



Streamlining community delivery to accelerate community contribution to / adoption of Mojaloop

Sankar Ramakrishnan

Agile Delivery, Product and Management Consultant

Contact: sankar.r.office@gmail.com

Background & Agenda

Background

- In Program Increment (PI) 10, we spotted some gaps and improvement opportunities in the Mojaloop community delivery processes
- Addressing these would foster better business / product Alignment and greater Autonomy for teams.
- We proposed and implemented a number of improvements in PI 11 with the community.
- We worked closely with Community Release Train Engineer and the community towards end of PI-10 and PI-11.
- Left product management items for the new product manager to come on board in the foundation.
- We still have a few more items to do, some of which are product led and some are community led.

Today, would like to showcase:

- the opportunities identified
- the proposals made
- what has been done
- what is needed to be done
- solicit further community participation in taking this further forward / simplifying this

Key Principles: to improve community delivery processes

RULES

- Minimal yet effective
- Promotes Collaboration, Autonomy, Transparency, Alignment
- No discontinuity
- Shared

RAILS

- Constraints
 - Github
 - Zenhub

Propose :: Experiment :: Community feedback :: Pivot/Preserve :: Accelerate adoption of Mojaloop

PI 10: Streamlining opportunities discovered in the community



Opportunities considered:
to improve alignment of delivery with Product
to improve autonomy of community / collaborating teams
to support better quality outcomes

Out of scope:
Opportunities to improve Mojaloop product / product management

PI 12: Where are we today?



- Product roadmap vs. PI plan
- Hypothesis / outcome validation
- Outcomes vs. objectives
 - ✓ Objectives vs. goal
 - ✓ Backlog
 - ✓ Structure
 - ✓ Features (partial)
 - ✓ Epics
 - ✓ Prioritization
- ✓ Minimal & effective Workflows
- ✓ Acceptance Criteria
- ✓ Complexity measures
- ✓ Uncertainty measures
- ✓ Improved Templates
- ✓ Labels
- ✓ Estimates
- ✓ Reports (partial)
- Radiators

Alignment, Harmony: Product, Features, Backlog, Workflow

PI 10
Backlog : Workstream driven

A screenshot of a Jira board interface. At the top, there are several filter buttons: 'Repos (1/33)', 'Labels', 'Milestones', 'Assignees', 'Epics', 'Releases', and a search bar 'Find Issues (f+i)'. Below the filters is a button 'New Issue' and a '+' sign. A large central area is labeled 'Add a Pipeline ...'. To the left, a sidebar lists navigation options: 'Board', 'Reports', 'Roadmap', 'Create...', 'Edit Workspace', and 'Invite'. At the bottom, there's a footer with links: 'View tutorials', 'Shortcuts', 'Open in web app', 'Support and training', and 'Changelog'.



PI 11
Backlog : Product / Team driven

A screenshot of a Jira backlog interface. At the top, there are filter buttons: 'Repos (1/34)', 'Labels', 'Milestones', 'Assignees', 'Epics', 'Releases', and a search bar 'Find Issues (f+i)'. Below the filters is a button 'New Issue' and a '+' sign. The main area is divided into three columns: 'Product (Features)', 'PI (Objectives)', and 'Sprint / Team (Issues: Story, Bug, Spike, Debt)'. The 'Product' column shows a single item: 'Closed 50+ Issues - 256 Story Points'. The 'PI' column shows items: 'Closed 11+ Issues - 45 Story Points', 'Review/QA 0 Issues - 0 Story Points', 'In Progress 1 Issue - 13 Story Points', 'Blocked Stories 4 Issues - 16 Story Points', 'Sprint Backlog 2 Issues - 6 Story Points', and 'PI Backlog 36 Issues - 169 Story Points'. The 'Sprint / Team' column shows items: 'Product Backlog - Epics 6 Issues - 8 Story Points' and 'Blocked Stories 7 Issues - 23 Story Points'. At the bottom, there's a footer with links: 'Add a Pipeline ...' and other navigation icons.

PI 11 Workflow : Minimal / Streamlined

Alignment, Autonomy: Story / Debt / Spike / Bug

PI 10

- Granular product feature alignment
- Linked to very granular epics
- No traceability to PI objectives
- Definition of Done = Acceptance criteria
- No measures for complexity, uncertainty
- Plethora of labels
- Focus on flexibility, nimbleness & not on cross-team / PI-level views



PI 11: Well-aligned & traceable

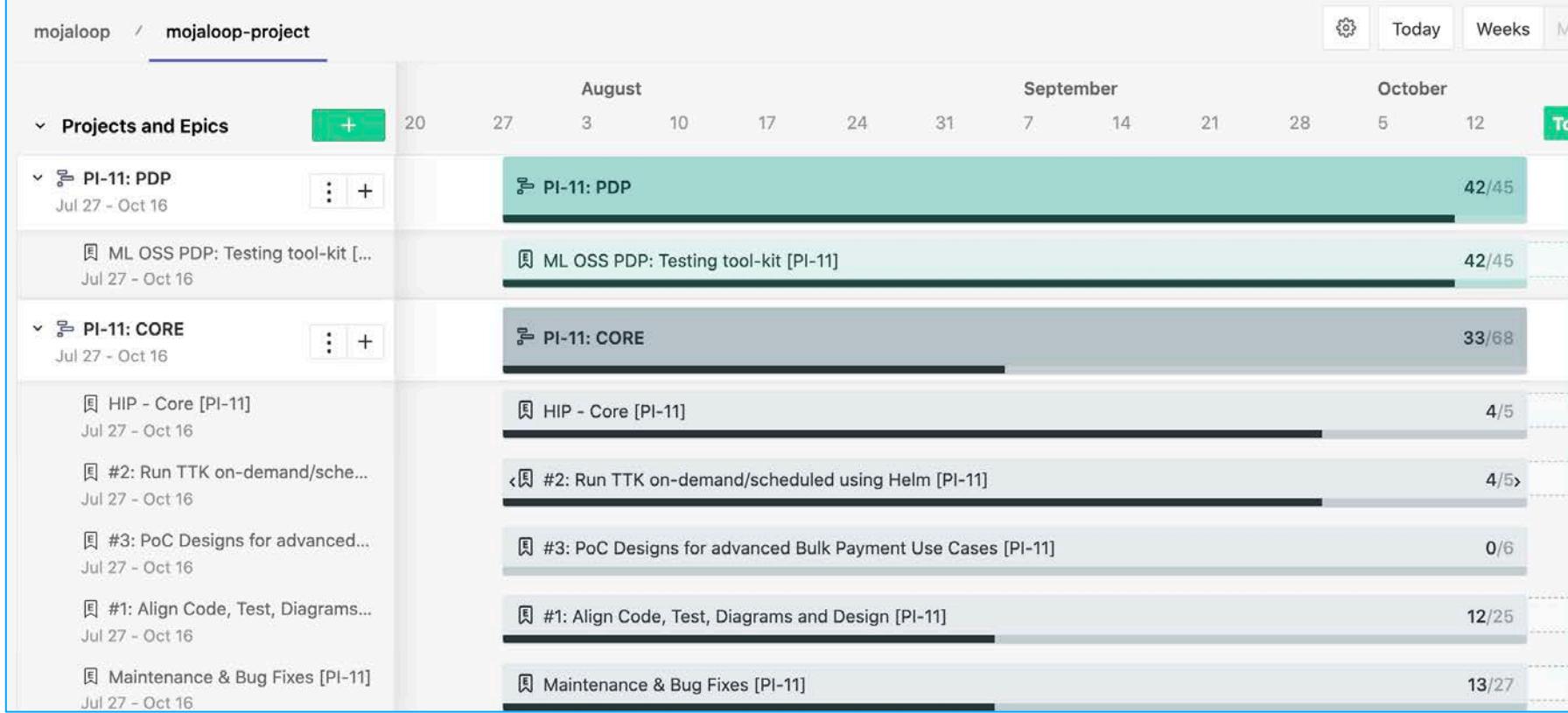
- ✓ Product Features defined as Product Epics
- ✓ PI Objectives defined as PI level epics
- ✓ Stories linked to Product Features & PI objectives
- ✓ Acceptance criteria delineated leveraging BDD style in Story template
- ✓ Measures for complexity, uncertainty introduced in template
- ✓ A core set of labels including workstream identifiers standardized
- ✓ All issue types have clear estimates

Autonomy, Transparency: Reporting

PROGRAM
INCREMENT
(PI)

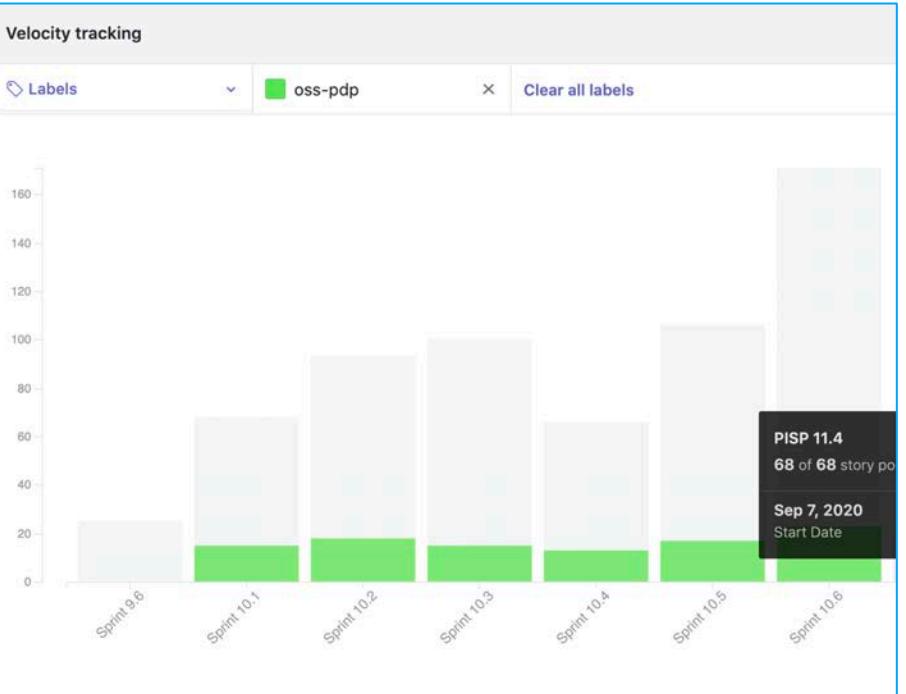
TEAM / SPRINT

Progress against PI objectives

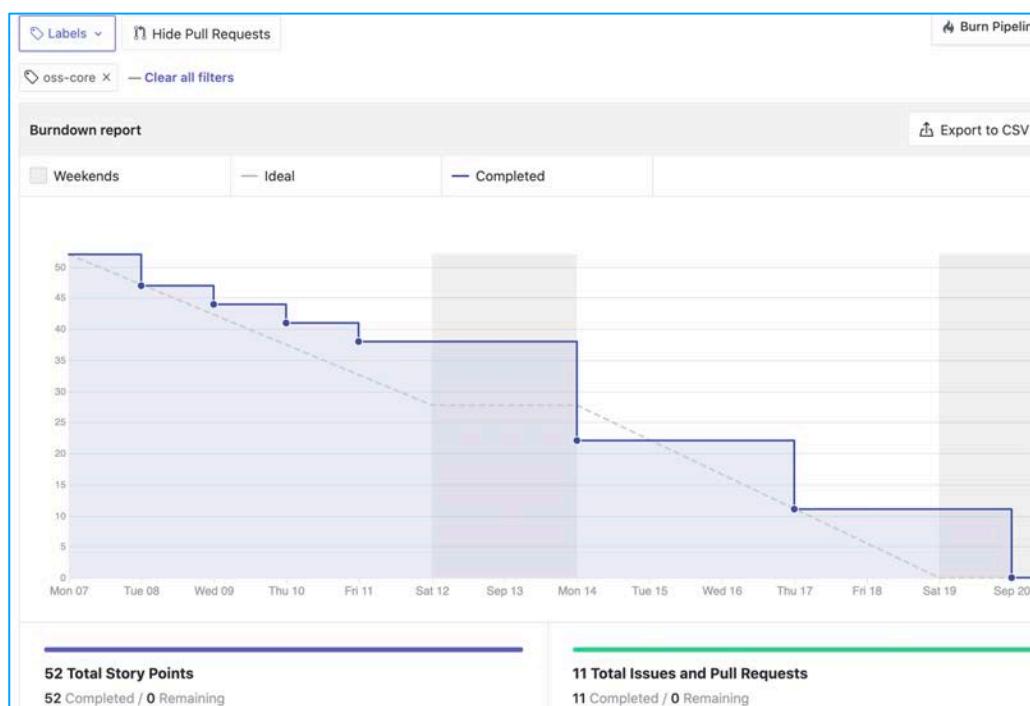


CORE - EXAMPLES

Velocity



Burndown



RELEASE TRAIN

"The changes introduced to align and streamline community delivery processes in PI 11 have immensely helped us in this PI. It really helped that we have all workstreams in Zenhub, with proper tagging and using similar processes which has improved transparency. Sprint stand-ups have become efficient. Especially, in the core team - stories are well written with good acceptance criteria and estimations. A smaller amount of stories were carried over sprint to sprint in the last PI. We matured considerably in the last PI. It is great to see the community adopting these changes robustly".

- Kimberley Walters, Release Train Engineer

Community Showcase

PISP

“The story template improvements in this PI made writing, refining and prioritizing stories much easier. Overall, the PISP team really benefited from the process improvements, which helped us (as a team upon which the sun never sets) work effectively in an asynchronous manner.”

- Lewis Daly, Crosslake
- JJ Geewax, Google

BACKLOG & WORKFLOW

The screenshot shows a Jira board for the 'mojaloop-project' project. The board is organized into four columns: Sprint Backlog, Blocked Stories, In Progress, and Review/QA. The In Progress and Review/QA columns are expanded to show individual issues. Each issue card includes a summary, priority (e.g., High priority), assignee, and labels (e.g., Story, oss-pisp). The Review/QA column also includes a Closed section with completed issues.

Community Showcase

STORY

TTK helm - Include TTK in mojaloop helm chart #1731

 Closed vijayg10 opened this issue 27 days ago · 0 comments



vijayg10 commented 27 days ago · edited

Member



...

Goal:

As a Individual implementer

I want to have TTK deployed as part of the mojaloop helm chart

so that I can use TTK to test the mojaloop deployment easily

Acceptance Criteria:

- ≡ when Mojaloop is deployed using its helm chart in a local machine,
 - then TTK should also be deployed automatically in the same environment as per the Mojaloop helm chart with proper configuration for endpoints, ingress URLs.
 - and We can test the deployed mojaloop instance using the accompanying TTK without further configuration.
- When Mojaloop is deployed using its helm chart and a custom value file in a remote cluster,
 - then TTK should also be deployed automatically in the same environment as per the Mojaloop helm chart with proper configuration for endpoints, ingress URLs.
 - and We can test the deployed mojaloop instance using the accompanying TTK without further configuration.

Complexity: Low

mojaloop
foundation

[Edit](#) [New issue](#)

Pipelines
M mojaloop-project Closed
Assignees
vijayg10
Labels
Story oss-pdp
Projects
None yet
Milestone
Sprint 11.6
Estimate
5

CORE

"The new changes introduced in PI 11 greatly helped me view the flow of stories, bugs through their lifetime and have views based on work-stream's labels which was not so straight-forward earlier", definitely a great improvement!

- Sam Kummery, **Modusbox**

PI 12: Suggested improvements for accelerated adoption of Mojaloop

- The proposed and implemented improvements proved positive, but only a start.
- Further opportunities include:

Alignment

- Hypothesis validation
- Outcome aligned objectives
- Definitive Product Feature list

Autonomy

- Definition of Done
- Release alignment
- Community bug triage

Inviting wider community participation and feedback.



Thank you!

Contact:

Sankar Ramakrishnan

Email: sankar.r.office@gmail.com



APPENDIX

Zenhub limitations / feature:

A story can't have more than 1 milestone

A story can have more than 1 epic

**USER
STORY**

BUG

PI EPIC

In general, every PI Epic will be associated with one Product Feature Epic

Rarely, however, a dummy PI specific EPIC called PI-x-sundry may be created to hold issues that are :

- either not directly related to a product feature
- Or only has a handful of disparate stories from multiple product features

PI OBJECTIVE EPIC

mojaloop
foundation

ML Issue Type and Tagging GUIDELINES

Mandated tags for issues in Sprint Backlog

Epics	<ul style="list-style-type: none"> - Product Feature Epic - PI Epic - PI Objective Epic
Milestone	<ul style="list-style-type: none"> - Sprint Number - Milestone
Labels	<ul style="list-style-type: none"> - Workstream / Squad identifier - Story type identifier (User Story, Technical Story, QA Story, Spike, Enabler, Debt)

Mandated tags for Epics in PI Backlog

Epics	<ul style="list-style-type: none"> - Product Feature Epic - PI Objective Epic - PI Epic (needed only for PI Objective Epic)
Milestone	<ul style="list-style-type: none"> - PI number (tbd, Zenhub constraints)
Labels	<ul style="list-style-type: none"> - Workstream / Squad identifier

ML BACKLOG STRUCTURE DEFINITION

MOJALOOP PRODUCT

Mojaloop = Collection of Mojaloop features

PRODUCT FEATURE EPIC

- defined for every feature; collection of stories that implement the feature

...

PROGRAM INCREMENT 1 (PI 1)

- Delivers product features, epics, stories and bugs to Done

...

PI n

PI OBJECTIVE EPIC

- Captures a PI Objective; A collection of PI Epics

PI EPIC

I will also be highlighting that you and I worked on some of these together at the end of PI10 and PI11.

- fits in one PI; defined for every PI; multiple Epics in PI;

= collection of Mojaloop feature slices (or) collection of stories/bugs

SPRINT 1

- Delivers stories, bugs to Done

...

SPRINT 6

STORY

- Fits in a Sprint; Story Feature (slice) | Enabler | Spike | Debt | QA Story

BUG

Test BUG

PROJECT / WORKSTREAM
= Funds one or more PI (or) one or more EPICS;
Delivered by 1 or more Squads

Label: Squads - 'oss-core', 'oss-pdp', 'pisp'