demonstrate that the project can operate independently:
 - Self-governance
 - Release management
- Establish legal framework
- Publicity



ServiceComb Incubating Journey



2017-5



2017-11



2018-10

Who is Willem Jiang ?

- Huawei OpenSource Center Technical Expert
- Former RedHat Principal Engineer
- Apache Member
- VP Apache ServiceComb

ServiceComb Introduction

- **Java Chassis** - High performance Java service engine with MicroServices governance
- **Service Center** - High performance Service Center based on Etcd
- **Pack** - A Central distributed transaction coordinator which supports Saga and TCC
- **Mesher** - A services mesh implementation based on Go
- **Kie** - A new designed MicroServices configuration center

Find a way to Apache

- Writing a proposal of your project
- Finding a Champion and Mentors
- Proposal discussion in IPMC
- Starting a vote of your proposal
- Creating a podling project inside of Apache Incubator
- SGA, CLA, iCLA Signing
- Podling name searching

Integrating with Apache Infrastructure

- **Mail lists** — “If it isn’t on an archived list, it didn’t happen”.
- **Issues** - Using JIRA or github issue to track the development related issues.
- **Source repository** — Viewing all parts of the system is open to the public.
- **Build/Test** — Most projects now incorporate continuous integration build and test processes.
- **Establish a Web Presence** — Each project has its own unique web page <https://project.apache.org>

Community

- **Users** — Users are the reason a project exists. No users ⇒ no project
- **Contributors** — Our projects depend on contributions from the community — bug reports, email discussions, bug fixes, documentation
- **Committers** — Frequent contributors become committers after demonstrating merit
- **Project Management Committee Members** — The PMC is responsible for setting the direction of the project, authorizing releases, and voting new committers and PMC members

Growing the community

- Raising The Profile
 - Improving the website, using grassroots media
 - Submitting talks to conference, writing articles
 - Working with downstream projects
- Building a community by stepping back a little
- Helping developers become committers

Voting Rules of Release

- Vote in the open on mailing list, open for 72 hours
- Release needs to comply with ASF policy
- Needs 3 +1 votes and more +1's than -1's
 - A -1 vote (a veto) doesn't block the release
 - -1 needs to have a reason to be valid
 - Can change your vote
- If vote passes then the IPMC votes on the release
- If vote fails then create new release candidate and vote again

Why you might vote -1 in IPMC

- Unexpected compiled code in release
- Includes software with incompatible license
- Issue with LICENSE and NOTICE files
- Issue with copyright
- Missing ASF headers
- Contains encryption software

Practice makes perfect

- Not expected to get it right first off
- Licensing is complex
- May not be familiar with policy
- Policy is more guidelines, there are few absolute rules
- DISCLAIMER-WIP
- Release early and release often
- Each release better than the last

Graduation from Apache Incubator

- Complete (and sign off) tasks documented in the status file
- Ensure suitable project name and product names
- Demonstrate ability to creating an Apache Release
- Creating an Open and Diverse community
- Check with apache project maturity model

<http://community.apache.org/apache-way/apache-project-maturity-model.html>

Support The ASF

- The ASF is funded through **tax-deductible contributions** from corporations, foundations, and private individuals
- **The ASF spends 10% or less on overhead**
- ASF Infrastructure keeps Apache critical support services running 24x7x365 for **less than US\$5K per project/year**
- Help keep Apache software projects **freely available to ALL** around the world