



Asynchronous Decision Making – why and how?

Bertrand Delacrétaz - Principal Scientist, Adobe - Board Member, Apache Software Foundation
FOSDEM Brussels January 2018 - @bdelacretaz, grep.codeconsult.ch



**“Of course we can make fast decisions ...
once we have considered the 4872 factors.”**

Images are from Adobe Stock, unless otherwise stated.

Be
Klebebande

**I will NOT attend
your meeting
I'm an Open Source person!**

Bertrand Delacrétaz
@bdelacretaz, grep.codeconsult.ch

Principal Scientist, Adobe, Basel
Apache Software Foundation Member and Director

ApacheCon Séville, November 2016
slides revision 2016-11-14



opensource.com

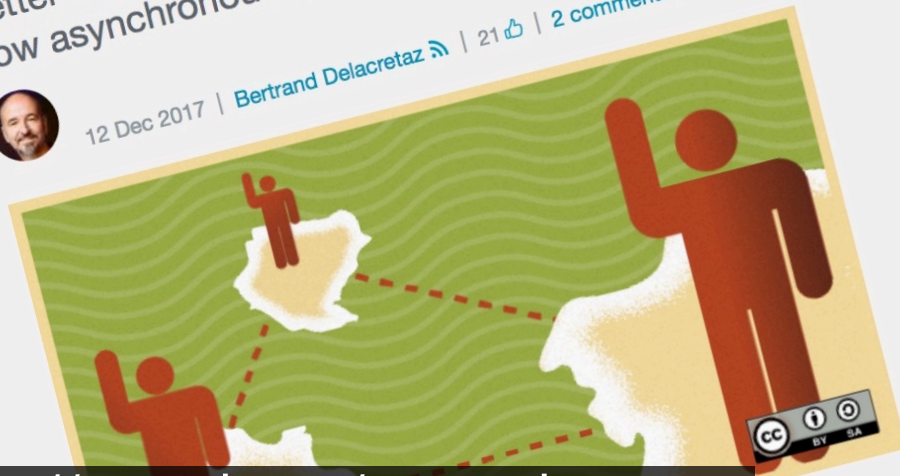


Asynchronous decision-making: Helping remote teams succeed

Better communication with fewer meetings isn't a pipe dream. Here's how asynchronous decision-making can make it happen.



12 Dec 2017 | Bertrand Delacretaz | 21 likes | 2 comments



<https://tinyurl.com/async-decisions>
on opensource.com

<https://s.apache.org/PMvk>
on blogs.apache.org/foundation

The Apache Software Foundation Blog

MONDAY FEBRUARY 06, 2017

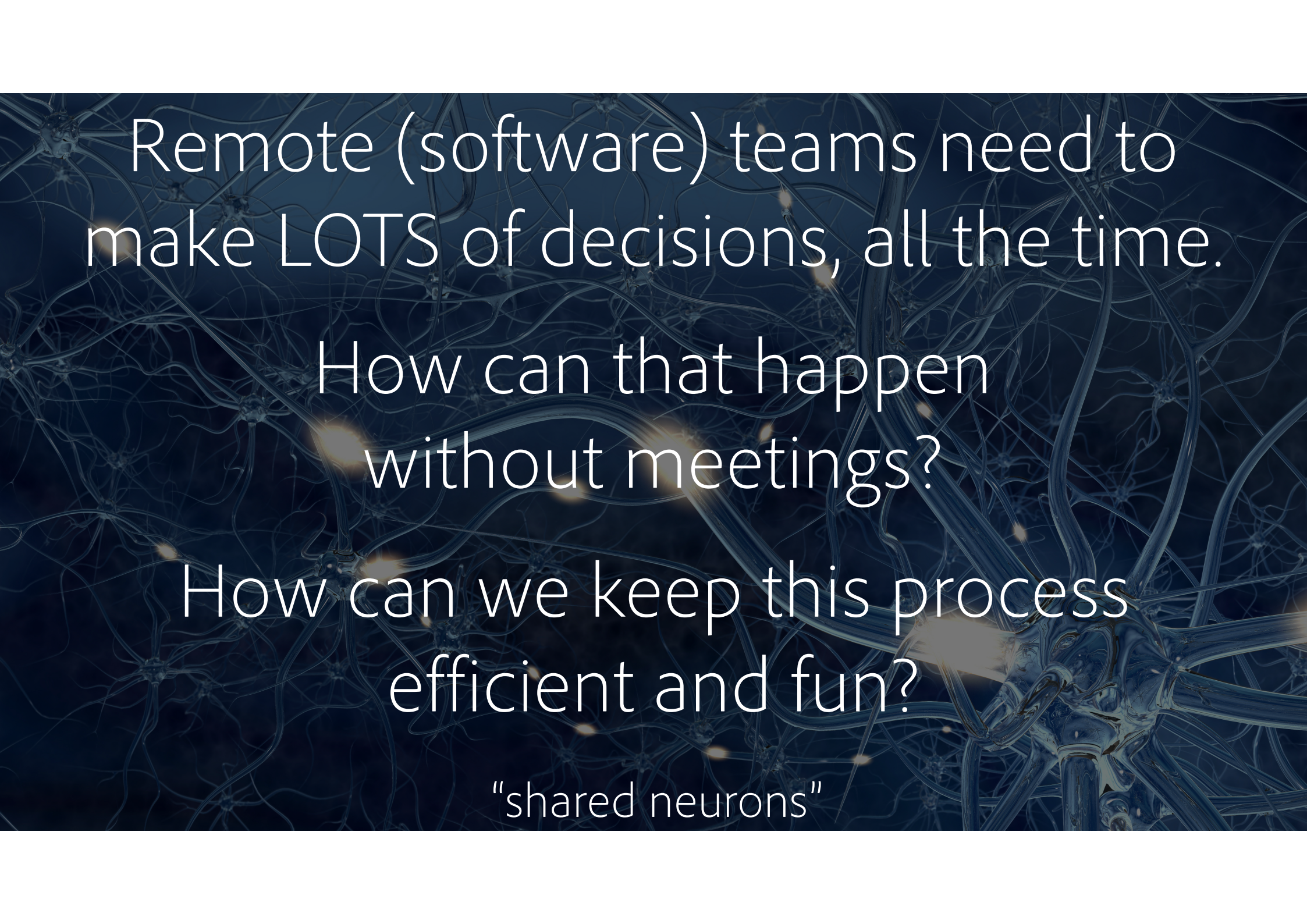
« The Apache News... | Main | The Apache Software... »

Success at Apache: Asynchronous Decision Making

by Bertrand Delacretaz

Asynchronous decision making is a key enabler of our geographically and culturally distributed Open source teams. In this post I'll explain the ingredients that make it work at ASF.

I became active in the ASF in 2001 via Gianugo Rabellino - he was the one who started the ASF with Apache Fop about me donating the jfor XLS-FO to RTF converter that I had developed earlier. It was already too late to uninvent RTF which is a terrible format, but I was currently a member of the Board of Directors of the ASF and have been giving presentations about what makes the ASF tick in terms of Shared Neurons.



Remote (software) teams need to
make LOTS of decisions, all the time.

How can that happen
without meetings?

How can we keep this process
efficient and fun?

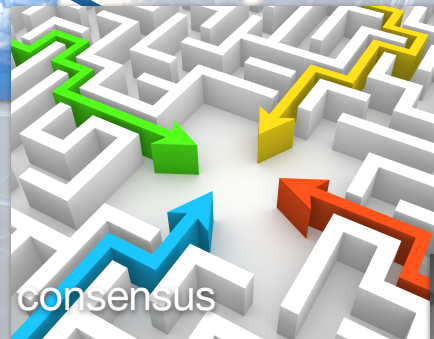
“shared neurons”



WHAT

is Asynchronous Decision Making ?

Four steps to a decision



Unless formally required by your organization, none of these require people to be in the same “place” at the same time: *everything can happen asynchronously.*

Tools for asynchronous decisions



shared async
comms channel

free-form, chatty



shared case
management tool

structured,
chronology

No meetings required


More time to think

More precise in a foreign language or if you're shy



WHY

do we need Asynchronous Decision Making ?

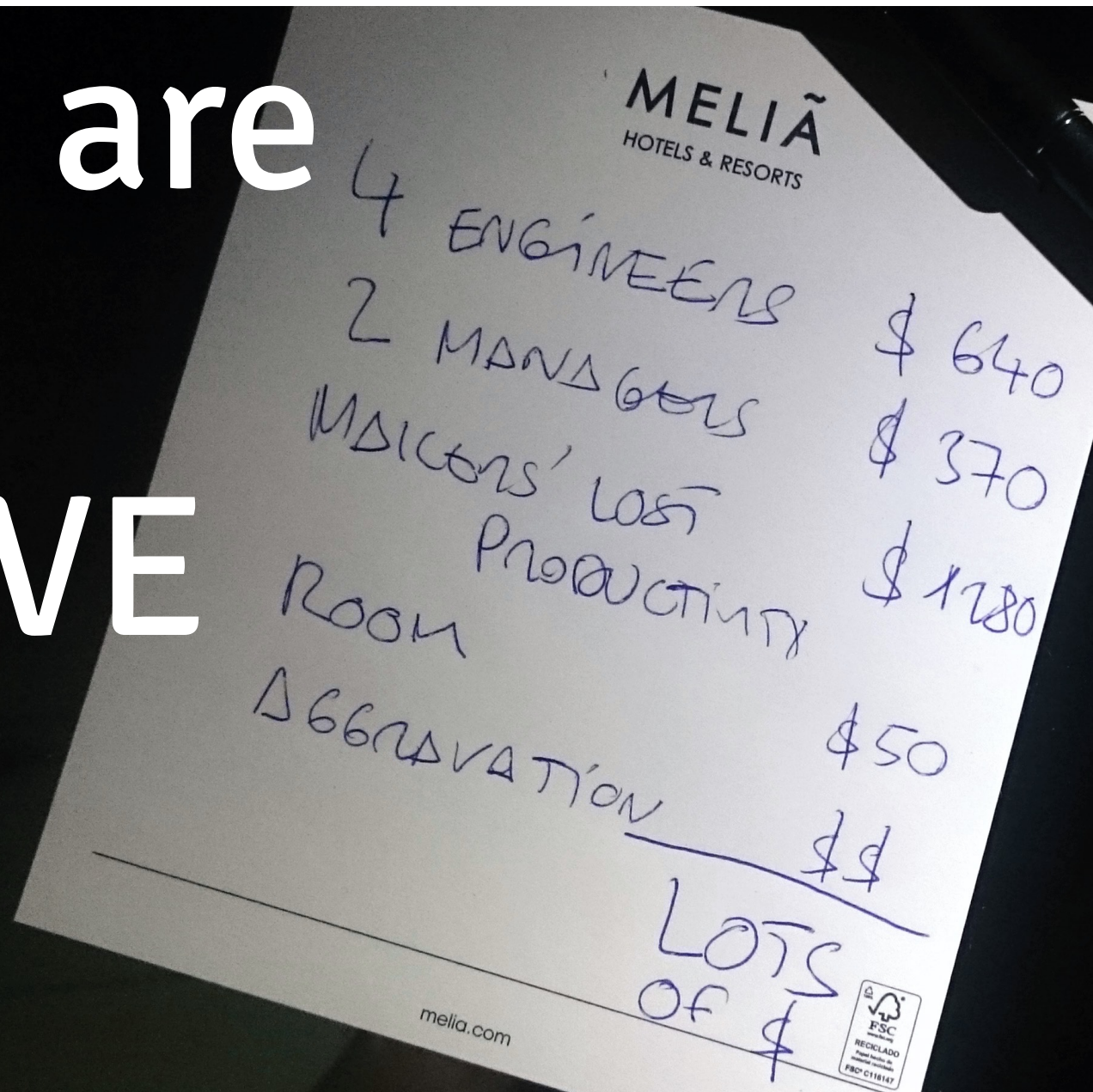
A bearded man with a focused expression is working in a workshop. He has a pencil in his mouth and is holding a wooden mallet in his right hand, poised to strike a piece of wood. The background is dark and filled with various tools and materials, suggesting a busy, hands-on environment.

On a Maker's Schedule,
a one-hour meeting
might waste half a day

On a Manager's Schedule
it's just another one-hour slot.

Credit to Paul Graham, blog post: <https://s.apache.org/ms>

Meetings are (crazily) EXPENSIVE





failed meetings?

are you on mute?

Bob forgot forgot Bob!

what's the goal?

wrong topic

say again?

we'll decide next week...

unprepared

who's in charge?



HOW

does Asynchronous Decision Making work?

Tools for asynchronous decisions



Apache projects (where I come from) use their **dev mailing lists** as a shared asynchronous communications channel and **issue trackers** for case management.

Your mileage may vary - but the principles are not limited to software development.

Building consensus



Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

Quorum

The Federal Council constitutes a quorum when at least four members are present. It decides by consensus whenever possible, and sometimes also holds a vote. In the event of a tied vote, the president's vote counts twice.

Consensus: *widespread agreement among people who have decision power.*

Natural consensus is best.
When that doesn't emerge,
clear voting rules can help!



Because one of the fundamental aspects of accomplishing things within the Apache framework is doing so by consensus, there obviously needs to be a way to tell whether it has been reached. This is done by voting.

There are essentially three types of vote:

1. Code modifications,
2. Package releases
3. Procedural

Votes on procedural issues follow the common format of majority rule unless otherwise stated. That is, if there are more favourable votes than unfavourable ones, the issue is considered to have passed – regardless of the number of votes in each category. (If the number of votes seems too small to be representative of a community consensus, the issue is typically not pursued. However, see the description of [lazy consensus](#) for a modifying factor.)

Votes on code modifications follow a different model. In this scenario, a negative vote constitutes a [veto](#), which cannot be overridden. Again, this model may be modified by a [lazy consensus](#) declaration when the request for a vote is raised, but the full-stop nature of a negative vote is unchanged. Under normal (non-lazy consensus) conditions, the proposal requires three positive votes and no negative ones in order to pass; if it fails to garner the requisite amount of support, it doesn't – and typically is either withdrawn, modified, or simply allowed to languish as an open issue until someone gets around to removing it.

Votes on whether a package is ready to be released or not use yet a different mechanism: are there at least three binding votes in favour of the release? See more about this [below](#).

BINDING VOTES

Who is permitted to vote is, to some extent, a community-specific thing.

PMC members have formally binding votes, but in general community members are encouraged to vote, even if their votes are only advisory.

IMPLICATIONS OF VOTING

In some cases and communities, the exercise of a vote carries some responsibilities that may not be immediately obvious. For example, in some cases a favourable vote carries the implied message 'I approve **and** I'm willing to help.' Also, an unfavourable vote may imply that 'I disapprove, but I have an alternative and will help with that alternative.'

The tacit implications of voting should be spelt out in the community's guidelines. However, **in no case** may someone's vote be considered invalid if the implied commitment doesn't appear to be met; a vote is a formal expression of opinion, *not* of commitment.

If the **R-T-C** policy is in effect, a positive vote carries the very strong implied message, 'I have tested this patch myself, and found it good.' Similarly, a negative vote usually means that the patch was tested and found to be *not*-good, although the veto (for such it is in this case) may be based on other technical grounds.

EXPRESSING VOTES: +1, 0, -1, AND FRACTIONS¶

The voting process in Apache may seem more than a little weird if you've never encountered it before. Votes are represented as numbers between -1 and +1, with '-1' meaning 'no' and '+1' meaning 'yes.'

The in-between values are indicative of how strongly the voting individual feels. Here are some examples of fractional votes and ways in which they *might* be intended and interpreted:

- +0: 'I don't feel strongly about it, but I'm okay with this.'
- -0: 'I won't get in the way, but I'd rather we didn't do this.'
- -0.5: 'I don't like this idea, but I can't find any rational justification for my feelings.'
- ++1: 'Wow! I like this! Let's *do* it!'
- -0.9: 'I *really* don't like this, but I'm not going to stand in the way if everyone else wants to go ahead with it.'
- +0.9: 'This is a cool idea and I like it, but I don't have time/skills necessary to help out.'

Votes should generally be permitted to run for at least 72 hours to provide an opportunity for all concerned persons to participate regardless of their geographic locations.



ok, ok but in

PRACTICE?

Be
Klebebande

cordova-discuss

The screenshot shows the GitHub repository for 'apache / cordova-discuss'. The repository has 137 commits, 3 branches, and 0 releases. The main branch is 'master'. A recent commit by 'janpio' is shown, titled 'fix link for "Cordova-lib Refactoring proposal"'. Below the commit list, there is a file tree showing 'proposals', 'FAQ.md', and 'README.md'. The 'README.md' file is selected, showing the following content:

cordova-discuss

This is the place where you can initiate proposals for new features/ideas in Cordova.

Adding a Proposal

For adding new proposals and ideas, please review instructions [here](#).

Active Proposals - Help needed

Proposal	Status/Notes
Support for Android N	Proposed - Ongoing
Support for third party package managers - Nuget & cocoapods	Proposed

In Progress Proposals

Proposal	Status/Notes
Templates in Cordova CLI Part 2	Proposed

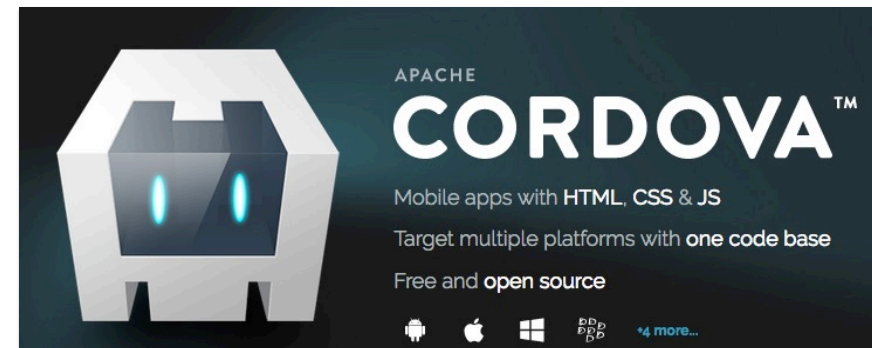
Suggested Process:

1. Create an issue with the description of the feature should be discussed.
2. Others can **discuss** about the proposal with **issue comments**
3. Once there is a rough **consensus**, create a **new file** in this folder with the proposal details.

...

slightly different tools, similar process:
brainstorm -> options -> consensus -> decision
all asynchronous and traceable

<https://github.com/apache/cordova-discuss>



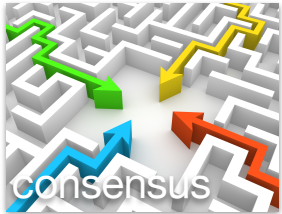
Apache project dev list + tracker

label:sling-dev

1-100 of many

<input type="checkbox"/>	<input type="star"/>	Jason .. Antonio (11)	sling-dev	value level encryption - dev@sling.apache.org Subject: Re: value level encryption EX
<input type="checkbox"/>	<input type="star"/>	Ian .. Carsten, Chetan (5)	sling-dev	Configuration Status dump. - http://localhost:4502/system/console/config/configuration
<input type="checkbox"/>	<input type="star"/>	Satya Deep Maheshwari (2)	sling-dev	What is the significance of escaping illegal JCR characters in JcrValueMap while read
<input type="checkbox"/>	<input type="star"/>	Carsten, Robert (4)	sling-dev	[git] Where is launchpad.base ? - the 'sling' team to get write access and I'll push the c
<input type="checkbox"/>	<input type="star"/>	Chetan .. Robert (5)	sling-dev	Etiquettes while merging PR from contributors - According to [1], there are 3 options: 'r
<input type="checkbox"/>	<input type="star"/>	Nicolas Peltier (2)	sling-dev	[pipes] nashorn engine not accessible from custom servlet - version of sling I can think
<input type="checkbox"/>	<input type="star"/>	Carsten, Bertrand (4)	sling-dev	[svn/git] Whiteboard not writeable anymore - apache/sling-whiteboard > > and I guess
<input type="checkbox"/>	<input type="star"/>	Chris .. Bertrand (6)	sling-dev	Contribute Slick? - On Tue, Oct 24, 2017 at 10:55 AM, Nicolas Peltier wrote: > Could v
<input type="checkbox"/>	<input type="star"/>	Robert .. Radu, Carsten (12)	sling-dev	[git] Migration to git COMPLETE - apache/sling-org-apache-sling-scripting-sightly-te >
<input type="checkbox"/>	<input type="star"/>	Roy .. Daniel, Davide (7)	jackra.../jackrabbit-users sling-dev	Daily content sync between environments - leverage the sling
<input type="checkbox"/>	<input type="star"/>	Nicolas, Robert (2)	sling-dev	connction - On Mon, 2017-10-23 at 10:32 +0200, Nicolas Peltier wrote: > Hey, > > th
<input type="checkbox"/>	<input type="star"/>	Robert .. Bertrand, Ian (9)	sling-dev	[git] Migration to git repositories sort of complete (was: Migration to git starting, p... - a
<input type="checkbox"/>	<input type="star"/>	Robert Munteanu	sling-dev	[git] Migration to git starting, please hold commits in 30 minutes from now - Hi, As ann
<input type="checkbox"/>	<input type="star"/>	Robert .. Ian, Bertrand (16)	sling-dev	[git] Migration scheduled for 18 Oct, 22:00 CEST and final notes - soon", probably this
<input type="checkbox"/>	<input type="star"/>	Robert, Konrad, Ian (7)	sling-dev	Prototype of building github pull requests with Jenkins - apache/sling/blob/trunk/.travis
<input type="checkbox"/>	<input type="star"/>	Ian .. Chetan, Robert (25)	sling-dev	Depending on Oak 1.7.x - Oak then Sling warrants the extra work of experimental bra

Sling developers mailing list



Issues: Open Issues

Order by Priority

- SLING-7157 metatype.properties file must not b...
- SLING-7043 Exporting com.codahale.metrics.M...
- SLING-6856 NPE in scheduler whiteboard handler
- SLING-7231 Move to owasp sanitizer library**
- SLING-5421 Allow JCR installer to recover from ...
- SLING-7014 Inheritance of configurations from p...
- SLING-6865 Default Config sling/xss/config.xml ...
- SLING-6344 Support filter.xmls being generated ...
- SLING-5417

Details

Type: Improvement

Status: OPEN

Priority: Critical

Resolution: Unresolved

Affects Version/s: None

Fix Version/s: XSS Protection API 2.0.2

Component/s: XSS Protection API

Sling "issue" tracker

Maybe a [VOTE]

ASF Board of Directors / Swiss Government



And a project report header and discussion space is as simple as this:

```
E. Apache Blazinator Project [Bob Blazer / Bertrand]
See Attachment E
[ Blazinator.
  approved: bd, mm, dd, db, jc, ldv
  comments:
    bd: Not sure why LEGAL-123 blocks their release
    ldv: They are waiting for the committer to supply
        an updated iCLA as the received one was
        incomplete.
    bd: Ok, thanks, approving the report then.
]
```

*Dead simple single text file + version control
for case management.*

<https://www.apache.org/foundation/board>



Schweizerische Eidgenossenschaft
Confédération suisse
Confederazione Svizzera
Confederaziun svizra

- The agenda is presented in colour-coded form. The **orange list** contains uncontested items of business, which are usually approved swiftly. The **blue list** consists of responses to parliamentary requests. The items on the **white list** are discussed and decided one by one due to their political importance; occasionally, the discussions can be spread over a number of meetings. And the green list contains confidential items of business, which are also discussed in detail.

*Color-coded lists of agenda items on paper
to structure meeting.*

<https://www.admin.ch/gov/en/start/federal-council/tasks/decision-making/federal-council-meeting.html>

Semi-async meetings - structured async preparation - similar principles!



coda

are we done yet?

Be
Klebebande

This works!



...as demonstrated by some world-changing Open Source projects over the years!

Switching between tools requires practice - adapt to your culture!

The exact tools are not important but their *roles* are:

**central async channel +
consensus building rules
case management**

I'm @bdelacretaz, thanks!

Reading list: <https://pinboard.in/u:bdelacretaz/t:collaboration>

Requirements for open development channels:

<https://github.com/bdelacretaz/opendev-channel-requirements>



Slides licensed under
a Creative Commons Attribution-ShareAlike 4.0
International License.

