

Topic: Transmission of Values and Culture in Open Source

Research Question:

To what extent can open source values and culture be effectively transmitted to new projects ?

Supervisor: Ing. Aleš Kubíček, Ph.D.

Supervisor: Shane Curcuru

Student: Sharan Foga

Date: 10<sup>th</sup> January 2019

# Table of Contents

Introduction.....	3
History and Background.....	3
Culture and Apache.....	4
Meritocracy.....	4
Theory.....	5
Methodology.....	5
Culture.....	6
Assessing “The Apache Way”.....	6
Data Source.....	9
Research Tools.....	9
Apache Kibble.....	9
Pony Factor.....	9
Augmented Pony Factor.....	10
Meta Pony Factor.....	10
Sentient Analysis.....	11
Key Phrase Extraction.....	12
Contributor Retention.....	12
Practical.....	13
Apache HTTP Server Project Cultural Baseline.....	14
Baseline Comparison with TLPs Over 5 Years.....	19
Baseline Comparison with TLP <= 5 Years.....	24
Conclusions.....	29
References.....	31

# Introduction

This paper will focus on culture and how it has evolved at the Apache Software Foundation (ASF). It will review existing literature and research to find the main cultural elements that comprise ASF culture and using tools and indicators, show whether these cultural elements can be successfully transmitted.

The ASF was founded in 1999 on a single open source project, called the Apache HTTP Server project. The values, behaviour, knowledge and governance model that developed as part of the creation of the Apache HTTP Server project is the source of the ASF culture that is called “The Apache Way”.

The Apache Way is dependent on one central tenet – meritocracy, and this is embedded into all layers of the ASF from the formal governance model and the election of directors and members all the way through to the projects and the recognition of individual contributions.

Today the ASF is made up of over 350 projects and software initiatives and it is claimed that each of these projects demonstrate and accept the Apache Way as their cultural model. This paper will attempt to define and create a cultural baseline for the ASF, and then test the baseline against other ASF projects to see if they have the same or similar profile.

To understand the ASF culture, we need to look closely at the origins of the first ever Apache project - Apache HTTP Server.

## History and Background

During the mid 1990s the most popular web server software in the world was completely free. It was called the HTTP daemon and had been developed by Rob McCool at the National Center for Supercomputing Applications (NCSA), University of Illinois. When Rob left the NCSA, the development and maintenance of the HTTP daemon effectively stopped even though many people were still actively using it.

Rather than abandoning the project, a small group of users decided to work together on fixes and extensions for the HTTP daemon software. They did this remotely and set-up tools to help them collaborate. These included:

- a shared mailing list
- an information space for documentation and;
- a central repository for code changes.

This group became known as the “Apache Group”. Less than a year later as a result of their collaboration, they released their own version of the HTTP daemon and called it the Apache HTTP Server. Apache HTTP Server became the most popular web server globally, even surpassing the popularity of the original NCSA version.

In 1999, members of the Apache Group established the Apache Software Foundation as a US 501(c)(3) membership-based not for profit foundation to provide an organisational, legal, and financial framework for developing Apache HTTP Server as well as other open source projects.

An important part of the framework was a cultural model that was a scaled up version of how the Apache Group had collaborated, communicated and interacted together. It was based on acknowledging individual contributions. Anyone showing their commitment by actively contributing could be recognised, trusted and given additional project responsibilities.

This is still a core part of the ASF today, and contributors that actively participate, contribute and collaborate in any of the ASF's software initiatives are eligible to become committers, members and in turn potential foundation officers or members of the Board of Directors.

Committers are ASF contributors that have been recognised by individual projects. Members are ASF contributors that have been nominated and approved by a majority of the existing members. Directors are ASF contributors who are members that have been nominated and approved by the membership. This means that the ASF is governed by the contributors that work and collaborate within its projects.

At nearly 20 years old, the Apache Software Foundation is funded by donations. It is predominantly run by volunteers with only 5 full time equivalent (FTE) staff paid roles. The Board of Directors is also made up of volunteers that are elected annually.

At the time of writing the ASF has 731 individual members and over 6000 committers and its central mission is:

***“to provide open source software for the public good”***

## **Culture and Apache**

Many open source projects are created as non profit entities. Examples include:

- The Document Foundation
- Mozilla Foundation
- Free Software Foundation
- Software Freedom Conservancy

This means that their mission and purpose is related to delivering value or benefits to the public, rather than making money. To recruit new contributors to help achieve their mission, open source projects need to provide something specific to attract contributors and encourage them to participate. Culture can be used as a way of attracting and retaining contributors.

The ASF has its own unique culture called “the Apache Way”. It is a culture that was created directly as a result of establishing the Apache HTTP Server project and it has evolved over time.

When the ASF was established in 1999, a key goal was to ensure that all ASF projects would continue to exist beyond the participation of individual volunteers. This is directly related to the NCSA HTTP daemon where the original developer left the project and the Apache Group worked together to revive, maintain and successfully evolve it.

## **Meritocracy**

The open source sector is one that has evolved over the last 20 years and during that time several

governance models have emerged. Examples include include:

- Autocratic
- Community Source
- Meritocratic

The ASF is based on the Meritocratic model where the contributions an individual makes are recognised. Each contribution earns “merit” and as an individual accumulates merit, they are rewarded by being allowed to take on more responsibility.

Meritocracy has also been referred to as a “do-ocracy” because those that do more have the ability to achieve more responsibility. This means that the power to influence a project lies with the people that are actively contributing.

This is the governance model that has been adopted by the ASF. The Apache Way has been described as a meritocratic process and is clearly documented in the ASF operating policies. The ASF guidelines state that:

***“individuals comprise the ASF”***

and promote a key strength as being that

***“affiliations do not cloud the personal contributions.”***

This means that ASF contributors are aware of the differing commercial and individual perspectives and that that merit accrues to the individual irrespective of their corporate affiliations.

## **Theory**

### **Methodology**

This paper will analyse the main elements of the ASF culture (“the Apache Way”) and attempt to provide evidence to show the extent that this culture can be successfully transmitted to new projects. It will test and explore the following hypothesis:

- As the original source of ASF culture, data from the Apache HTTP Server project can be extracted and used to create a cultural model
- This cultural model can be used as a baseline to compare new ASF projects to see if they exhibit a similar cultural profile or not

Based on the previous assumption, this paper will focus on the following:

- Define the main concepts and values of the ASF cultural model using publicly available information
- Mine the public data available from the ASF code repositories and mailing lists for the Apache HTTP Server project to create a cultural baseline

- Mine the public data available from the ASF code repositories and mailing lists for a series of Apache projects that were created after the Apache HTTP Server project to look for data indicators that validate or disprove the demonstration of elements found in the Apache HTTP cultural baseline

## Culture

Allen (2003) highlighted that there are over ninety different definitions of culture. For the purposes of this paper we will use the definition of culture as follows:

***“Culture ... refers ... to learned, accumulated experience. A culture ... refers to those socially transmitted patterns for behaviour characteristic of a particular social group”***  
(Keesing [22, p. 68]).

This means that culture is the source of the values and behaviours that "contribute to the unique social and psychological" environment. Culture can be positive, negative or a combination of the two. More importantly it can be something that makes a group or organisation distinct enough to have other people aspire to become a part of it.

Culture is central to individual motivation. Ryan & Deci (2000) defined intrinsic motivation as

***“doing an activity for its inherent satisfaction”***

linked to enjoyment or creativity, and extrinsic motivation as:

***“doing an activity to attain some separable outcome”***

linked to achievement or professional goals.

Research by Lakhani & Wolf (2001) confirm that both intrinsic and extrinsic motivations are present in the open source sector but the single most important motivation factor was intrinsic.

In 2002 Yochai Benkler defined the term ‘peer production’ as a way to describe the way that software is developed as a result of the combined efforts of individual volunteers. The main benefit was that the resulting product would continue to grow and evolve over time.

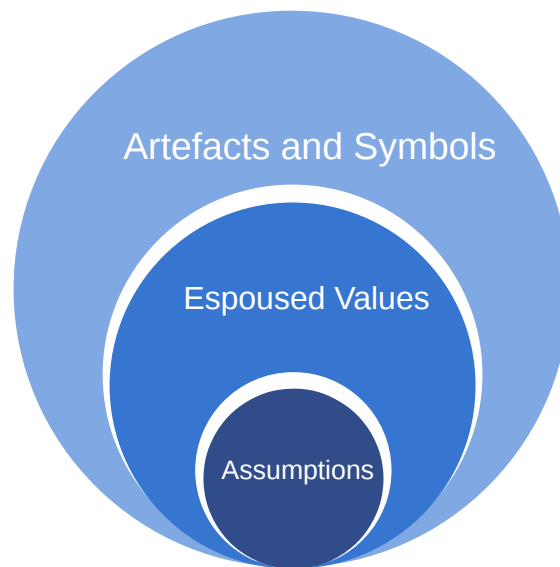
This is very relevant to the ASF because this research supports the ASF philosophy of creating a community of volunteers, working together because they want to contribute to the longevity of a project.

## Assessing “The Apache Way”

Schein’s cultural model is often used to identify and define aspects of a culture. It is made up of 3 distinct layers:

1. Artefacts and Symbols
2. Espoused Values
3. Assumptions

Schein's model is sometimes referred to as the “onion” model because to analyse and view the inner layers, you first need to understand or experience the outer ones. Schein’s model will be used for assessing the ASF culture, the Apache Way.



**Fig 1: Schein's Organisational Culture Model (“Onion Model”)**

### **Artefacts and Symbols**

All organisations use some form of branding that is strongly associated with its identity. It is mainly external facing and is the first impression that outsiders get of the organisation. This is not only limited to logos, but also includes specific processes or corporate uniforms that define a visible and recognisable identity.

All open source projects maintain some form of visual identity ranging from logos and catch phrases to individual people. Some of the ASF artefacts and symbols are as follows:

- The “Apache” name
  - All Apache projects include this name as part of their project name.
  - All Apache projects are at the \*.apache.org domain
  - Apache as a brand is actively protected and trademarks for projects have been registered.
- The Feather Logo
  - The Apache feather logo is also a strong part of its identity.
  - Most of the Apache project logos include a representation of the Apache feather.
- People
  - Individuals from the original Apache Group are still active within the ASF.

### **Espoused Values**

These are the core values of an organisation that have been integrated into the governance structure and processes. By-laws and codes of conduct document the expected behaviours and value systems for the project. The ASF publicises its core values and highlights the following six core principles are central to “The Apache Way”:

- collaborative software development
- commercial-friendly standard license
- consistently high quality software
- respectful, honest, technical-based interaction
- faithful implementation of standards
- security as a mandatory feature

These values are not unique, many organisations can claim to have similar principles yet the resulting culture may be completely different. It is the way these values are implemented and the resulting behaviour that also define a culture.

To understand how some of these values are transmitted, the following are examples of some of the most common terms and phrases frequently used to express ASF culture both internally and externally:

- Merit
  - People contribute to the ASF as individuals.
  - People who are active are recognised and rewarded for their contributions.
- Openness and Collaboration
  - As well as providing open source software, interactions must be open and transparent.
  - All ASF projects have publicly archived mailing list where project discussions are held.
- Asynchronous Communication
  - Individuals are located globally so allow time (e.g. 72 hours) for people to respond.
  - Not everyone speaks the same language and may need time to translate and formulate a response.
- Community
  - All ASF projects are built around communities.
  - Projects come to the ASF because they want to successfully grow their communities.

### **Assumptions**

Assumptions can be described as unconscious behaviours and values that are deeply embedded in an organisation. These behaviours occur so naturally to the people who are part of the organisational culture that they may not even be aware of them. In many cases only people outside that are not already connected with the culture can identify or recognise when these assumptions are being displayed.

All open source projects are different and it will be these inherent assumptions that will form the core of each individual project culture.

The following are some of the observed potentially inherent assumptions:

- Consensus
  - Decisions are made by consensus.
  - Consensus is not voting, it is showing an opinion or support.



- Anyone that is part of an ASF community has the power to help decide.
- Getting consensus from a community is better than voting.
- Community Self Correction
  - Each project community is independent.
  - Allow communities to self correct if something negative occurs.
  - Intervention from the Board of Directors is a last resort solution.
- Community over Code
  - It is more important to have a collaborative community than it is to have to the best software and a dysfunctional community.
  - Respectful interactions and collaboration can still occur even if both sides have opposite views on technical or project related issues.
- Different Hats
  - The understanding different perspectives.
    - Commercial v Employee perspective
    - Community vs Individual perspective
- Mailing List as the Source of Truth
  - If it's not on the mailing list, it didn't happen!

## Data Source

All ASF projects have publicly archived mailing lists. The Apache mailing list are a permanent searchable archive that are publicly available. They are a legacy communication medium from the creation of the ASF that now forms an integral part of any ASF project. It is the heart of a project and a place where people interact, communicate, collaborate, argue, agree and disagree. This means that it is an appropriate place to mine data to look for cultural indicators.

## Research Tools

The research will be carried out using the following tools, formulas and indicators.

### Apache Kibble

Apache Kibble is a suite of tools for collecting, aggregating and visualising data and activity in software projects.

The following Kibble indicators will be used:

### Pony Factor

The Pony Factor (PF) measures the diversity of a project based on the contributions from individual contributors. It can be defined as:

- *“The lowest number of contributors whose total contribution makes up the majority”*

of whatever is being measured (e.g. lines of code written, number of messages sent etc.).

Mathematically it is written as follows:

$$\sum_{n=1}^P C_n \geq K \cdot V$$

Where:

- P is the Pony Factor
- $C_n$  is the number of contributions made by contributor n sorted by descending number of contributions
- K is the percentage of the total contributions we are looking for
- V is the total volume of contributions made

A higher Pony Factor means that a project has a good tolerance for continuing to survive if one or more of the core contributors leaves.

**NOTE:** Pony Factor includes all contributions from contributors whether they are still active or not.

## Augmented Pony Factor

The Augmented Pony Factor (APF) is an adjustment to the standard Pony Factor calculation where contributions from contributors that are no longer active are not included.

$$\sum_{n=1}^P C_n \geq K \cdot V \quad - \text{Contributions from non active contributors}$$

**NOTE:** The Augmented Pony Factor will not be used as part of the assessment but a description of it has been included here for completeness.

## Meta Pony Factor

The Meta Pony Factor calculation is a work in progress. It attempts measure the affiliation of a contributor based on the email address linked to the contribution. If developed further then this could help identify distinct organisations that are contributing.

The following example can help explain the use of Pony Factor, Augmented Pony Factor and Meta Pony Factor.

### **Example:**

An open source project has 35 active contributors:

- Contributor A has made 19% of the contributions, works for Company X
- Contributor B has made 15% of the contributions, works for Company X
- Contributor C has made 22% of the contributions but is no longer active, works for Company Y
- Contributor D has made 12% of the contributions, works for Company Z
- Contributor E has made 9% of the contributions has no company affiliation

- All other contributions were made the other 30 other contributors

**Pony Factor:** The Pony Factor would be 3 because contributions from Contributors A, B and C make up over 50% (I.e. 56% in this case) of the project contributions.

**Augmented Pony Factor:** The Augmented Pony Factor would be 4 because Contributor C is no longer active so the 22% that they contributed would not be included as part of the calculation. This means that contributions from Contributors A, B,D and E make up 55% (the majority) of the project.

**Meta Pony Factor:** The Meta Pony Factor is 4 because we have 4 visible affiliations (Company X,Y, Z and no affiliation) associated with the Pony Factor contributions.

### **Relevance for the ASF**

ASF projects are made up of individual contributors. Some of these individuals contribute to the software through code and others via knowledge (e.g. documentation, testing, marketing etc.). The Pony Factors can measure the diversity of a project in terms of the division of work among the contributors. The higher the Pony Factor the stronger the tolerance for continuing to survive if one or more of the core contributors leaves the project.

The Apache HTTP Server project was created directly as a result of a core contributor leaving the NCSA project so a key part of ASF culture is ensuring that projects can survive even when main contributors leave.

The Pony Factors can also help indicate:

- the growth of a project
- new contributors are being accepted
- merit is being rewarded by committer status

### **Sentient Analysis**

Pang and Lee (2008) defined “Sentient analysis” is a tool that is

***“a kind of text mining, which is used to predict human mind, specifically the emotional state of a person by extracting specific emotional expressions from the text”***

This means that it can be used as an indicator to gauge people’s opinions and reactions to certain ideas. Data is collected in the form of text and an algorithm is used to identify keywords associated with an emotion. Any communication can be linked to several emotions so weightings are used to highlight the strength of the sentiment.

All ASF projects have public mailing lists as the main method of communication. Everything that occurs in a community will be reflected in the text interactions being recorded on the mailing lists. This means that running sentient analysis over the mailing lists can give an indication of the emotional state of a community.

The following mood sets will be used for the analysis:

- `moods_good = set(['trust', 'joy', 'confident', 'positive'])`
- `moods_bad = set(['sadness', 'anger', 'disgust', 'fear', 'negative'])`

- `moods_neutral = set(['anticipation', 'surprise', 'tentative', 'analytical', 'neutral'])`

## Key Phrase Extraction

Key Phrase Extraction (KPE) is a method where key phrase or words are extracted that can summarise the main ideas or themes of a document. It has been successfully used for indexing journals and online content but in this paper it will be used to extract any text that could indicate cultural ideas or language.

If the culture is embedded in a community then the language used should be indicative of the culture. The expectation is that words and phrase that have a strong significance with ASF culture would be present.

## Contributor Retention

Contributor retention is related to how successful a project is at attracting and retaining contributor and can be broken down into the following areas:

- Active Contributors:
  - How many contributors are active within the project
  - Are contributors regular and remain active over a longer timespan
- Retained Contributors:
  - How long a contributor has been contributing
  - The longer a contributor has been retained the more successful a project is at retaining them
- Contributors that have Left
  - How many contributors are leaving
- Past Contributors that have Returned
  - How many contributors have contributed in the past and have returned to rejoin the community

Contributor retention is a very important metric because as well as activity and non activity, it also shows if a project has a good mix of active contributors, is accepting new contributors or is mainly dependent on experienced ones.

# Practical

## Data Selection

Data to create the cultural baseline will be extracted from the Apache HTTP Server project and the following data will be used:

- Apache HTTP Server project Mailing List Archives 1996 – Current
- Apache HTTP Server project Code Repositories 1996 – Current

The following 15 projects have been selected for comparison because:

- They have different varying years of being an ASF project ranging from 1 to 11 years
- They have data available in Apache Kibble
- None of them existed at the time that HTTP Server project was created
- The culture they exhibit would have been created after ASF was established

## List of 15 Apache Top Level Projects (TLPs) for Cultural Comparison

<b>Project Name</b>	<b>Years as an Apache Top Level Project (TLP)</b>
Apache Beam	2
Apache Clerezza	5
Apache Cloudstack	5
Apache Fineract	1
Apache Jena	6
Apache Ignite	3
Apache Kudu	1
Apache Netbeans	2
Apache OFBiz	11
Apache Phoenix	4
Apache Pivot	8
Apache Sling	8
Apache Stanbol	6
Apache Subversion	8
Apache Traffic Server	8

***NOTE: Years have been rounded down to previous full year as at 1<sup>st</sup> November 2018.***

The projects can be broken down into two sub groups as follows:

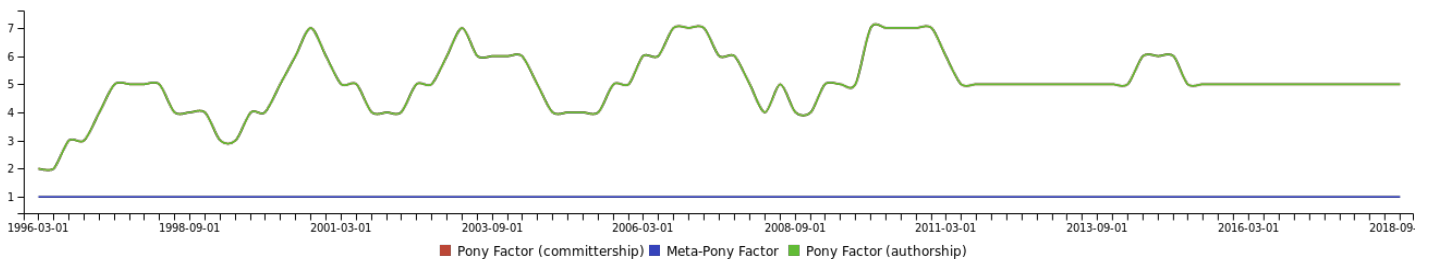
Projects 5 Years and Less as TLP	Projects Over 5 Years as TLP
Apache Beam	Apache Jena
Apache Fineract	Apache OFBiz
Apache Ignite	Apache Pivot
Apache Kudu	Apache Sling
Apache Netbeans	Apache Stanbol
Apache Phoenix	Apache Subversion
Apache Clerezza	Apache Traffic Server
Apache Cloudstack	

- Source data from project mailing lists, source repositories and issue trackers for these projects was be loaded into two Kibble views
- Data for each Apache Kibble indicator was reviewed against the Apache HTTP Server project baseline.

## Apache HTTP Server Project Cultural Baseline

### Indicator 1: Baseline Pony Factor Codebase

The following graph shows the Pony Factor for the Apache HTTP Server project code base from 1996 until 2018.



**Fig 2: Apache HTTP Server project Baseline Pony Factor Codebase 1996-2018**

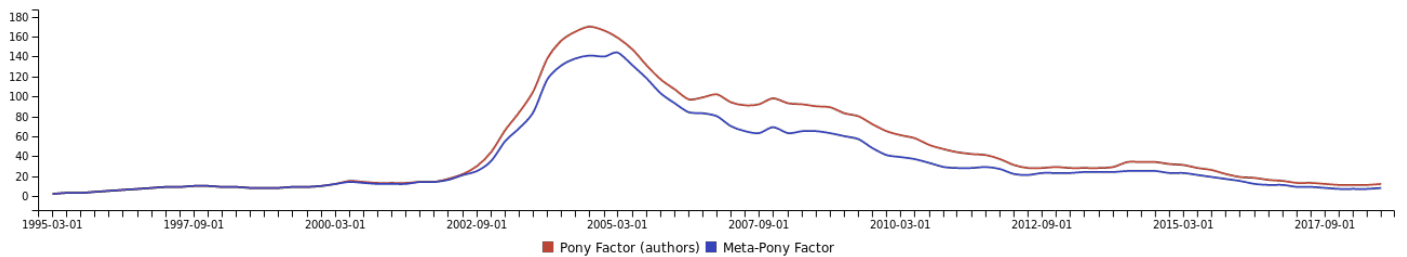
At the beginning of the project, in 1996 the Pony Factor is low (2). This is at the time of the creation of the Apache Group so it would be expected that the number of core maintainers would be low. From 1996 onwards there is an increase in the Pony Factor, highlighting the growth in the number of people contributing to the project. Over time the Factor rises and falls and the 2018 figure is still high and remains constant.

The Pony Factor for committership not distinctly visible follows the same profile as authorship.

The Meta Pony Factor remains constant at 1 because all interactions are associated with the ASF affiliation.

### **Indicator 2: Baseline Pony Factor – Email**

The following graph shows the Pony Factor for the Apache HTTP Server project mailing list from 1995 until 2018.



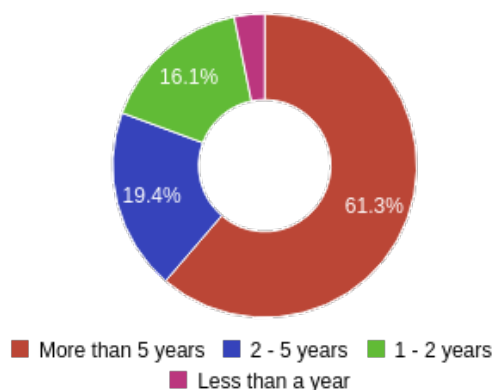
**Fig 3: Apache HTTP Server Project Baseline Email Pony Factor 1995-2018**

As with the Apache HTTP Server project codebase, the Pony Factor for the Apache HTTP Server project mailing lists in 1995 is low (2) highlighting that there were a limited number of people active on the mailing lists. The number of active mailing list contributors gradually rises to a peak of up to 170 during 2005. It is now reducing and has stabilised at 12.

The Meta Pony factor increases because email contributions are more flexible and affiliations can be easily captured. It follows the same curve as the Pony Factor and has now reduced and stabilised at 11.

### **Indicator 3: Baseline Contributor Experience**

The following graph shows a breakdown the length of time contributors have been contributing to the codebase:

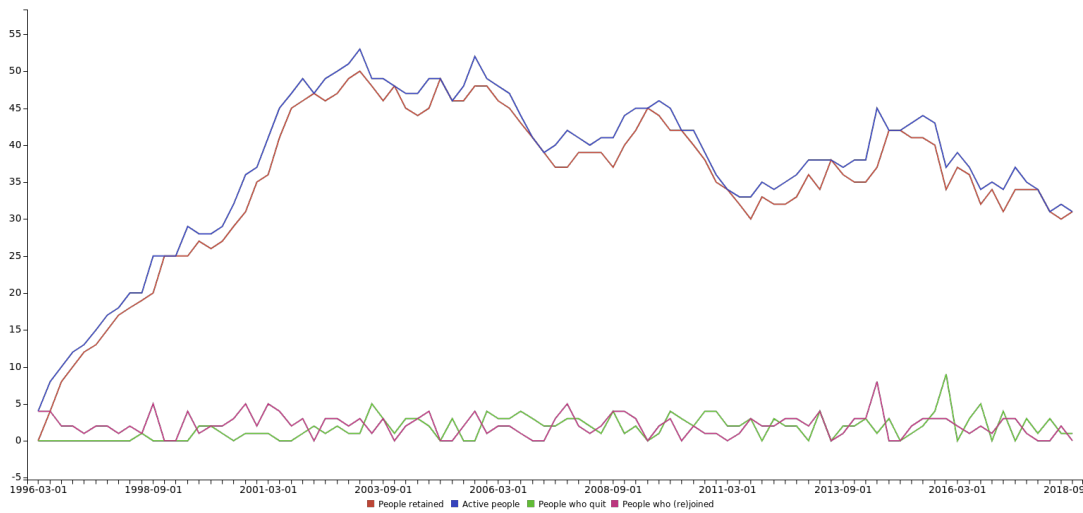


- The Apache HTTP Server project is over 20 years old and the oldest Apache project
- Over 61% of their contributors have been contributing to the project for more than 5 years
- Only 3 % of their contributors have less than a year's experience.

**Fig 4: Apache HTTP Server Project Baseline Contributor Experience**

### **Indicator 4: Baseline Contributor Retention Codebase**

The following graph shows how many people have been retained as part of the community and contribute to the Apache HTTP Server project codebase.



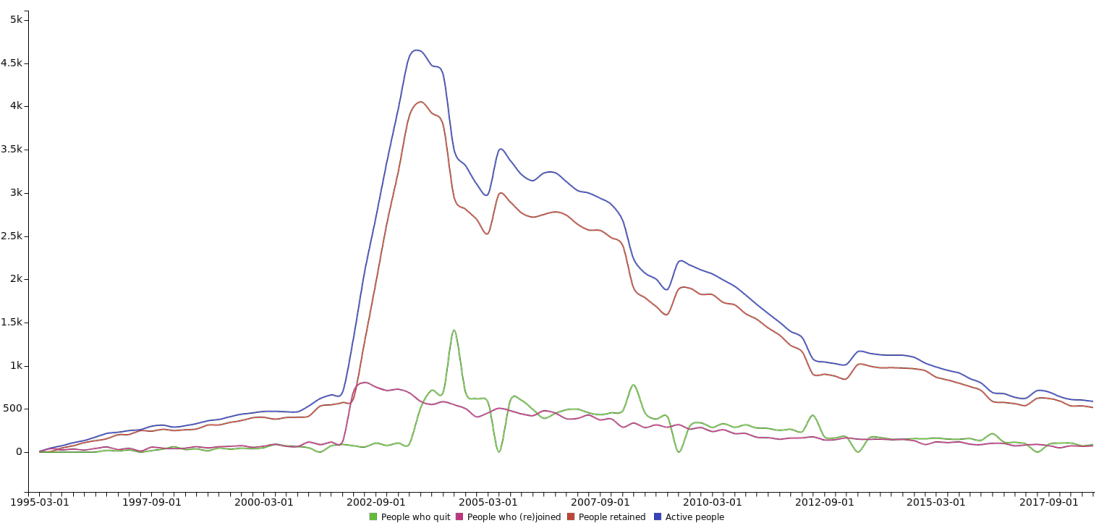
**Fig 5: Apache HTTP Server Project Baseline Contributor Retention Codebase 1996-2018**

The number of active people and those retained follow a similar curve. This is expected as those who are continually active are being retained by the community.

There is also appears to be a relationship between the people who have left and those who rejoin as it has remained at a stable level across the life of the project.

**Indicator 5: Baseline Contributor Retention Email**

The following graph shows how many people have been retained as part of the community and contribute to the mailing list discussions.



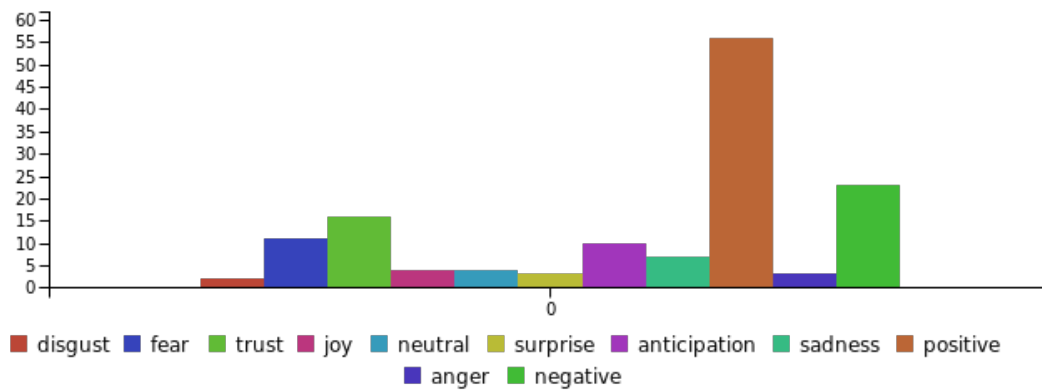
**Fig 6: Apache HTTP Server Project Baseline Contributor Retention – Email 1995-2018**

During the beginning of the project the number of active people and those being retained was growing slowly until around 2001 when this substantially increased. It is also interesting to note during the main activity peak, there was also the highest number of people leaving the project. Both activity and retention have stabilised at a reduced level.



### **Indicator 6: Baseline Sentient Analysis as at 31/10/2018**

The following graph shows the mood analysis of the last 6 months of the Apache HTTP Server project on 31<sup>st</sup> October 2018.



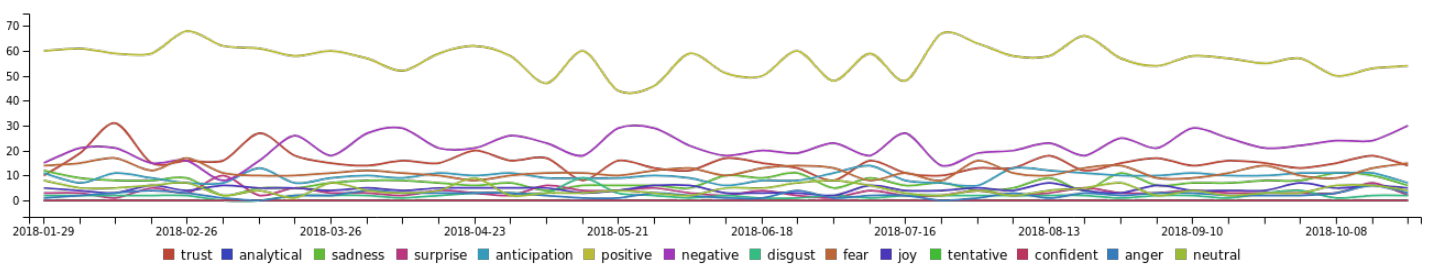
**Fig 7: Apache HTTP Server Project Baseline Sentient Analysis 31/10/2018**

The highest mood sentiment is positivity (55) negativity (23) is second, followed by trust (15), fear (11) and anticipation (10).

The bottom 5 sentiments showing are sadness (8), joy (5), neutral (4), surprise (3), anger (2) and disgust (2).

### **Indicator 7: Baseline Sentient Analysis Over Time**

The following graph shows the mood analysis of the Apache HTTP Server project from the January 2018 – October 2018 time frame.

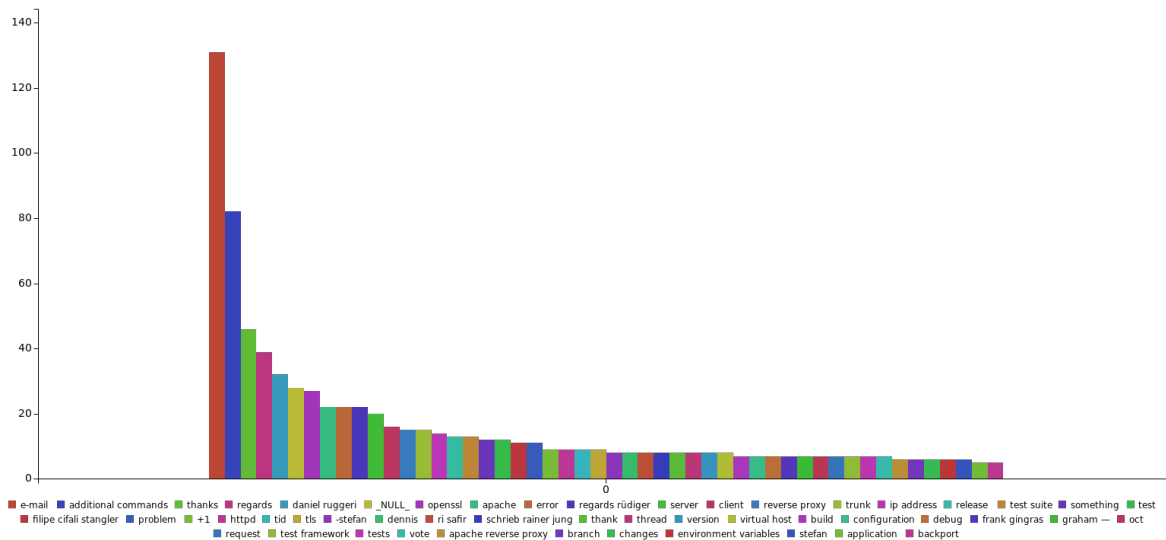


**Fig 8: Apache HTTP Server Project Baseline Mood Analysis January – October 2018**

The analysis over time shows that the positive communication style is established and is the most strongest mood expressed.

### **Indicator 8: Baseline Key Phrase Extraction**

The following graph shows the key phrase extraction analysis of the most common phrases used in the Apache HTTP Server project from January 2018 – October 2018.



**Fig 9: Apache HTTP Server Project Baseline Key Phrase Extraction: January – October 2018**

The phrases can be broken down into several areas:

- General or standard (e.g. email, thanks, regards, names etc.)
- Technical discussions (e.g. httpd, server, openssl, release, client, trunk, ip address, reverse proxy, \_NULL\_, virtual host)
- Collaborative (e.g. error, test suite, problem, thank, test, additional commands, something, request, backport)
- Cultural (+1, apache, vote, thread)

Some phrases appear to be aligned to ASF values as follows:

- Openness: Technical conversations in the open
- Consensus: +1 = indication of consensus
- Collaboration: Polite and collaborative communication

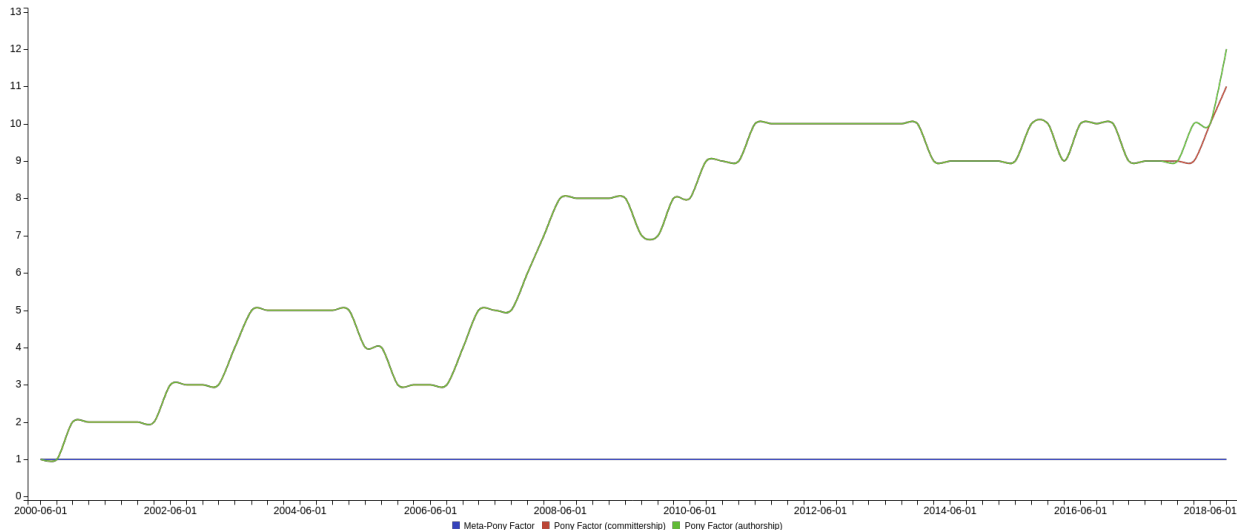
**NOTE:** The KPE appears to be in alignment with the mood analysis of positive communication. No negative phrases are coming out as significant.

## Baseline Comparison with TLPs Over 5 Years

### Indicator 1: Comparison TLP>5Years Pony Factor Codebase

The following graph shows the Pony Factor of the codebase for sub group of ASF projects that have been Top Level Projects (TLP) for more than 5 years.

Note that with the Apache HTTP Server project excluded, the project history begins in 2000.



**Fig 11: Comparison TLP>5 Years Pony Factor 2000-2018**

As with Apache HTTP Server, during the initial project stages the Pony Factor is low (1) and gradually increases over time to its current level (12). Notice that the actual shape of the graph is very similar to the shape and progression of Apache HTTP Server. However during 2006 there was a significant growth and unlike Apache HTTP Server, the Pony Factor is still increasing for these projects.

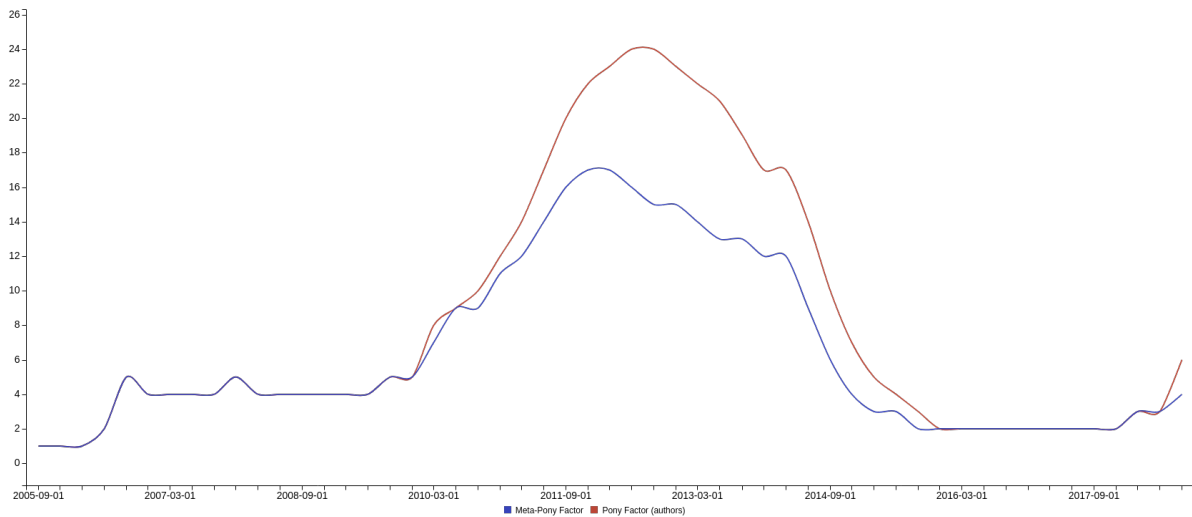
The Pony Factor for committership follows exactly the curve for authorship except for a small current difference (11 instead of 12).

As with Apache HTTP Server, the Meta Pony Factor remains constant at 1 because all interactions are associated with the ASF affiliation.

### Indicator 2: Comparison TLP>5 Years Pony Factor - Email

The following graph shows the Pony Factor for the ASF Comparison projects that have been in the ASF for more than 5 years.

**NOTE:** The mailing list statistics cover the period 2005 to 2018



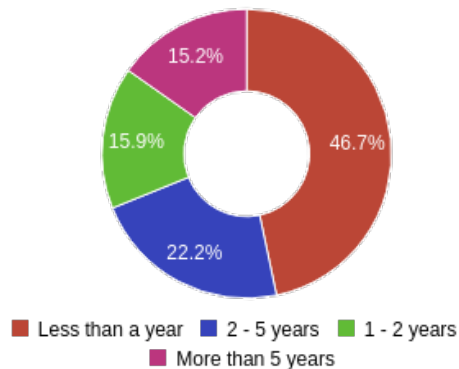
**Fig 12: Comparison TLP> 5 Years Email Pony Factor 2005-2018**

As with the codebase, the mailing list Pony Factor for the comparison projects over 5 years old begins low (1). The number of active mailing list contributors gradually rises to a peak of up to 24 during 2011 then reduces significantly and stabilises. It is currently 6. Once again this is a very similar pattern to Apache HTTP Server.

As with Apache HTTP Server for this metric, the Meta-Pony Factor follows the Pony Factor curve and is currently 4.

**Indicator 3: Comparison TLP> 5 Years Contributor Codebase Experience**

The following graph shows a breakdown the length of time contributors have been contributing to the codebase:

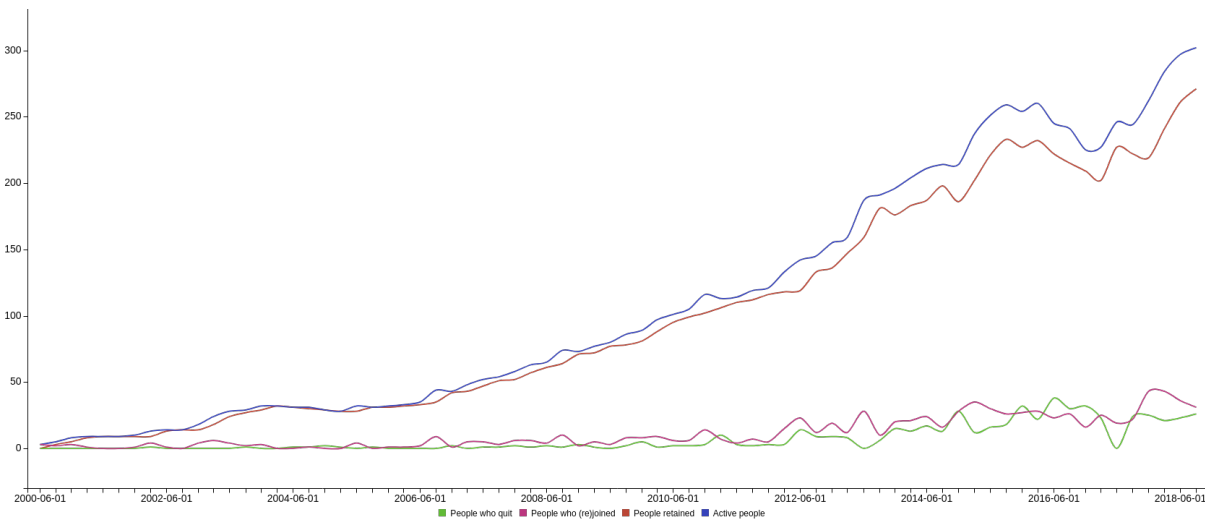


- Over 46% of their contributors have been contributing to the project for less than a year
- Over 38% of contributors have been contributing for between 2 to 5 years
- Only 15 % have more than 5 years experience.
- **NOTE:** This is a significant difference from Apache HTTP Server profile

**Fig 13: Comparison TLP>5 Years Contributor Codebase Experience**

**Indicator 4: Comparison TLP>5 Years Contributor Codebase Retention**

The following graph shows how many people have been retained as part of the community and contribute to the comparison projects codebase.



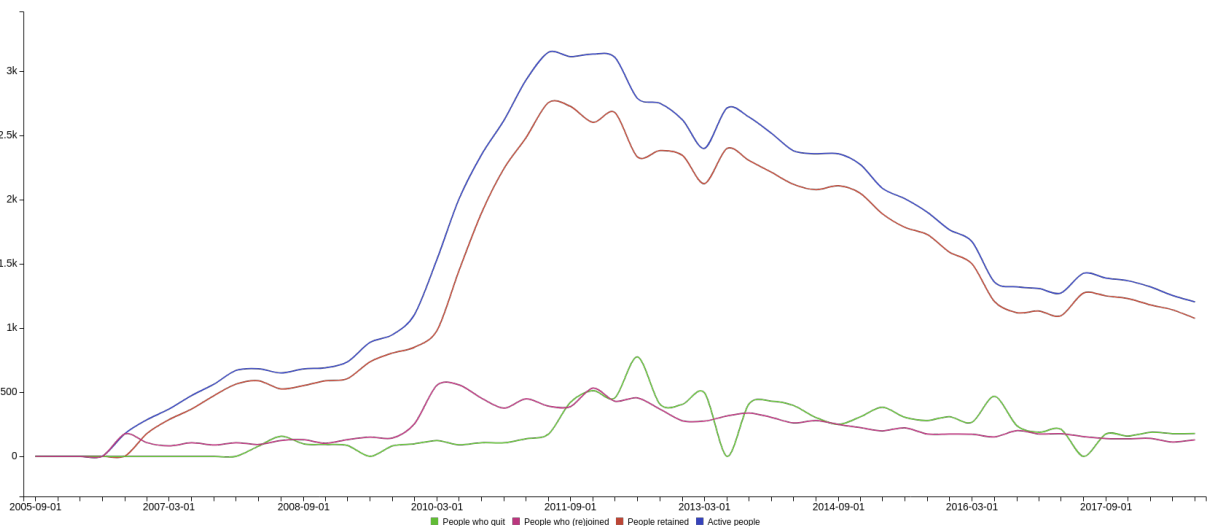
**Fig 14: Comparison TLP>5 Years Contributor Codebase Retention 2000-2018**

This shows a gradual increase of active people and those retained over time. It is a very different profile to Apache HTTP Server and the retention rate is still increasing. As at the time of writing, the current 2018 figures show:

- 302 active people
- 271 retained
- 31 re-joined
- 26 quit.

**Indicator 5: Comparison TLP>5 Years Contributor Retention over Time**

The following graph shows how many people have been retained as part of the community and contribute to the mailing list discussions.



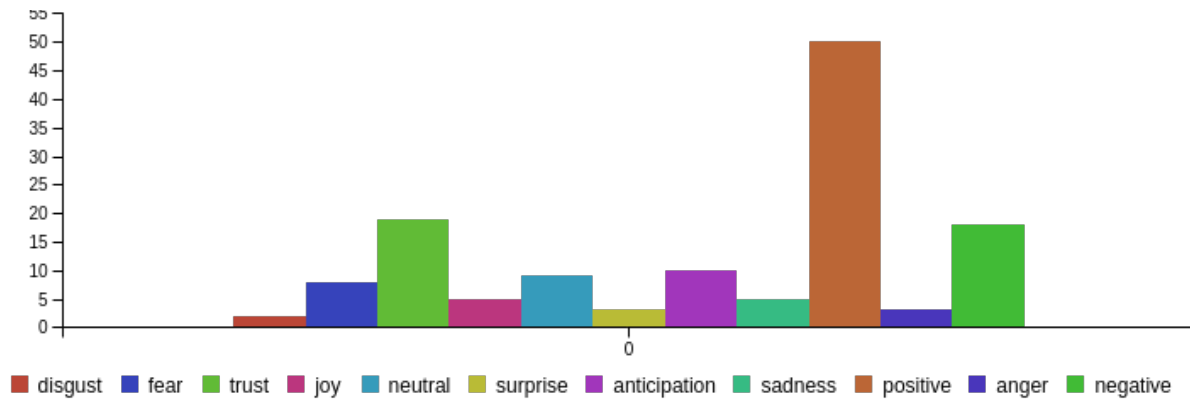
**Fig 15: Comparison TLP>5 Years Email Contributor Retention 2005-2018**

This graph looks very similar to Apache HTTP Server where there is a sharp increase in active

people being retained. This number then reduces over time.

**Indicator 6: Comparison TLP> 5 Years Sentient Analysis as at 31/10/2018**

The following graph shows the consolidated mood analysis of all the comparison projects that have been TLPs for over 5 years on 31<sup>st</sup> October 2018.



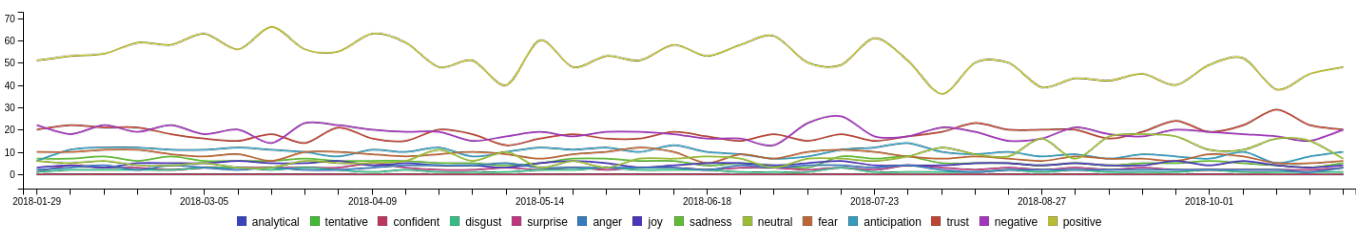
**Fig 16: Comparison TLP > 5 Years Sentient Analysis as at 31/10/2018**

- The highest mood sentiment is positivity (50) , trust (19) is second, followed by negativity (18) and anticipation (10) and neutral (9).
- The bottom five sentiments showing are fear (8), joy (5), sadness (5), surprise (3), anger (3) and disgust (2).

**NOTE** that the 3 strongest sentiments are the same as for Apache HTTP Server and similarly distributed. There are some fluctuations in the other sentiments but they are not significant.

**Indicator 7: Comparison TLP>5 Years Sentient Analysis Over Time**

The following graph shows the mood analysis of the for projects that have been TLP for over 5 years.



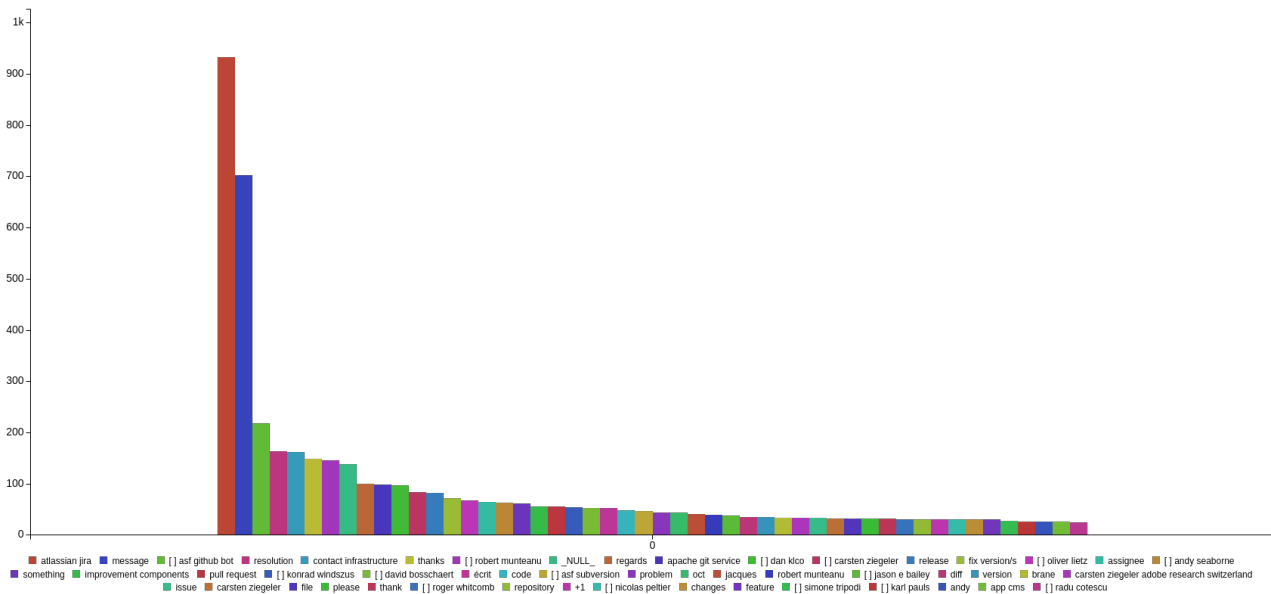
**Fig 17: Comparison TLP>5 Years Sentient Analysis Over Time January – October 2018**

The analysis over time shows that the positive communication style is established and is the most strongest mood expressed. This is very similar to the Apache HTTP Server baseline model.

**Indicator 8: Comparison TLP> 5 Years Key Phrase Extraction**

The following graph shows the key phrase extraction analysis of the most common phrases used in

the comparison projects excluding Apache HTTP Server from January 2018 – October 2018.



**Fig 18: Comparison TLP>5 Years Key Phrase Extraction**

The phrases can be broken down into several areas:

- General or standard (e.g. message, thanks, email, regards, names etc.)
- Technical discussions (e.g. \_NULL\_, additional commands, release, error, file, repository. code )
- Collaborative (e.g. atlassian jira, asf github bot, contact infrastructure, assignee, something, improvement components, pull request, problem, issue, changes, feature, thank)
- Cultural (e.g. +1, resolution )

Several cultural indicators are aligned to ASF values as follows:

- Openness: Technical conversations in the open
- Consensus: +1 = indication of consensus
- Collaboration: Polite communication, information requests, issue assignment, problem resolution

This KPE profile is very similar to Apache HTTP Server and many common phrases are being used. The +1 consensus indicator is culturally significant.

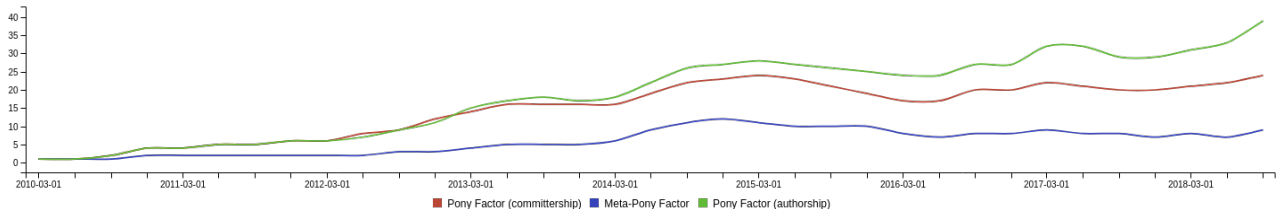
**NOTE:** The KPE is also in alignment with the mood analysis which highlights positive communication. There are a lot of collaborative phrases that are being extracted and no negative phrases are coming out as significant.

## Baseline Comparison with TLP <= 5 Years

### Indicator 1: Comparison TLP <= 5 Years Pony Factor Codebase

The following graph shows the Pony Factor of the codebase for all the Apache comparison projects excluding Apache HTTP Server that have been in the ASF less than 5 years as a top Level Project.

**IMPORTANT NOTE:** The project history begins in 2010. This is because the projects went through Apache Incubator before graduating to become a Top Level Project.



**Fig 19: Comparison TLP <= 5 Years Codebase Pony Factor 2010-2018**

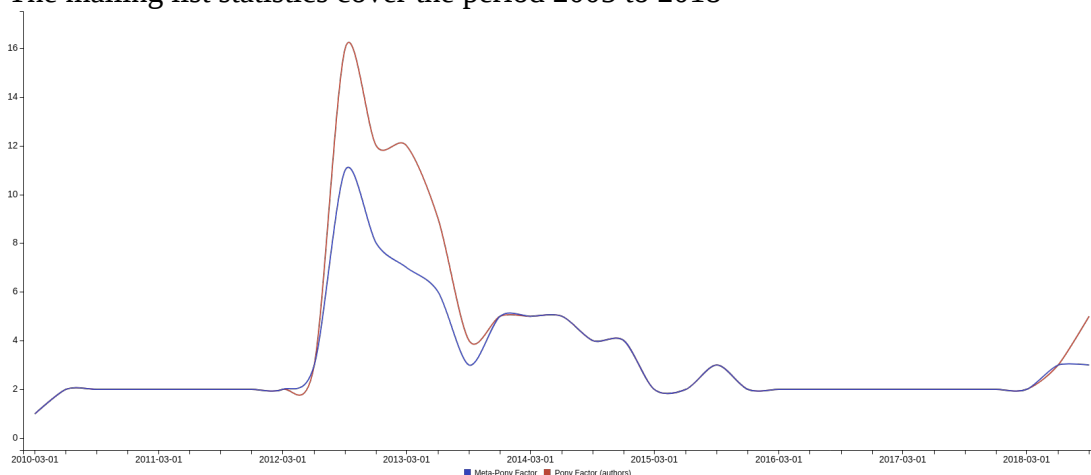
As with Apache HTTP Server during the initial project stages the Pony Factor is low (1) and gradually increases over time to its current level (39). Notice that the actual shape of the graph is not the same shape as the Apache HTTP Server curve. This one is more gradual with no sharp changes.

The Pony Factor for committership follows the same trend as the curve for authorship except with a significant difference (24 instead of 39). Notice that the Meta Pony Factor in blue is not constant at 1 but varies. At its highest in 2015 it is showing as 10 and has now reduced to 9. This is interesting because it is the first metric showing more than one affiliation for contributors to the codebase. This could be as result other version control repositories (e.g. github) being used for newer projects rather than Apache Subversion.

### Indicator 2: Comparison TLP <= 5 Years Email Pony Factor

The following graph shows the Pony Factor for the ASF Comparison projects that have been in the ASF for more than 5 years.

NOTE: The mailing list statistics cover the period 2005 to 2018



**Fig 20: Comparison TLP <= 5 Years Email Pony Factor 2010-2018**

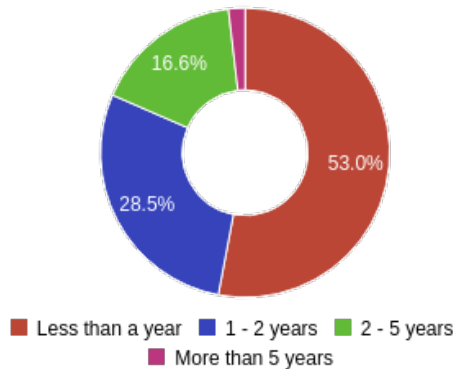


As with the codebase, the mailing list Pony Factor for the comparison for TLPs 5 years or less begins low (1). The number of active mailing list contributors remains constant, perhaps through the incubation phase, then rises significantly to a peak of 16 during 2013 then gradually and stabilises. It is currently 5. This is a very similar pattern to Apache HTTP Server.

As with Apache HTTP Server for this metric, the Meta-Pony Factor follows the Pony Factor curve and is currently 3.

**Indicator 3: Comparison TLP <=5 Years Contributor Experience**

The following graph shows a breakdown the length of time contributors have been contributing to the codebase:

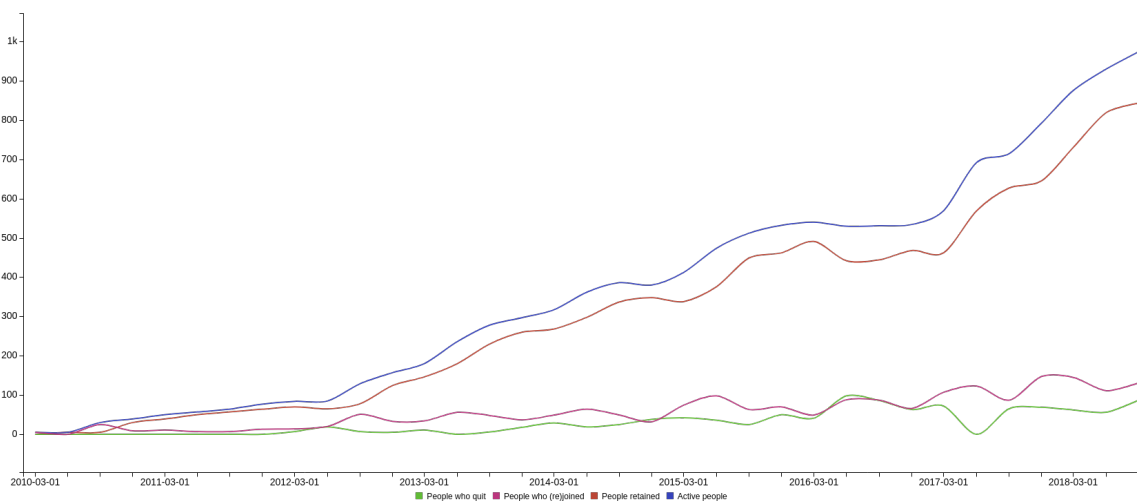


- Over 53% of their contributors have been contributing to the project for less than a year
- Over 45% of contributors have been contributing for between 2 to 5 years
- Only 2 % have more than 5 years experience.
- **NOTE:** This is a significant difference from Apache HTTP Server profile

**Fig 21: Comparison TLP <=5 Years Contributor Experience**

**Indicator 4: Comparison TLP <=5 Years Contributor Retention**

The following graph shows how many people have been retained as part of the community and contribute to the comparison projects codebase.



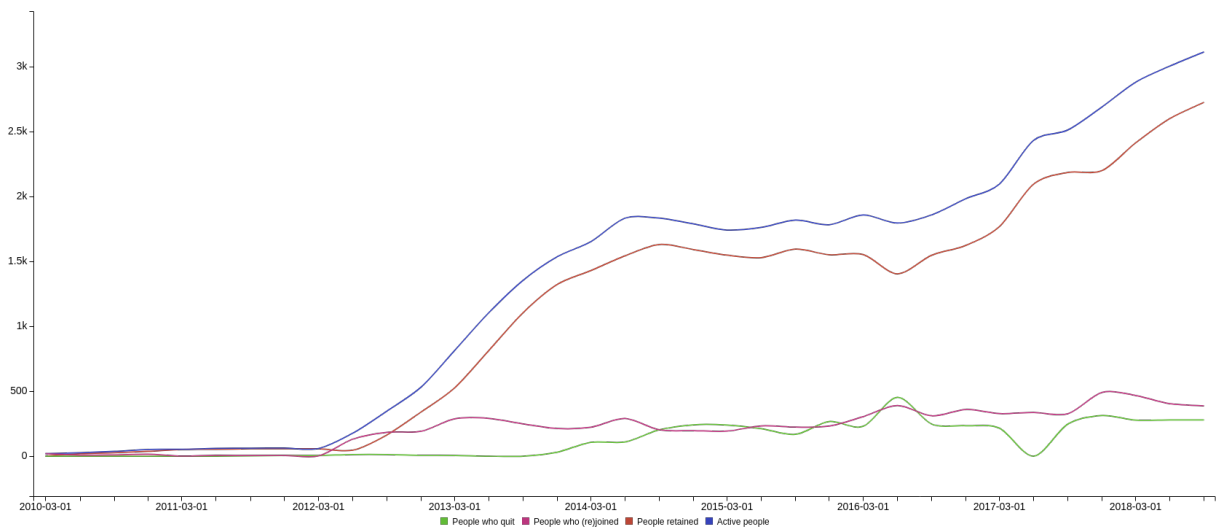
**Fig 22: Comparison TLP <= 5 Years Codebase Contributor Retention 2010-2018**

This shows a gradual increase of active people and those retained over time. It is a very different profile to Apache HTTP Server and the retention rate is still increasing. As at the time of writing, the current 2018 figures show:

- 974 active people
- 843 retained
- 131 rejoined
- 87 quit.

**Indicator 5: Comparison TLP<=5 Years Contributor Retention Over Time**

The following graph shows how many people have been retained as part of the community and contribute to the mailing list discussions.

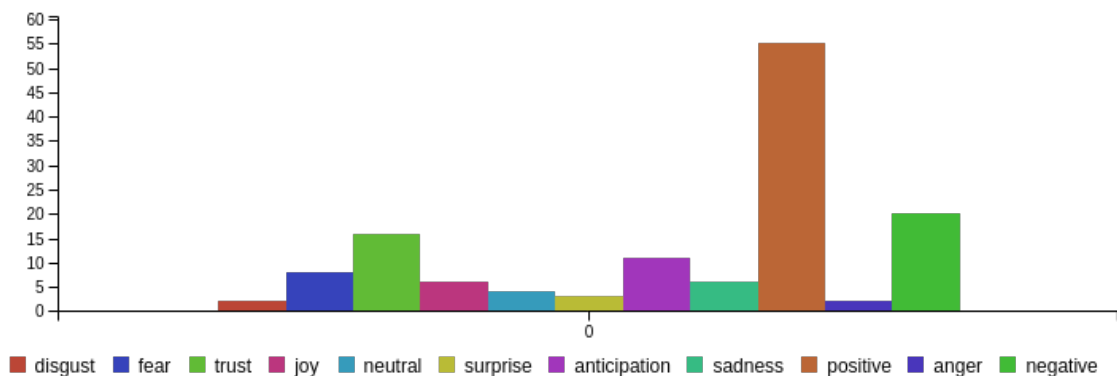


**Fig 23 Comparison TLP<=5 Years Email Contributor Retention 2005-2018**

This graph does not look the Apache HTTP Server baseline. Instead it has a gradual increase over time that is still rising. This means that the number of contributors is actively growing and that they are being retained.

**Indicator 6: Comparison TLP<=5 Years Sentient Analysis as at 31/10/2018**

The following graph shows the consolidated mood analysis of all the comparison projects over 5 years on 31<sup>st</sup> October 2018.



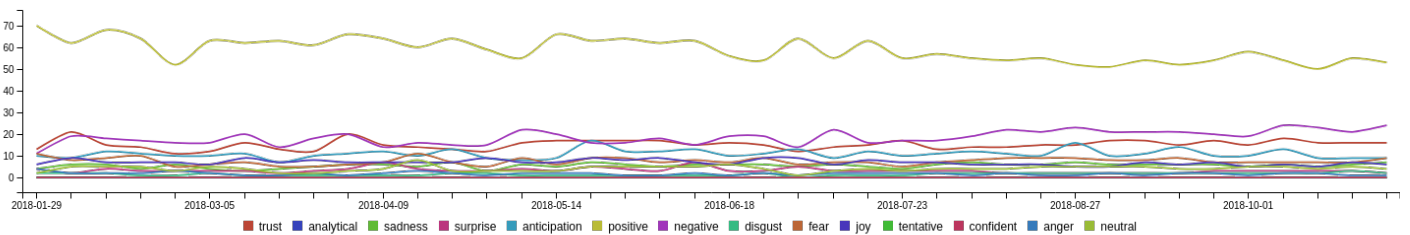
**Fig 24: Comparison TLP<=5 Years Sentient Analysis as at 31/10/2018**

- The highest mood sentiment is positivity (56) , negativity (20) is second, followed by trust (15) and anticipation (11) and fear (8).
- The bottom five sentiments showing are joy (6), sadness (6), neutral (4), surprise (3), anger (2) and disgust (2).

**NOTE** that the 3 strongest sentiments are the same as for Apache HTTP Server and similarly distributed. There are some fluctuations in the other sentiments but not significant.

**Indicator 7: Comparison TLP<=5 Years Sentient Analysis Over Time**

The following graph shows the mood analysis of the Comparison less than 5 years projects from the January 2018 – October 2018 time frame.

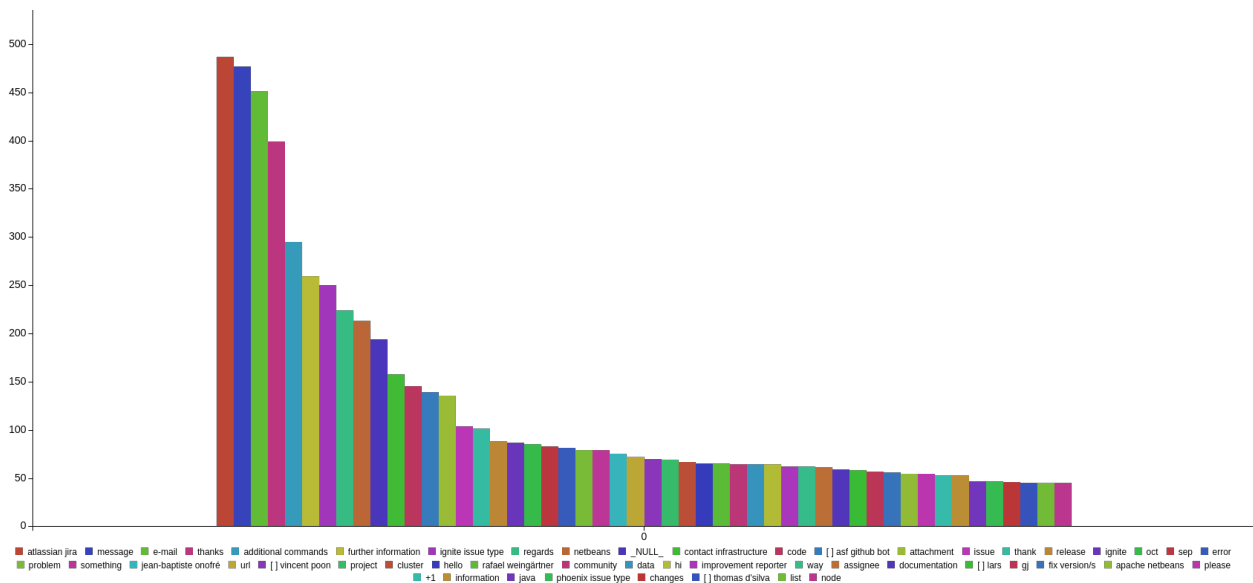


**Fig 25: Comparison TLP<=5 Years Sentient Analysis Over Time January – October 2018**

The analysis over time shows that the positive communication style is established and is the most strongest mood expressed. This is very similar to the Apache HTTP Server baseline model.

**Indicator 8: Comparison TLP<=5 Years Key Phrase Extraction**

The following graph shows the key phrase extraction analysis of the most common phrases used in the comparison projects excluding Apache HTTP Server from January 2018 – October 2018.



**Fig 26: Comparison TLP<= 5 Years Key Phrase Extraction**

The phrases can be broken down into several areas:

- General or standard (e.g. message, thanks, email, regards, names etc.)
- Technical discussions (e.g. \_NULL\_, additional commands, release, error, file, repository. code )
- Collaborative (e.g. atlassian jira, contact infrastructure, further information, something, problem, issue, error, thank, project, documentation, please, attachment, improvement reporter)
- Cultural (e.g. community, +1, list)

Several cultural indicators are aligned to ASF values as follows:

- Openness: Technical conversations in the open
- Consensus: +1 = indication of consensus
- Collaboration: Polite communication, information requests, issue assignment, problem resolution
- Community: mailing list is the communication medium

This KPE profile is very similar to Apache HTTP Server and many common phrases are being used including 'community'. The +1 consensus indicator is culturally significant.

**NOTE:** The KPE is also in alignment with the mood analysis which highlights positive communication. There are a lot of collaborative phrases that are being extracted and no negative phrases are coming out as significant.

# Conclusions

This paper focussed on examining the culture that has evolved at the Apache Software Foundation (ASF) and investigating the extent to which their values and culture can be effectively transmitted to new projects. It used a set of tools and indicators to create a cultural baseline based on the values and behaviours shown by the first ever ASF project, Httttd. A set of 8 indicators was used to create the baseline by mining the data publicly available from the ASF project archives.

For the Apache HTTP Server baseline, the following indicators were used to capture, highlight and measure a potential range of cultural elements:

- **Pony Factors:** Diversity of the community, confirmation that merit is being rewarded, indication of community growth, retention of contributors
- **Sentient Analysis:** Dominant emotions being displayed in community interactions, communication style, overall mood of the communication (negative, positive or neutral) over time
- **Key Phrase Analysis:** Identifying the most common important phrases and words being used, indication of collaboration, identifies the use of unique cultural language

Fifteen ASF projects were selected and divided into two groups for comparison against the Apache HTTP Server cultural baseline model. The groups were as follows:

- Projects that have been ASF Top Level Projects (TLPs) for **Over 5 Years**
- Project that have been ASF Top Level Projects (TLPs) for **5 Years or Less**

The results of the comparison against the Apache HTTP Server baseline is shown below:

## **Pony Factors**

The older projects appeared to follow the Apache HTTP Server baseline, showing that merit is being rewarded (i.e. no difference between codebase authorship and committership). These projects also showed a significant increase in Pony Factor meaning increased diversity of contributors and also community growth.

The younger projects did not appear to follow the Apache HTTP Server baseline and had a difference between codebase authorship and committership showing that they are perhaps not as frequent in recognising merit based on contributor activity. The increase in Pony Factor shows that they are also increasing the diversity of their contributors but not at the same rate as the older projects.

## **Contributor Retention**

For the older projects, over 45% of their total contributors are new (i.e. have been contributing for less than a year) which means these projects are very successful at attracting new people. Nearly 40% of their contributors have 2 – 5 years experience meaning they have a good flow of people coming into the community and staying. Contributor retention for these older projects is good and very different to, and a lot higher than the Apache HTTP Server baseline.

For the younger projects, over 50% of their contributors are new and have been contributing for less than a year which means these projects are also very successful at attracting people. Over 45% of contributors have 2 -5 years experience meaning they have a good flow of people coming into the

community. Contributor retention is extremely high (almost triple the number of the older projects) which is significantly different to the Apache HTTP Server baseline.

### **Sentient Analysis**

Both older and younger projects have similar profiles to the Apache HTTP Server baseline. The communication style was consistently positive and although negativity was listed in the top 5 for all groups, over time it did not affect the overall communication mood.

### **Key Phrase Analysis**

Once again both older and younger projects have similar profiles to the Apache HTTP Server baseline. They all showed elements of standard everyday communication and technical interactions. It was interesting to see that they also showed indications of cultural expression of ASF values, such as openness, collaboration and community.

The most significant cultural element which appeared in the baseline as well as both older and younger projects was the “+1” indicator. This is unique to the ASF as an indicator of consensus.

### **Summary**

The results show that although Apache HTTP Server was a very successful project during the past, it is currently going through a period of potential decline in attracting new contributors. Even so, the culture that created the Apache HTTP Server project is still very strong and is being transmitted to other ASF projects. The values, communication and expected behaviours are reinforced in the community via their mailing lists.

It appears that as ASF projects age they become more confident and proficient in demonstrating key aspects of the culture and recognising merit. Older projects appear to be experienced at retaining a good mix and flow of contributors, while the younger projects are the most successful at attracting new contributors.

This paper highlights that values and culture can be transmitted and the extent to which the transmission occurs is dependant on how long a project is exposed to the culture. The longer the exposure, the more naturally the culture appears to manifest itself. Perhaps the focus of certain cultural elements can be interpreted and adopted more easily than others, and this could be the areas where the younger projects are more successful than the older projects. Ongoing exchange between older and younger projects could provide good ways to create cultural balance in both.

Culture is always something that is evolving so it would be useful to be able to build on the results of this paper with further research in the following areas:

- Analysis of non ASF related open source projects and comparison with an ASF projects to confirm if ASF values and behaviour are cultural specific / unique
- Analysis of an ASF incubating project that has a completely different cultural profile to see if cultural changes occur as part of the project evolution to graduation to Top Level Project
- Investigation of ASF projects that have gone into decline to see if the decline could have been predicted using any of the indicators used in this paper
- Interviewing or surveying ASF projects about their interpretation of the Apache Way culture
- Interviewing people (e.g. Directors, Board Members, Officers) involved with managing open source governance to collect their feedback about the values and culture that they expected or want to create and do a comparison of the results.

## References

Duenas, J. C., Parada G, H. A., Cuadrado, F., Santillan, M., & Ruiz, J. L. (2007). Apache and eclipse: Comparing open source project incubators. *IEEE Software*, 24(6), 90-98. doi:10.1109/MS.2007.157

Perspectives on Free and Open Source Software, Feller, Fitzgerald, Hissam, Lakhani  
Lakhani K.R & Wolf R.G 2001 Why Hackers Do what they Do?

Mulder, P. (2013). *Organizational Culture Model by Edgar Schein*. Retrieved [insert date] from ToolsHero: <https://www.toolshero.com/leadership/organizational-culture-model-schein/>

Ceraso, A., & Pruchnic, J. (2011). INTRODUCTION: OPEN SOURCE CULTURE AND AESTHETICS. *Criticism*, 53(3), 337-375. Retrieved from <http://ezproxy.techlib.cz/login?url=https://search-proquest-com.ezproxy.techlib.cz/docview/912509449?accountid=119841>

Benkler, Y., & Nissenbaum, H. (2006). *Commons-based peer production and virtue*. *Journal of Political Philosophy*, 14(4), 394-419. doi:10.1111/j.1467-9760.2006.00235.x

Birukou A., Blanzieri E., Giorgini P., Giunchiglia F. (2013) A Formal Definition of Culture. In: Sycara K., Gelfand M., Abbe A. (eds) *Models for Intercultural Collaboration and Negotiation*. *Advances in Group Decision and Negotiation*, vol 6. Springer, Dordrecht

Christoph Brumann, "Writing for Culture: Why a Successful Concept Should Not Be Discarded," *Current Anthropology* 40, no. S1 (February 1999): S1-S27.

Pony Factor Definition and Example: <https://ke4qqq.wordpress.com/2015/02/08/pony-factor-math/>

Allen, D. (2003). Cultural conundrum. *Nursing Standard (through 2013)*, 17(43), 16-8. Retrieved from <http://ezproxy.techlib.cz/login?url=https://search-proquest-com.ezproxy.techlib.cz/docview/219813712?accountid=119841>

Intrinsic and Extrinsic Motivations: Classic Definitions and New Direction, Ryan. R.M & Deci E.L. (2000)

Markus, M.L. *J Manage Governance* (2007) 11: 151.