Developer's Guide to Contributing Code to Kafka

Mickael Maison
Tom Bentley

Who are we?

Mickael

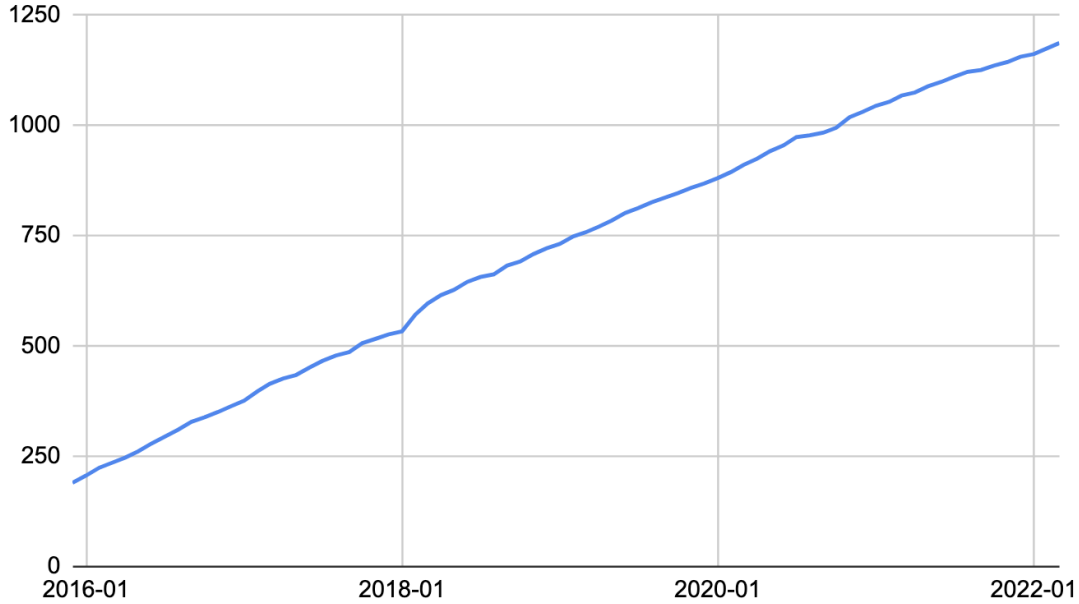
- Contributor since 2016
- Committer since 2019
- PMC member since 2020

Tom

- Contributor since 2017
- Committer since 2021
- PMC member since 2021

Apache Kafka: An Open Source Project

Become a contributor!



https://github.com/apache/kafka/blob/trunk/CONTRIBUTING.md

Collaboration

Dev Mailing List: https://kafka.apache.org/contact

Wiki: https://cwiki.apache.org/confluence/display/KAFKA/Index

Bug tracker: https://issues.apache.org/jira/projects/KAFKA/issues

Code: https://github.com/apache/kafka

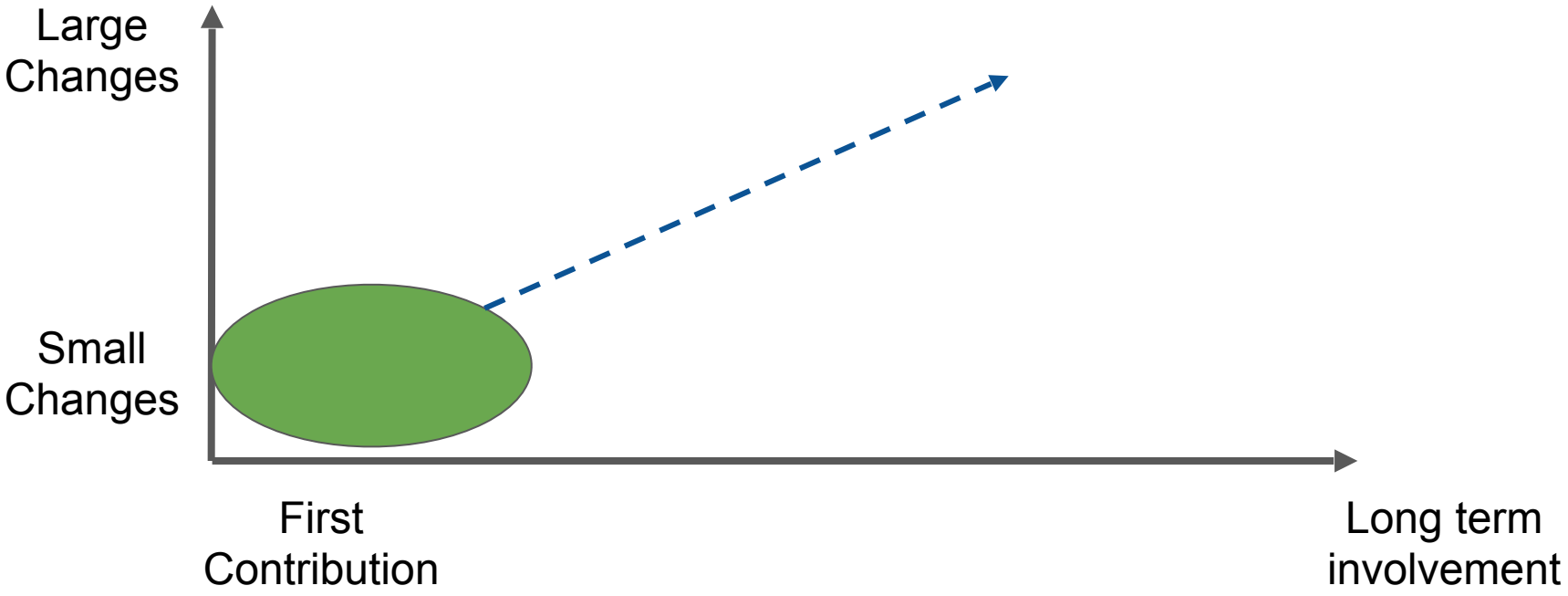
Finding something to work on

“I just need to fix this one bug”

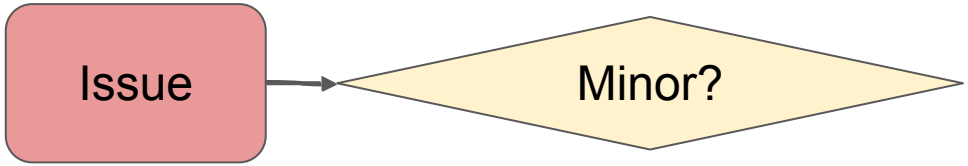
“I just want to contribute”

Finding something to work on

Finding something to work on

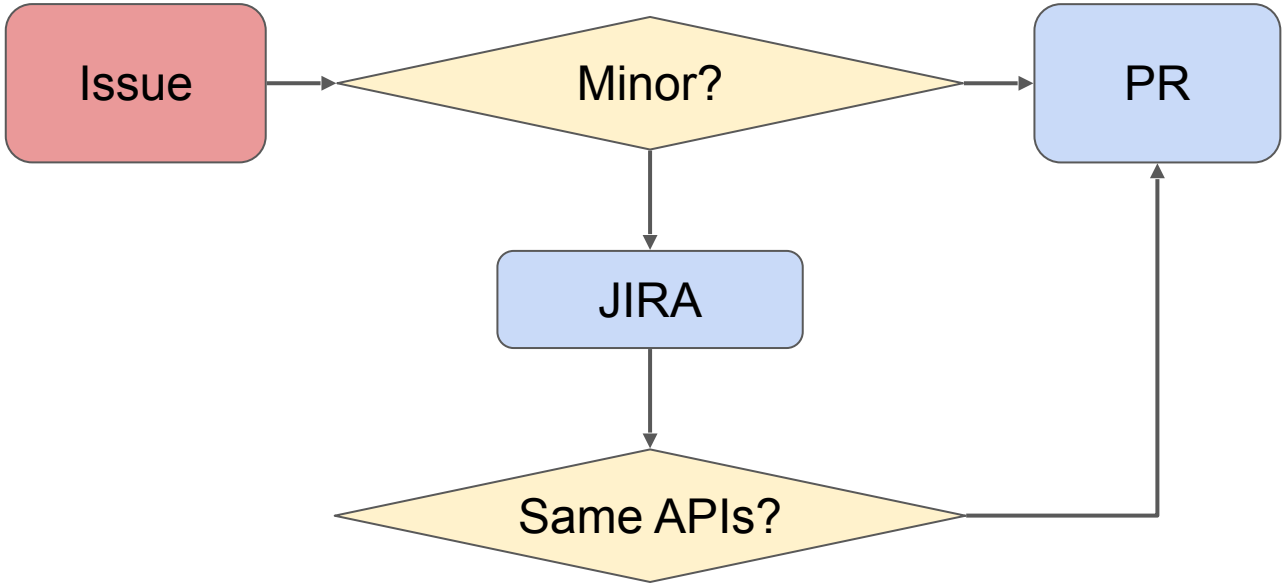


Contributing Code Changes



Contributing Code Changes

Contributing Code Changes



Writing code *and* tests

https://github.com/apache/kafka/blob/trunk/README.md

Gradle Tasks

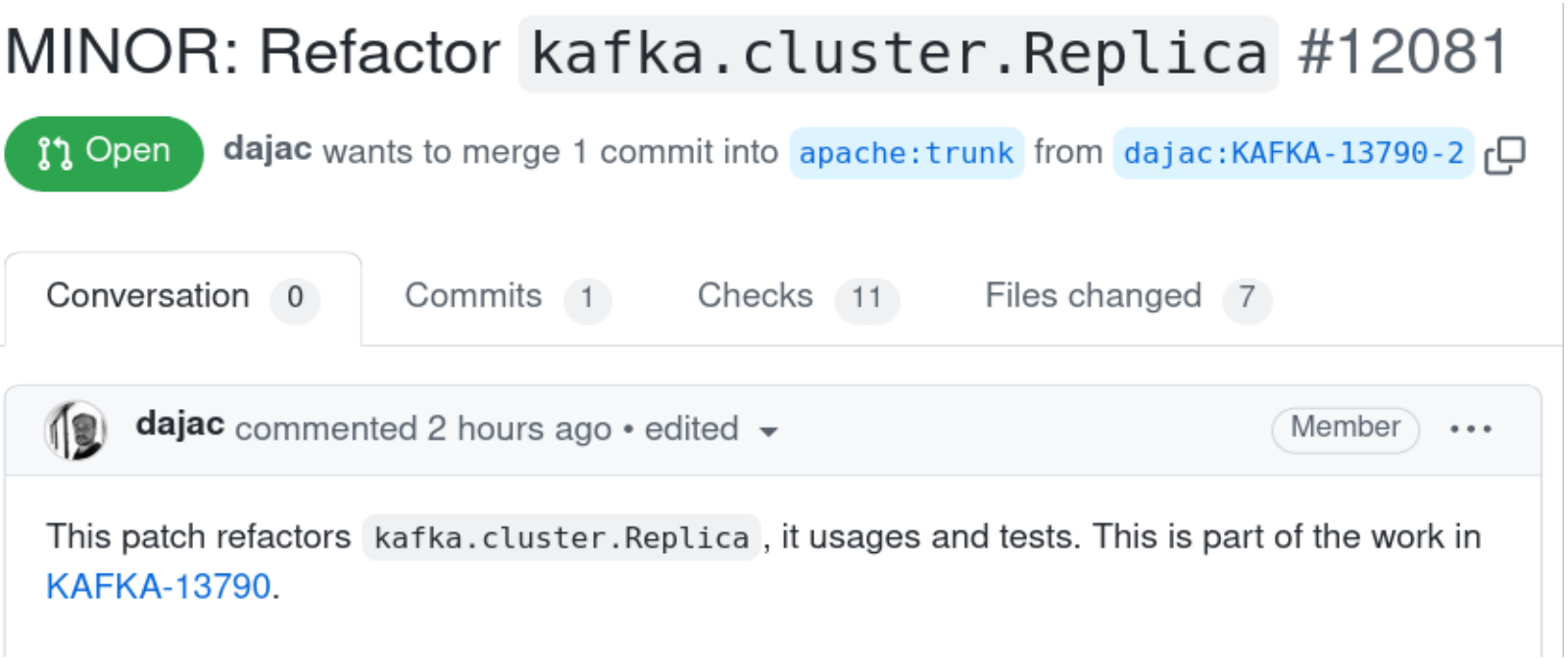
Compile: `assemble`  2 minutes

Build + run test: `build`  20-30 minutes

Run test on specific project: `clients:test`  variable

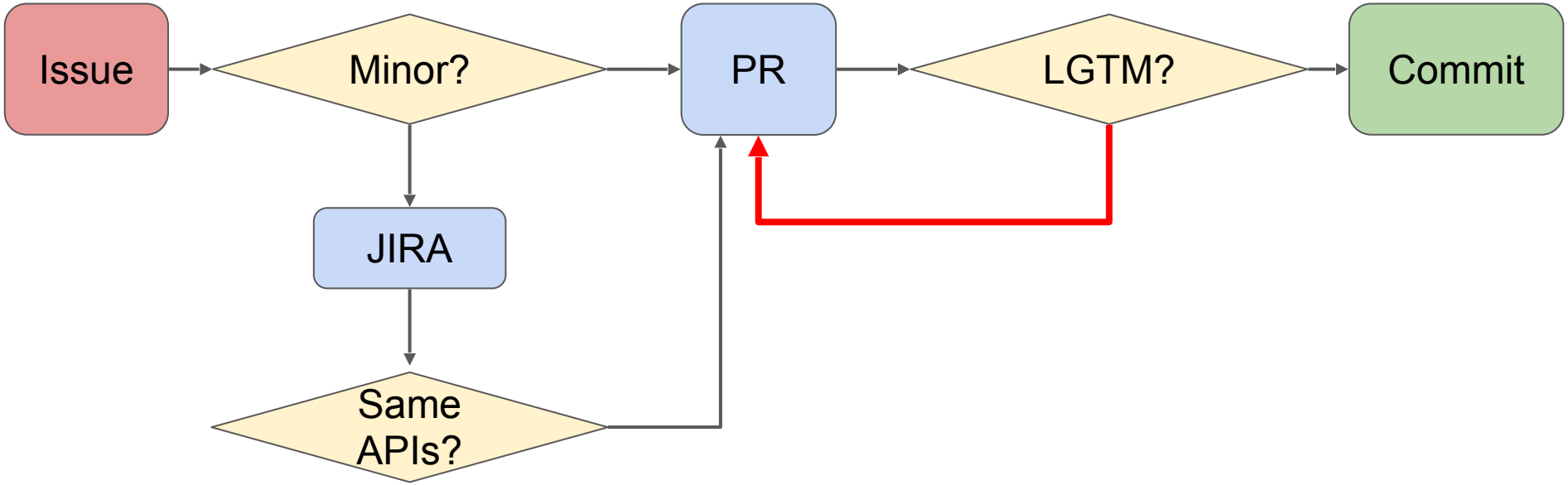
Run specific tests: `core:test --tests "*PartitionTest"`

Opening a Pull Request

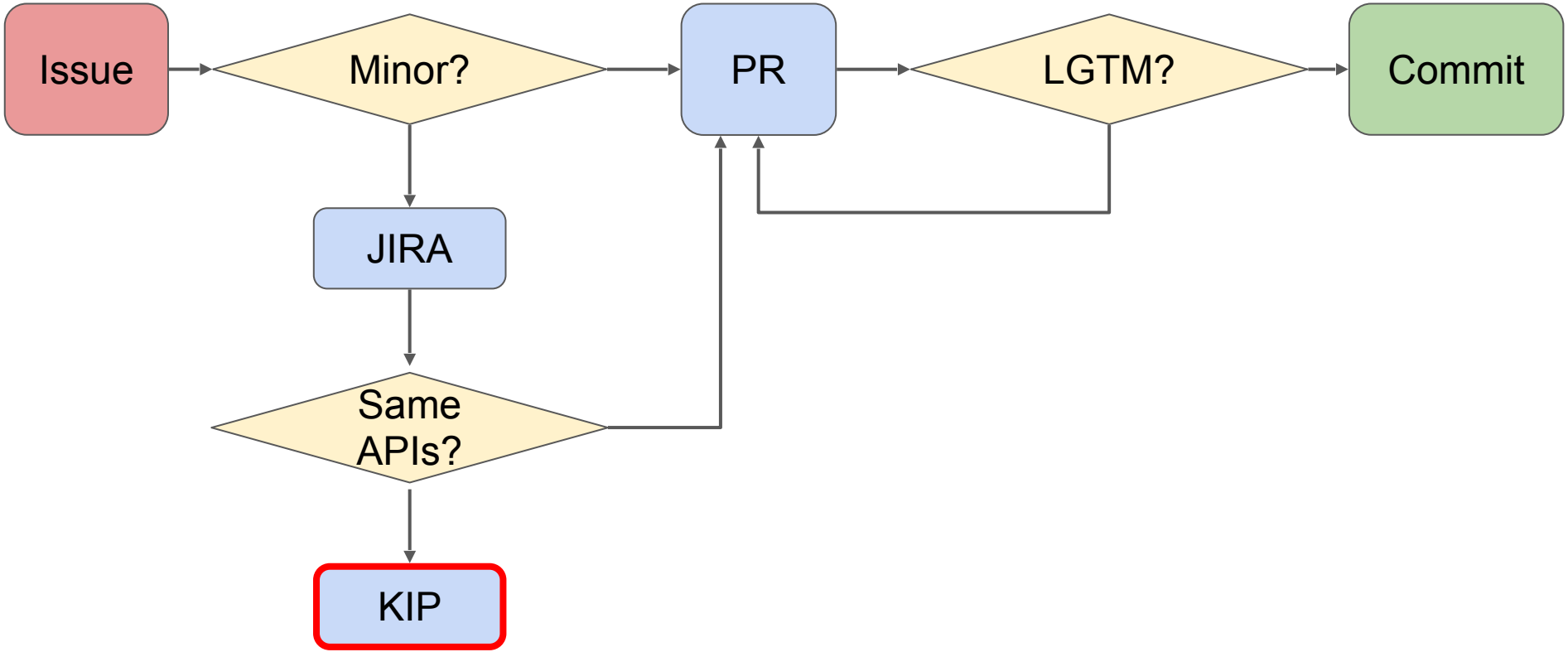


Reviewers: https://kafka.apache.org/committers

Making a Change



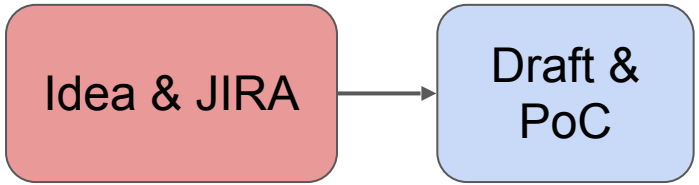
Kafka Improvement Proposals



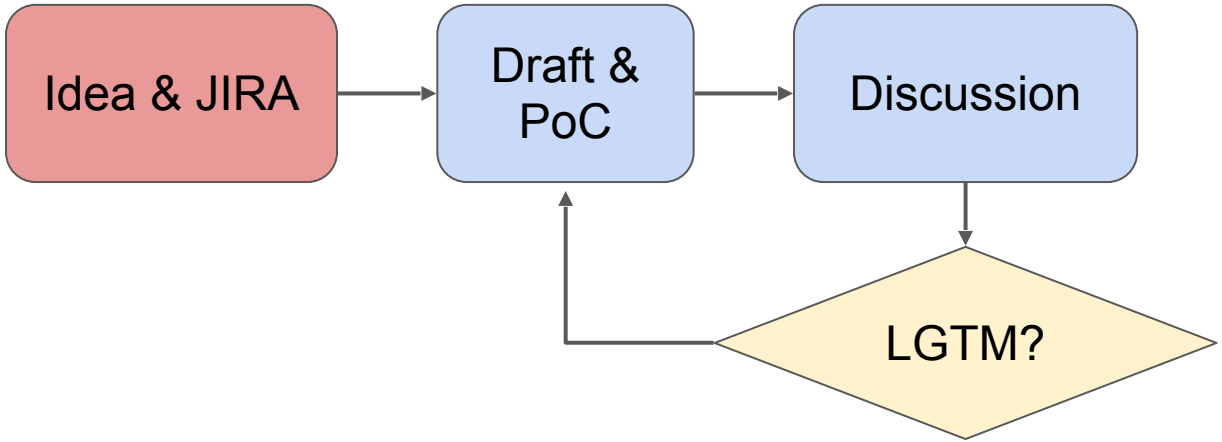

ONE DOES NOT SIMPLY
CHANGE A PUBLIC KAFKA API
imgflip.com

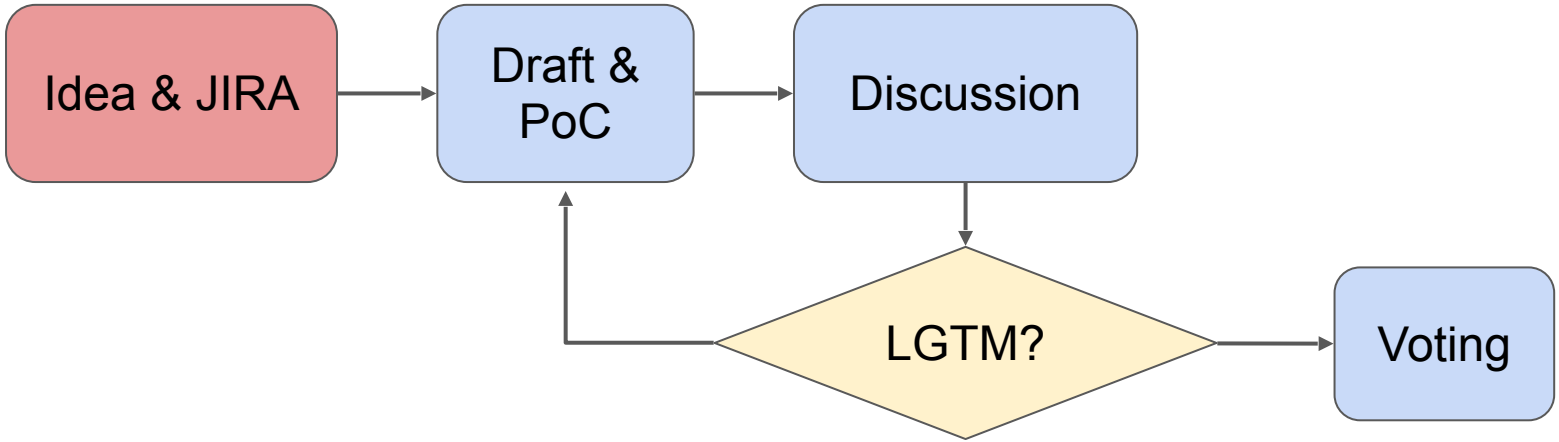

Configs
Interfaces
Public Classes
Network Protocol
Log Format
Behaviour
Metrics
Tools

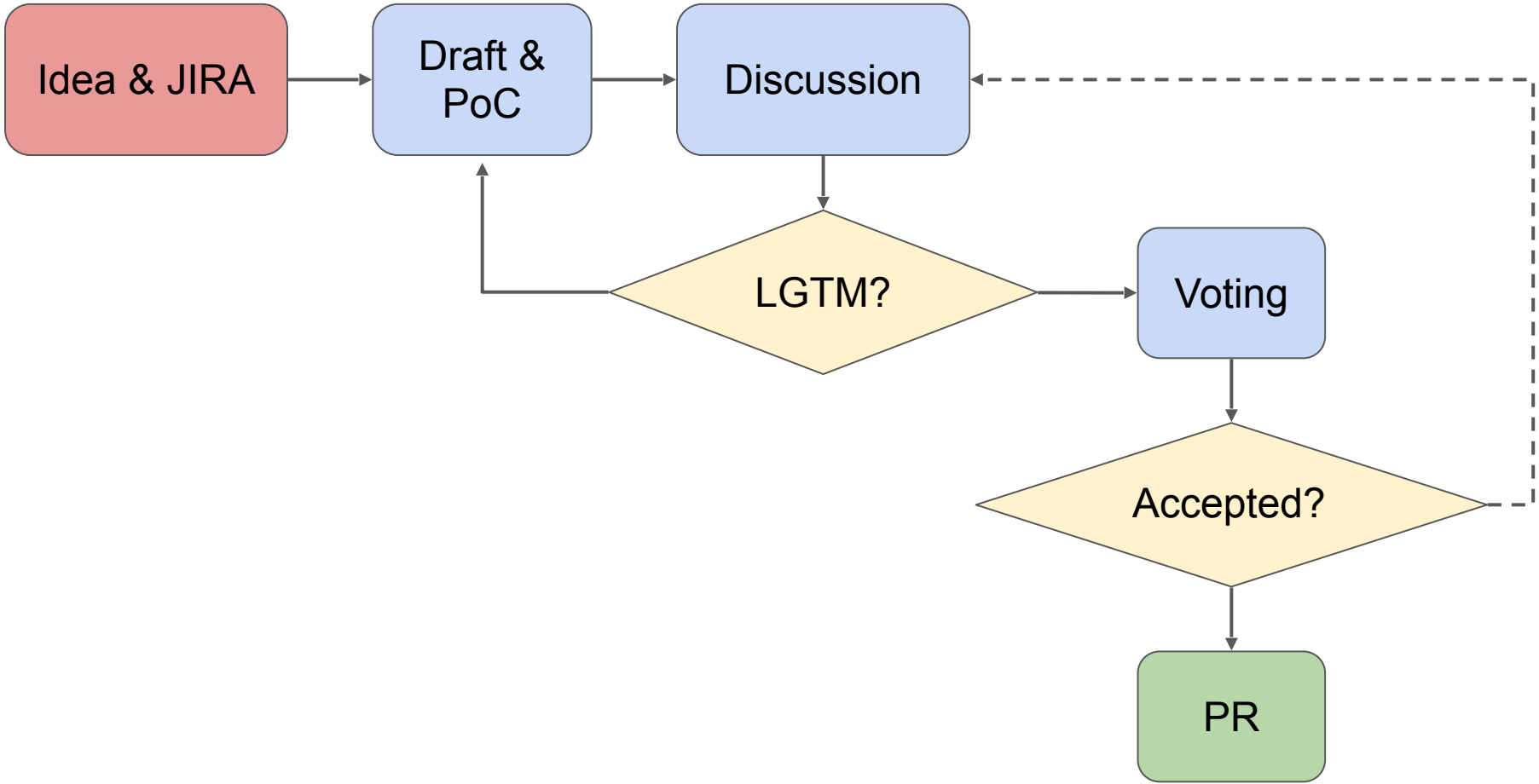
Idea & JIRA


Idea & JIRA
Draft &
PoC


Idea & JIRA
Draft & PoC
Discussion


Idea & JIRA
Draft & PoC
Discussion
LGTM?


Idea & JIRA
Draft & PoC
Discussion
LGTM?
Voting


Idea & JIRA
Draft & PoC
Discussion
LGTM?
Voting
Accepted?
PR

Draft & PoC

Anatomy of a KIP

KIP Index lists all the KIPs, and their current state

Standard sections:

- Status
- Motivation
- Public interfaces
- Proposed changes
- Compatibility, Deprecation, Migration
- Test Plan
- Rejected alternatives

Draft & PoC

Anatomy of a KIP - Status section

Status

Current state: *[One of "Under Discussion", "Accepted", "Rejected"]*

Discussion thread: *here*

JIRA: *here*

Released: <Kafka Version>

Please keep the discussion on the mailing list rather than commenting on the wiki (wiki discussions get unwieldy fast).

Draft & PoC

Anatomy of a KIP - Motivation section

Motivation

Describe the problems you are trying to solve.

Example:
KIP-730: Producer ID generation in KRaft mode

Draft & PoC

Anatomy of a KIP - Public interfaces

Public Interfaces

Briefly list any new interfaces that will be introduced as part of this proposal or any existing interfaces that will be removed or changed. The purpose of this section is to concisely call out the public contract that will come along with this feature.

Example:
KIP-664 Provide tooling to detect and abort hanging transactions

Draft & PoC

Anatomy of a KIP - Proposed changes

Proposed Changes

Describe the new thing you want to do in appropriate detail. This may be fairly extensive and have large subsections of its own. Or it may be a few sentences. Use judgement based on the scope of the change.

Example:
KIP-98: Exactly Once Delivery and Transactional Messaging

Draft & PoC

Anatomy of a KIP - Compatibility, Deprecation, Migration

Compatibility, Deprecation, and Migration Plan

- *What impact (if any) will there be on existing users?*
- *If we are changing behavior how will we phase out the older behavior?*
- *If we need special migration tools, describe them here.*
- *When will we remove the existing behavior?*

Example:
KIP-802 Validation Support for Kafka Connect SMT and Converter Options

Draft & PoC

Anatomy of a KIP - Test plan

Test Plan

Describe in few sentences how the KIP will be tested. We are mostly interested in system tests (since unit-tests are specific to implementation details). How will we know that the implementation works as expected? How will we know nothing broke?

Example:
None! This is new

Draft & PoC

Anatomy of a KIP - Rejected alternatives

Rejected Alternatives

If there are alternative ways of accomplishing the same thing, what were they? The purpose of this section is to motivate why the design is the way it is and not some other way.

Example:
KIP-618: Exactly Once support for Source Connectors

Discussion

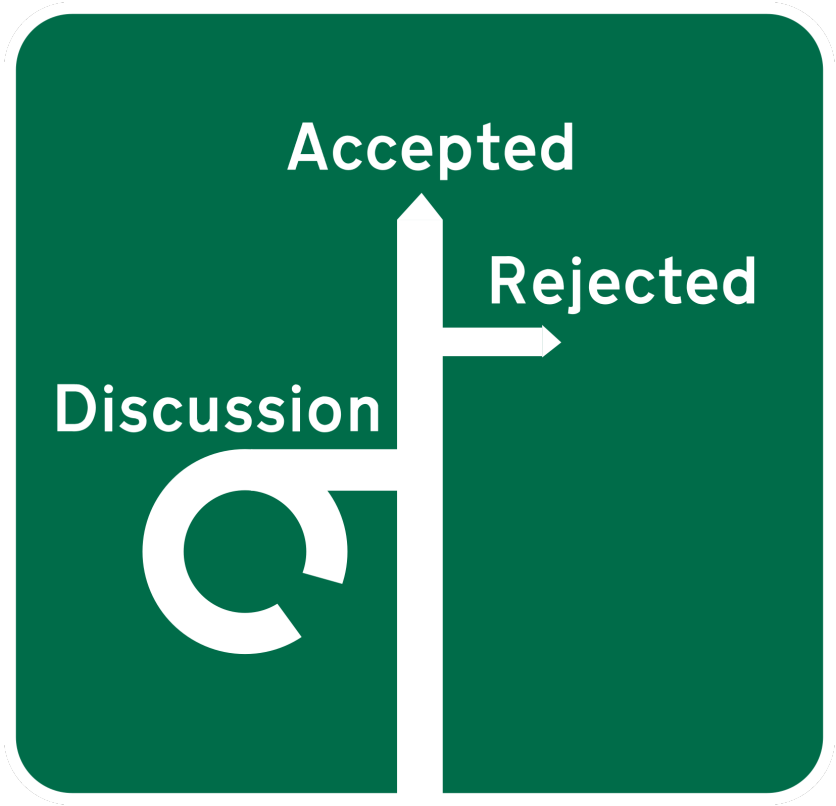
To: dev@kafka.apache.org
Subject: [DISCUSS] KIP-XXX: My great idea

Voting

To: dev@kafka.apache.org
Subject: [VOTE] KIP-XXX: My great idea


IT'S FINALLY OVER
WHEN YOUR PR IS MERGED
imgflip.com

Interminable discussion



Real problem?

Clear + strong motivation?

Too narrow?

Too broad?

Listened to feedback?

Contributing regularly

Thank you!

Key Resources:

https://kafka.apache.org/contributing

https://cwiki.apache.org/confluence/display/KAFKA/Kafka+Improvement+Proposals

https://kafka.apache.org/committers

https://github.com/apache/kafka