mojaloop

Design principles of the Mojaloop platform

Michael Richards
James Bush

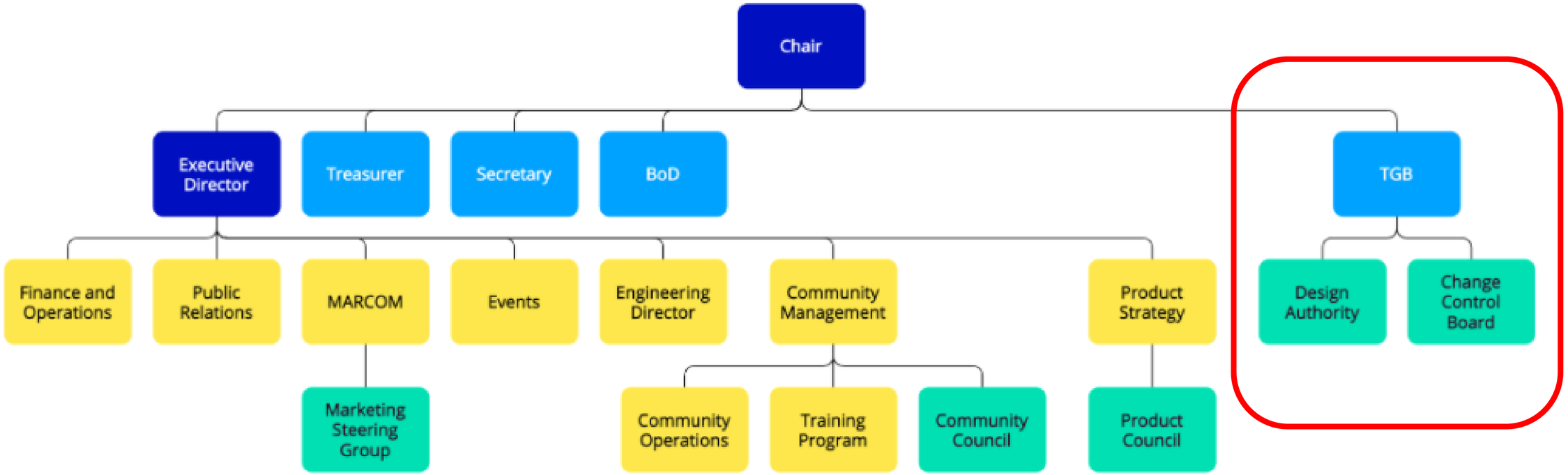
mojaloop

Mojaloop objectives:

Attract and/or grow new constituencies of contributors

- This is about making it easier for people to make technical contributions

Mojaloop foundation: technical organisation



The Technical Governing Board

- The Technical Governing Board plans and oversees all the technical work conducted within the organization
- It ensures a diversity of perspectives and approaches regarding the technical work at hand
- It is empowered to:
  - Make all decisions regarding technical direction;
  - Approve technical program activities that fall within the scope of the mission, such as:
    - Code release decisions
    - Standards document approvals
    - Authorisation of code maintainers

The Change Control Board

- Has overall responsibility for the APIs which provide access to the platform for:
  - DFSPs
  - PISPs
  - Hub administrators
  - Hub operators
- Co-ordinates and oversees the work of individual Special Interest Groups which work on the individual APIs in the Mojaloop family.
- Ensures that the APIs in the Mojaloop family have a family resemblance

The Design Authority

- Ensures a uniform architecture
- Forms a channel for bringing ideas to design and verification
- Defines technical strategies
- Verifies that designs are aligned with architectural standards
- Drives, publicises and promulgates preferred design methodologies
- Provides resources to advise on productive design in the Mojaloop context.

So the question is:

- How can we help new members of the technical community to contribute as quickly and productively as possible?

So the question is:

- How can we help new members of the technical community to contribute as quickly and productively as possible?
  - ➢People who want to integrate with a Mojaloop platform

So the question is:

- How can we help new members of the technical community to contribute as quickly and productively as possible?
  - ➢People who want to integrate with a Mojaloop platform
  - ➢People who want to start a Mojaloop platform

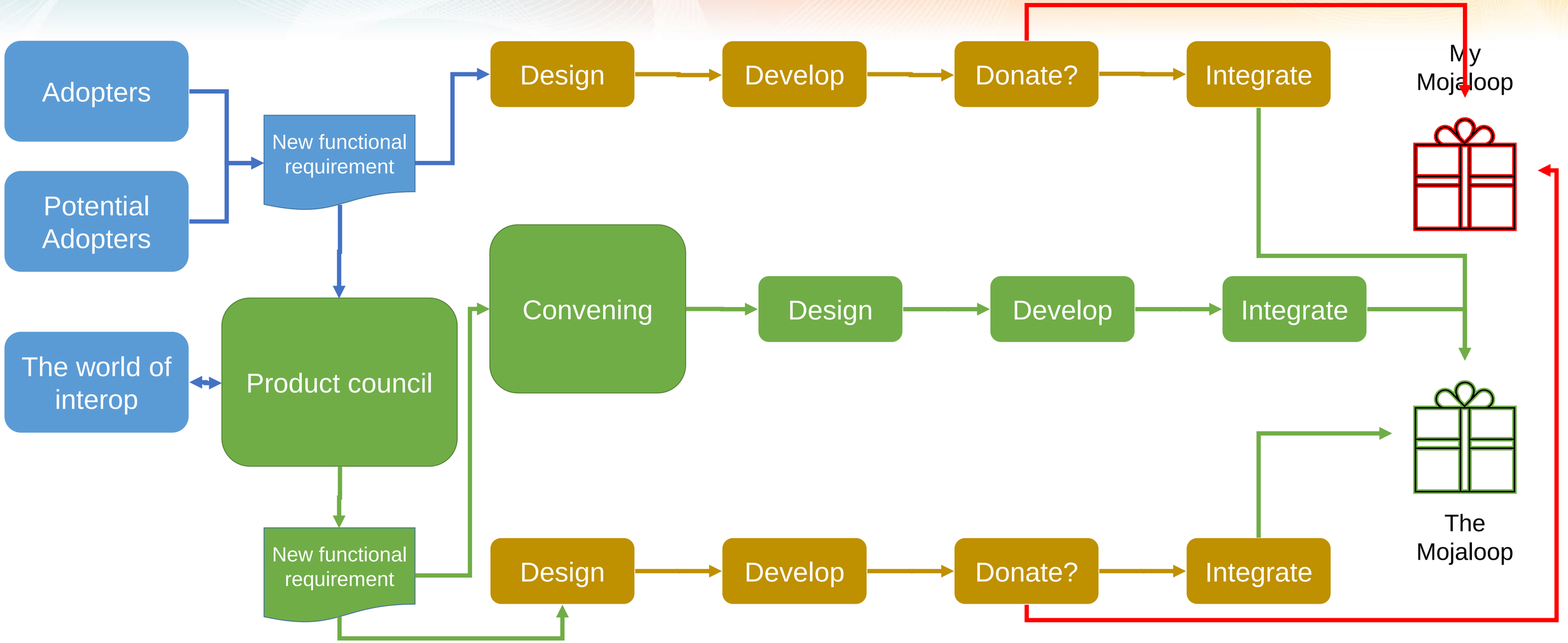
So the question is:

- How can we help new members of the technical community to contribute as quickly and productively as possible?
  - ➢People who want to integrate with a Mojaloop platform
  - ➢People who want to start a Mojaloop platform
  - ➢People who want to add new functional areas to the Mojaloop platform

So the question is:

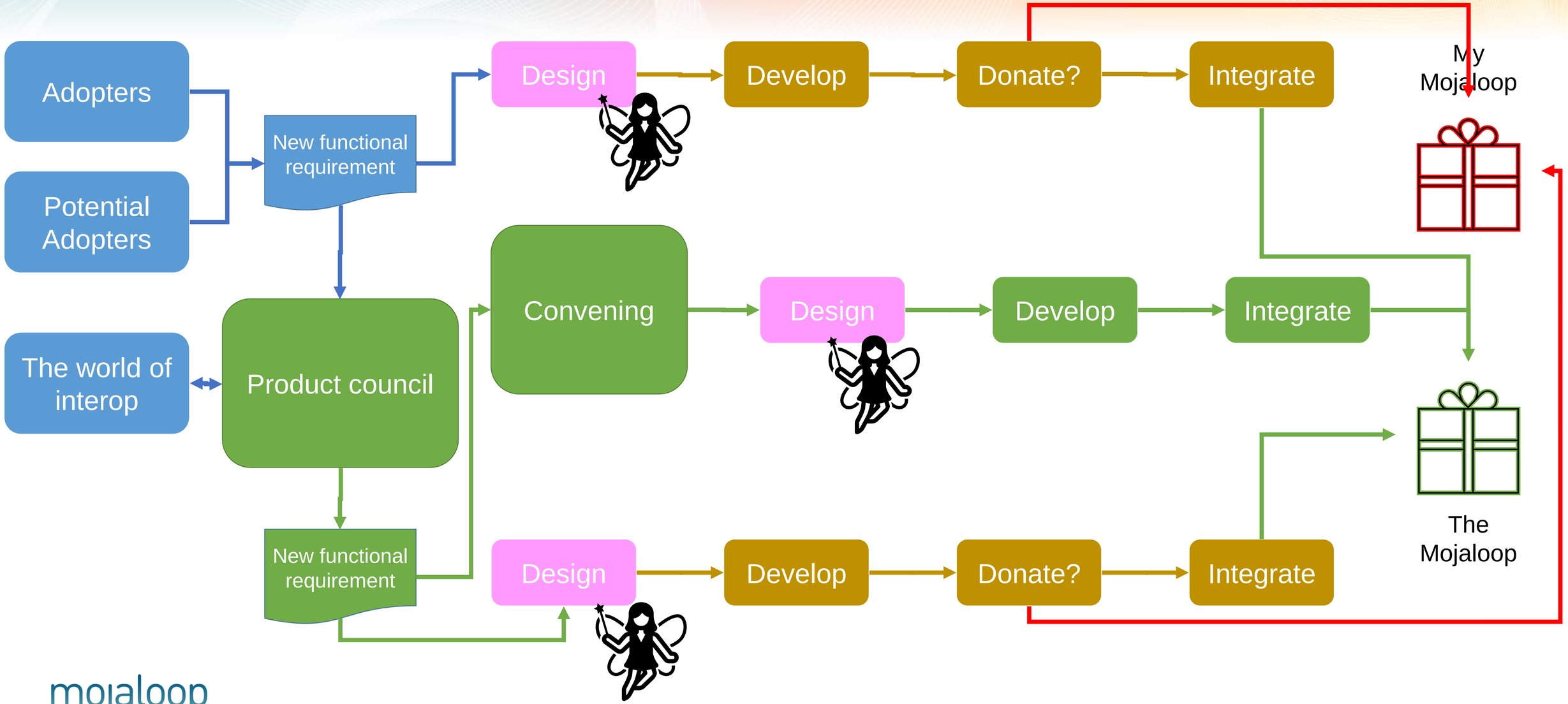
- How can we help new members of the technical community to contribute as quickly and productively as possible?
  - People who want to integrate with a Mojaloop platform
  - People who want to start a Mojaloop platform
  - People who want to add new functional areas to the Mojaloop platform
  - People who want to extend or enhance the existing functionality of the Mojaloop platform

The Mojaloop evolution model



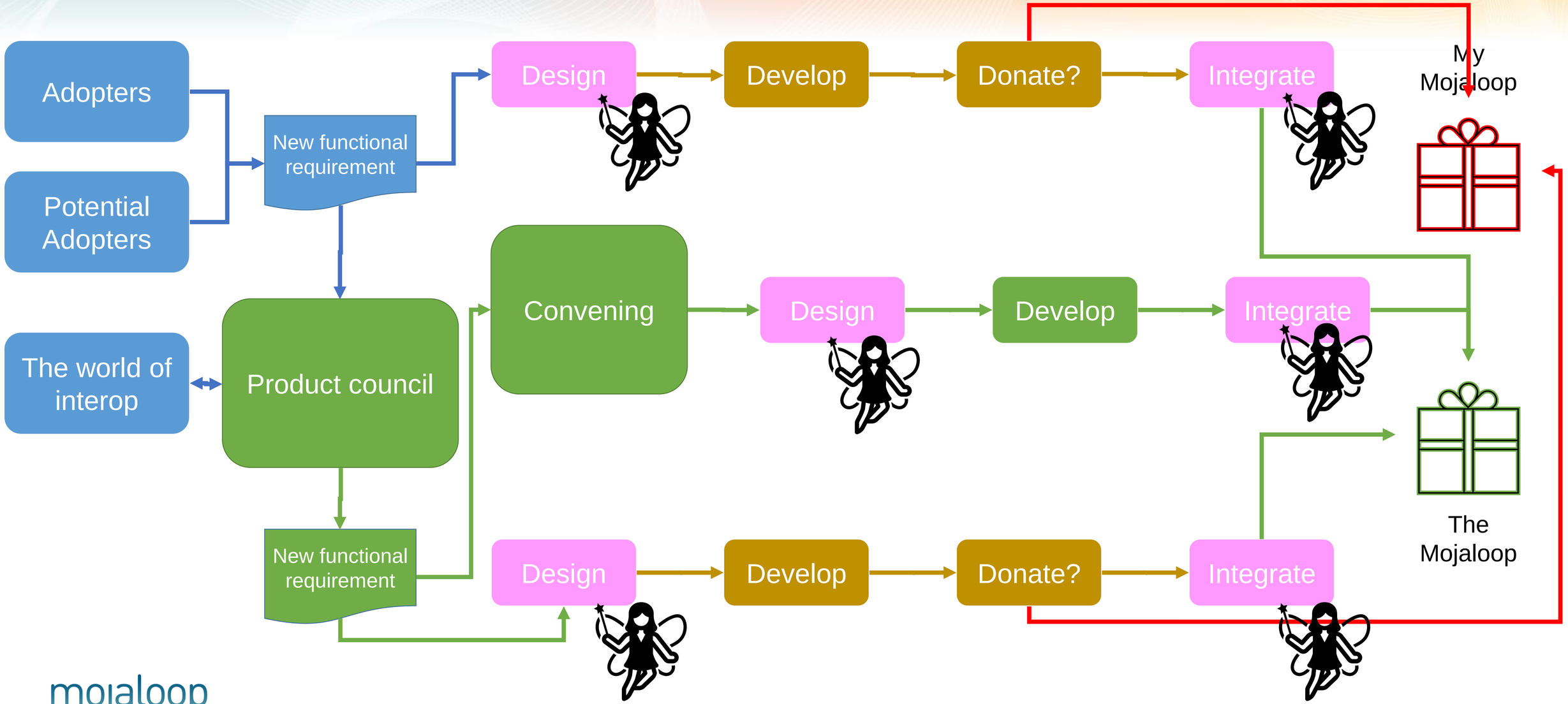
mojaloop

Where does the DA fit in?



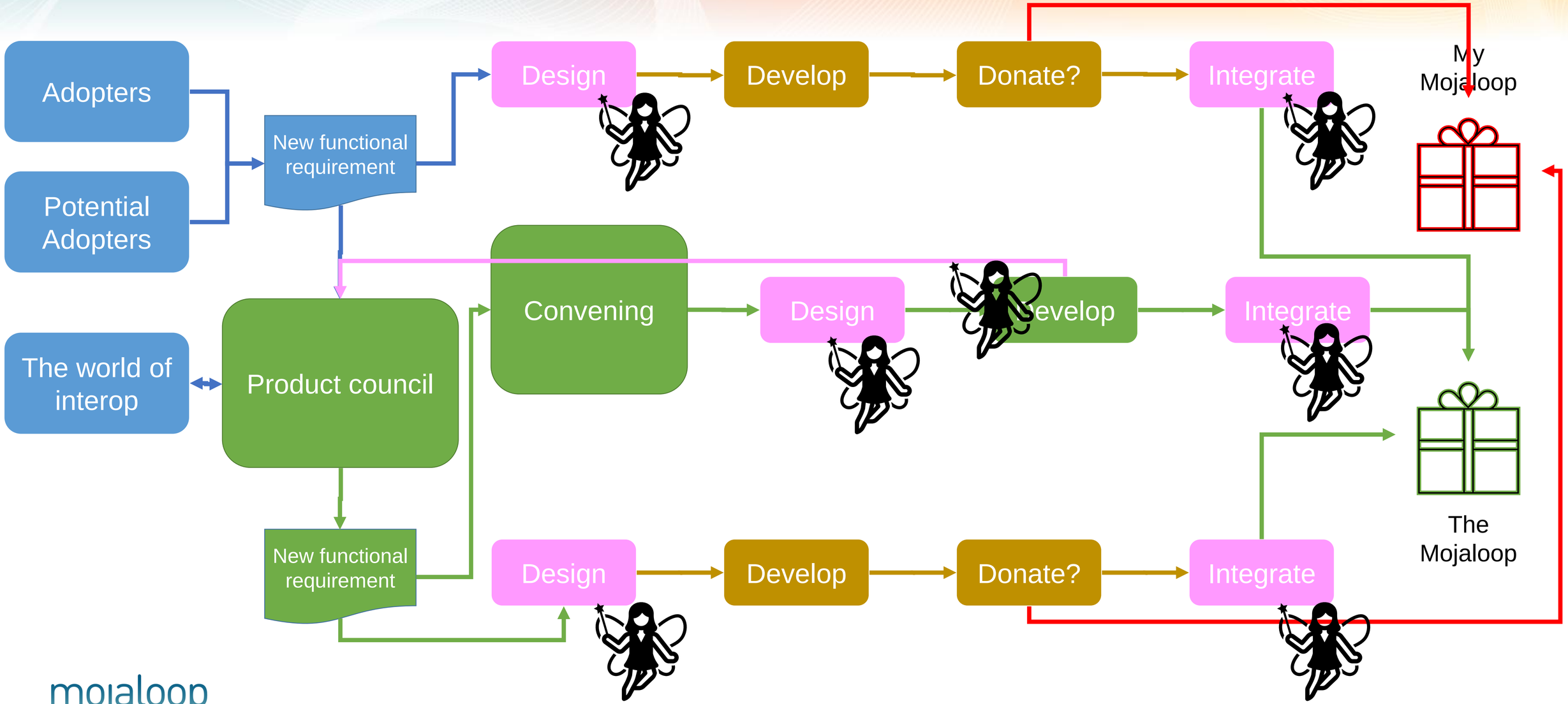
mojaloop

Where does the DA fit in?



mojaloop

Of course, this is an agile world…



mojaloop

Getting up to speed…

- There are training courses…
- We discuss the current and planned functionality of the platform at the convenings…
- Anyone can join the *#design-authority* channel on the Mojaloop Slack workspace…
- You can view or raise issues on the DA's GitHub page: https://github.com/mojaloop/design-authority/issues
- There's a DA Community forum page: https://community.mojaloop.io/c/governance-discussions/da/13

Getting up to speed…

- There are training courses…
- We discuss the current and planned functionality of the platform at the convenings…
- Anyone can join the *#design-authority* channel on the Mojaloop Slack workspace…
- You can view or raise issues on the DA's GitHub page: https://github.com/mojaloop/design-authority/issues
- There's a DA Community forum page: https://community.mojaloop.io/c/governance-discussions/da/13
- *And we're preparing a document describing the Design Principles of the Mojaloop platform*

Design principles of the Mojaloop platform

Functions of the Design Principles document

- Originates from the Technical Governing Board

Functions of the Design Principles document

- Originates from the Technical Governing Board
- Sets the design parameters within which the DA operates

Functions of the Design Principles document

- Originates from the Technical Governing Board
- Sets the design parameters within which the DA operates
  - Objectives

Functions of the Design Principles document

- Originates from the Technical Governing Board
- Sets the design parameters within which the DA operates
  - Objectives
  - Constraints

Functions of the Design Principles document

- Originates from the Technical Governing Board
- Sets the design parameters within which the DA operates
  - Objectives
  - Constraints
  - Requirements

Functions of the Design Principles document

- Originates from the Technical Governing Board
- Sets the design parameters within which the DA operates
  - Objectives
  - Constraints
  - Requirements
  - Preferences

Functions of the Design Principles document

- Originates from the Technical Governing Board
- Sets the design parameters within which the DA operates
- Provides a framework which the DA uses to evaluate proposed solutions

Functions of the Design Principles document

- Originates from the Technical Governing Board
- Sets the design parameters within which the DA operates
- Provides a framework which the DA uses to evaluate proposed solutions
- Allows new technical contributors to Mojaloop to understand how to contribute productively to the core product

Overall objectives

1. Consistent with the Level One Principles

The Level One Principles

1. It should be an *open-loop system*, which allows DFSPs of different kinds and sizes to interact with each other independently of their type.
2. It should use *open, international standards*, which can be understood and adopted by all.
3. Payments should be credited to their recipients directly and in *real time*.
4. Payments should be *irrevocable*: once a payment has been credited to a customer's account, it may not be removed from that customer's account without the explicit consent of the customer.
5. The platform should support at least *same-day settlement*, and preferably settlement intervals which are shorter than that.
6. The platform should be operated by a scheme which has *pro-poor governance practices*; in particular, it should ensure that all participants are engaged in the structure and management of the scheme.
7. The platform should be operated by a scheme which is *supported and regulated by a government financial authority*, to ensure that the rights and interests of its users are properly protected by the system.
8. The platform should *leverage tiered Know Your Customer (KYC)* structures to enable participants to adhere to regulatory requirements and to avoid fraudulent use of the system.
9. The system should be operated on a *not-for-loss or cost-recovery-plus-investment basis*. Its objectives are to support the transfer of funds as a utility, and to take into account the needs of women, the poor and other disadvantaged groups.
10. The platform should support a shared investment in scheme and platform services (for example, *fraud and risk management*) where collaboration between participants can give an advantage to all.

Overall objectives

1. Consistent with the Level One Principles

But beyond that…

2. Security
3. Scalability

Security

- Implementable over the public internet
- Open loop
- Real time payments
- Two-phase commit

Scalability

- Cheap to start
- Can cope with success
- Via a defined roadmap…

Scalability

- Cheap to start
  - Does not rely on particular environments or infrastructures
  - Breaks down into individual services.
  - Services are self-contained and can be implemented separately from each other.
  - Each service uses no more resources than it needs.
  - Resources are shared wherever advantageous (*and we always have a eye to our advantage*)
  - Follows the dictum of Einstein: *everything should be made as simple as possible, but not simpler*.
  - Minimises the working capital required from participants.
- Can cope with success
- Via a defined roadmap…

Definitions of a micro-service architecture

- Components of the platform should be independently deployable.
- Components should run as separate services. That is to say:
  - They should run as independent processes and should be deployable independently of each other.
  - It should be possible independently to create multiple instances of a service where this is required to meet performance demands.
  - Services which are using another service should access it through an interface which does not require the user to know the location details of the service they are using, nor the number of instances of the service deployed.
  - The system should efficiently manage the allocation of work among multiple instances of the same service. This management should be transparent both to users of the service and to instances of the service.
- Components should communicate with each other using standard protocols and documented, well defined interfaces.
- Each component should be responsible for persisting its own state. Persisted data should not be available to other services except through the defined interfaces.

Scalability

- Cheap to start
- Can cope with success
  - Supports straight through processing.
  - Codes for success.
  - Idempotent processing.
  - Deterministic outcomes.
  - Components are properly instrumented to support performance analysis.
- Via a defined roadmap…

Next steps

- We will publish the Design Principles document on the foundation site
- We will accept comments and suggestions on clarity or readability
- We will not accept comments suggesting alterations to content
- We will publish a final version of the Design Principles document
- We will work with designers and developers to ensure that their development engagement with the Mojaloop platform is efficient and productive
- We'll report back next time

Thank you