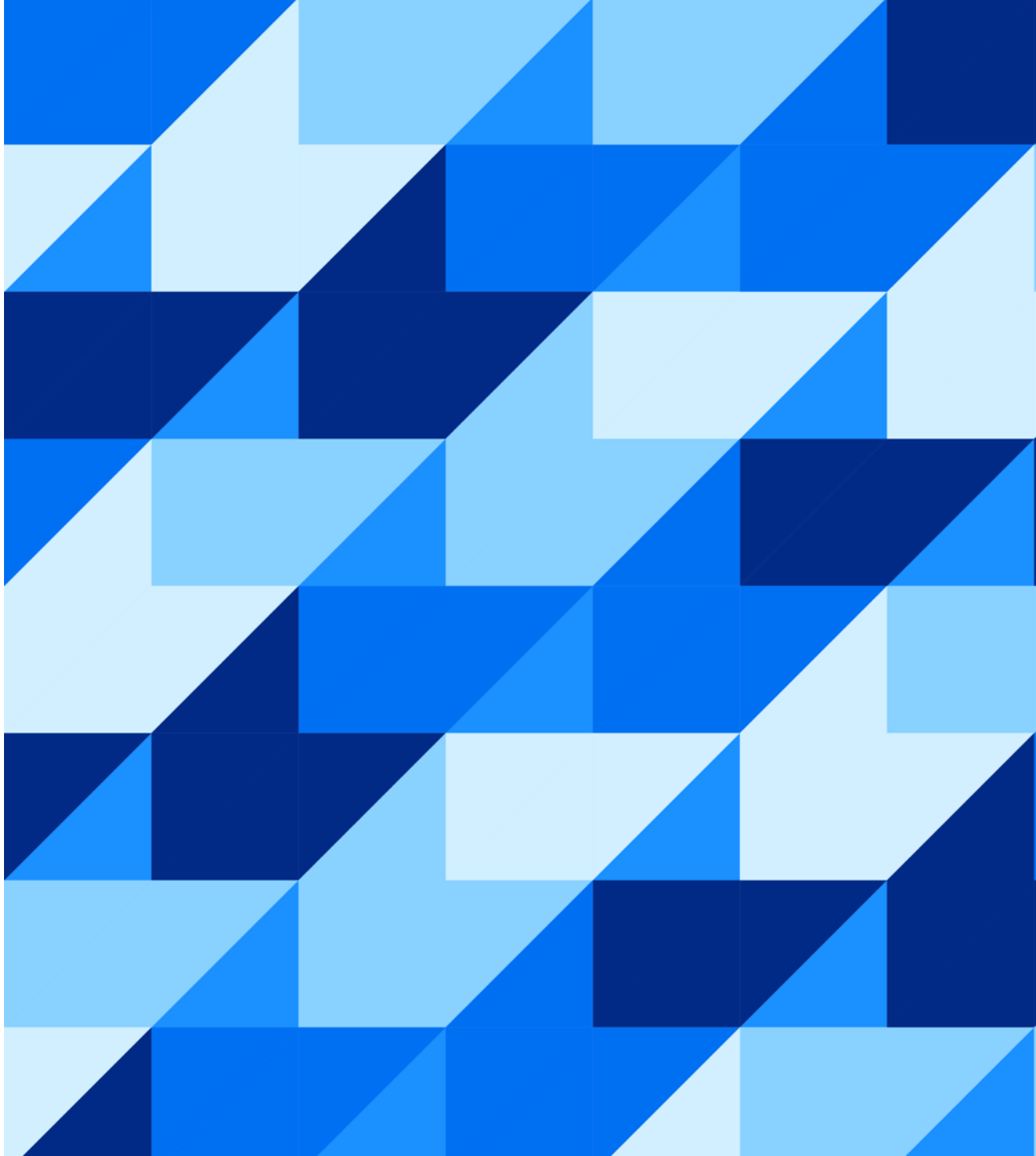


Platform Engineering XXL

oder:
Wie wir eine CI/CD Platform für 30.000+ Entwickler:innen bauen und betreiben

Dirk Lehmann (he/him), SAP
June 19th 2024

Public

techcamp HAMBURG

About Me

- With SAP since 2001
- Implemented the first daily delivery process at SAP (2014)
- Current job: Solution Manager for internal development platform
- SAP Contact person for german language user group (DSAG) for the topic “DevOps”
- Co-Organizer DevOpsDays Zurich
  https://devopsdays.ch
- Conference speaker
  https://doergn.github.io/
- 1 Wife, 2 Kids
- @doergn@mastodon.social

SAP Facts

Founded 1972

Enterprise software solutions
- Tailored for 26 industries
- Offered in 180 countries (in 30+ languages)

107.602 employees worldwide (08.02.2024)
- 36.444 (33.9%) employees in Research and Development (31.12.2023)

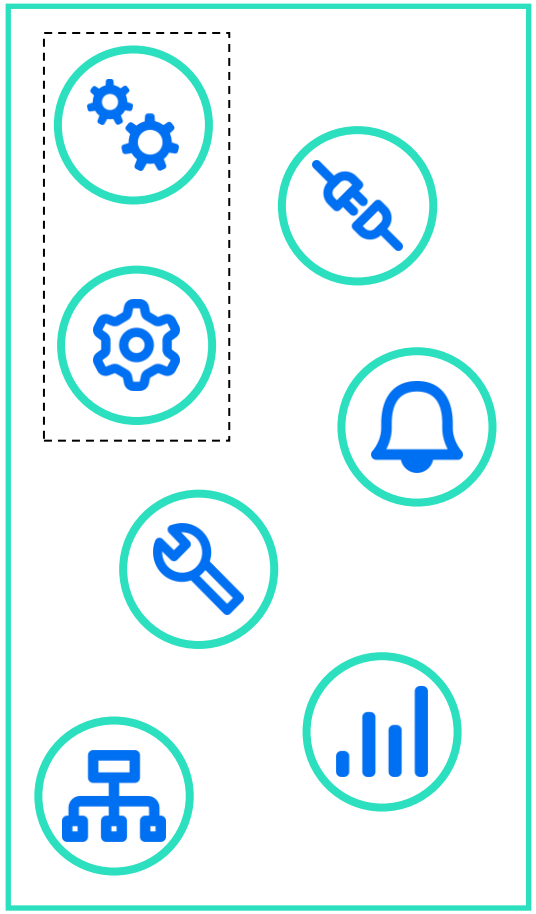
Programming languages
- ABAP (Advanced Business Application Programming)
  - Own proprietary language and technology stack in (mainly) core ERP suite
- Non-ABAP
  - Java ~30%
  - JavaScript ~15%

Delivery channels
- On premise, Cloud, Hybrid, Mobile
- Multi-Cloud Strategy (Hyperscaler + own 52 Data Centers)


```
CLASS zbp_generate_bookings_xxx DEFINITION
  PUBLIC
  FINAL
  CREATE PUBLIC .

  PUBLIC SECTION.
    INTERFACES if_oo_adt_classrun.
  PROTECTED SECTION.
  PRIVATE SECTION.
ENDCLASS.


CLASS zbp_generate_bookings_xxx IMPLEMENTATION.

  METHOD if_oo_adt_classrun~main.
    DATA:it_bookings TYPE TABLE OF ztbooking_xxx.

*    read current timestamp
    GET TIME STAMP FIELD DATA(zv_tsl).
*   fill internal table (itab)
    it_bookings = VALUE #(
        ( booking  = '1' customername = 'Buchholm' numberofp
          country = 'Germany' dateofbooking ='20180213125959
        ( booking  = '2' customername = 'Jeremias' numberofp
          country = 'USA' dateofbooking ='20180313125959' da
     ).

*   Delete the possible entries in the database table - in c
    DELETE FROM ztbooking_xxx.
*   insert the new table entries
    INSERT ztbooking_xxx FROM TABLE @it_bookings.

*   check the result
    SELECT * FROM ztbooking_xxx INTO TABLE @it_bookings.
    out->write( sy-dbcnt ).
    out->write( 'data inserted successfully!' ).

  ENDMETHOD.

ENDCLASS.
```

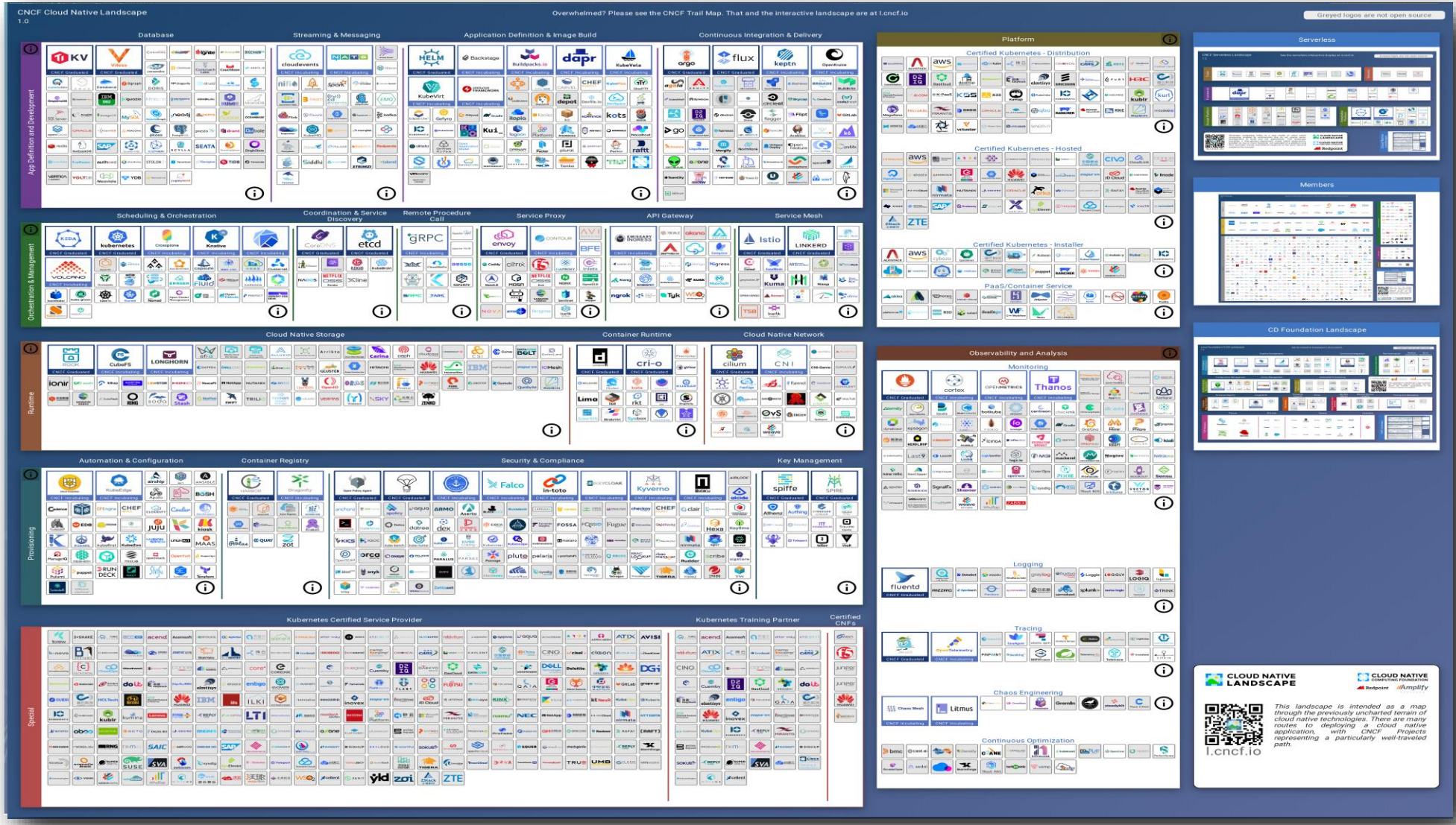
The Situation of a Development Team

Development Team



Tools

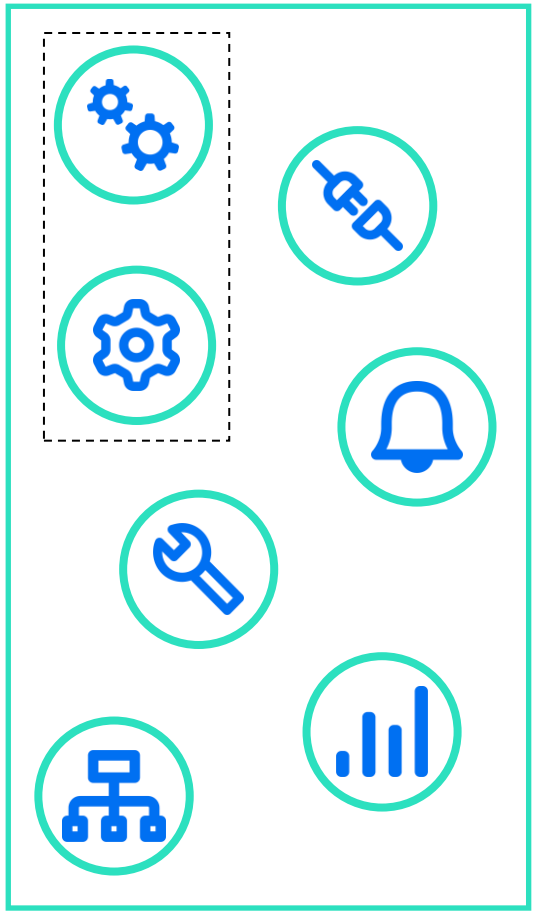
Normal Tools. Just Innocent Tools.



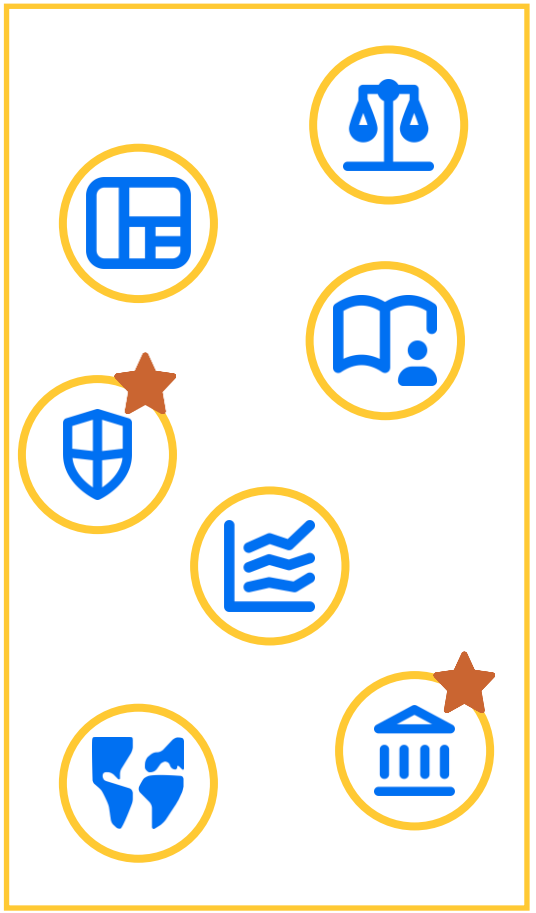
Source: Cloud Native Computing Foundation, CNCF Landscape, Oct. 2020

The Situation of a Development Team

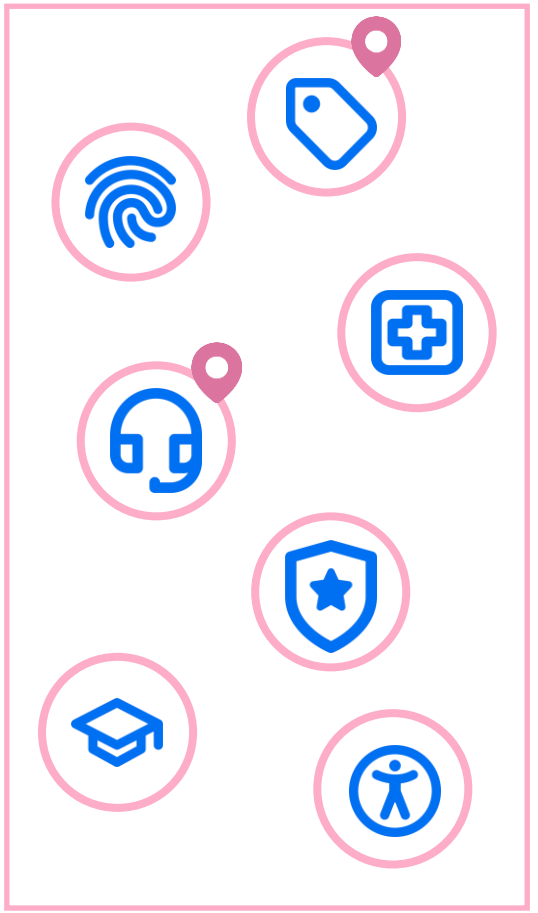
Development Team



Tools



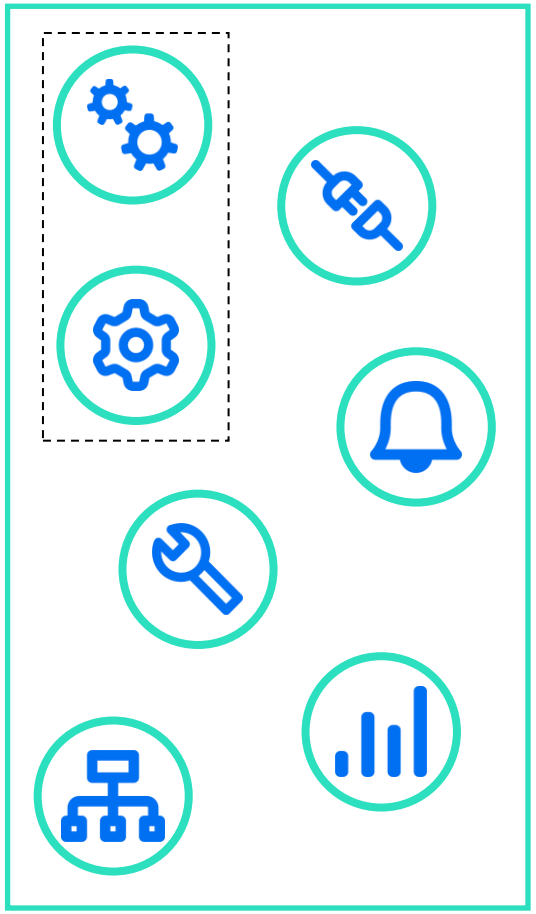
Compliance



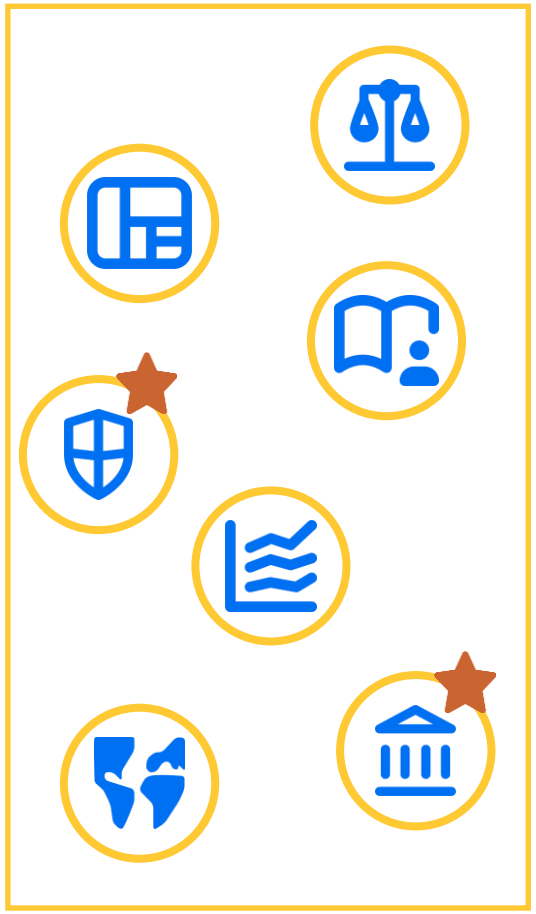
Guidelines

The Situation of a Development Team

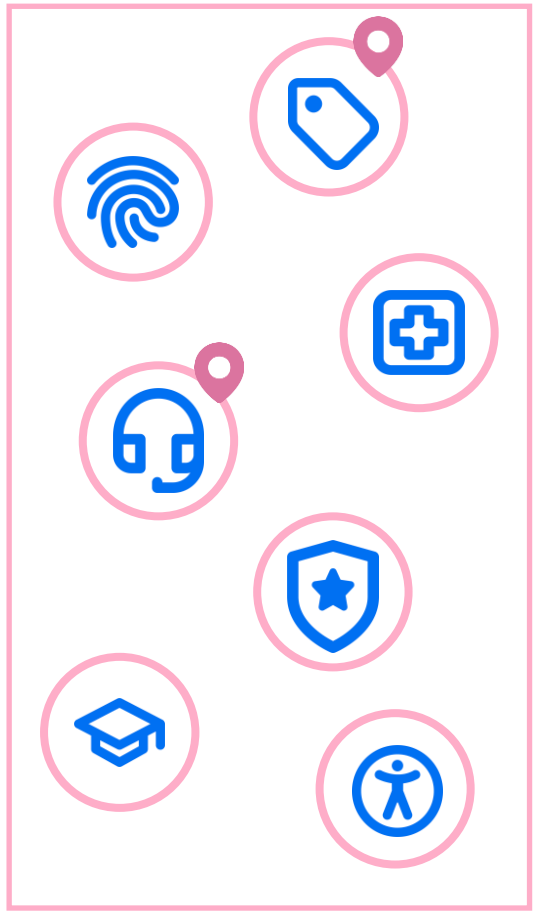
Development Team



Tools



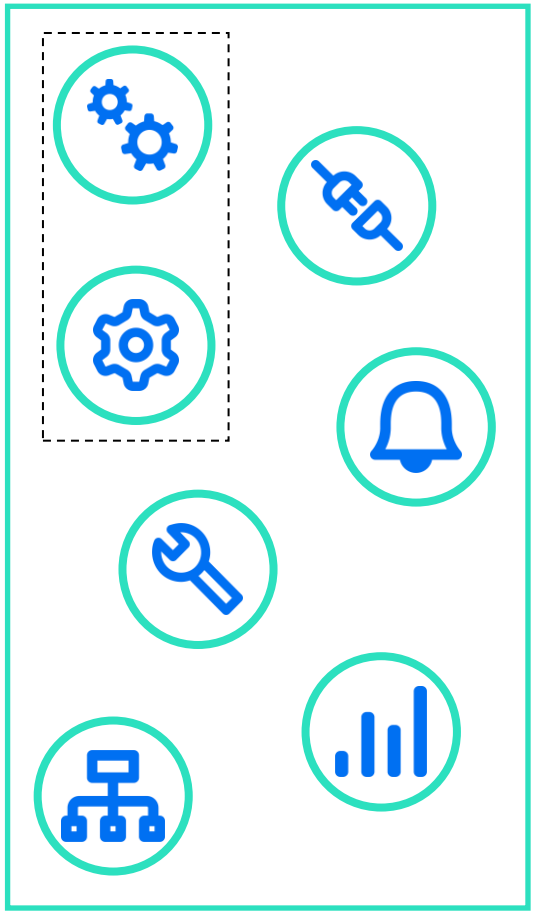
Compliance



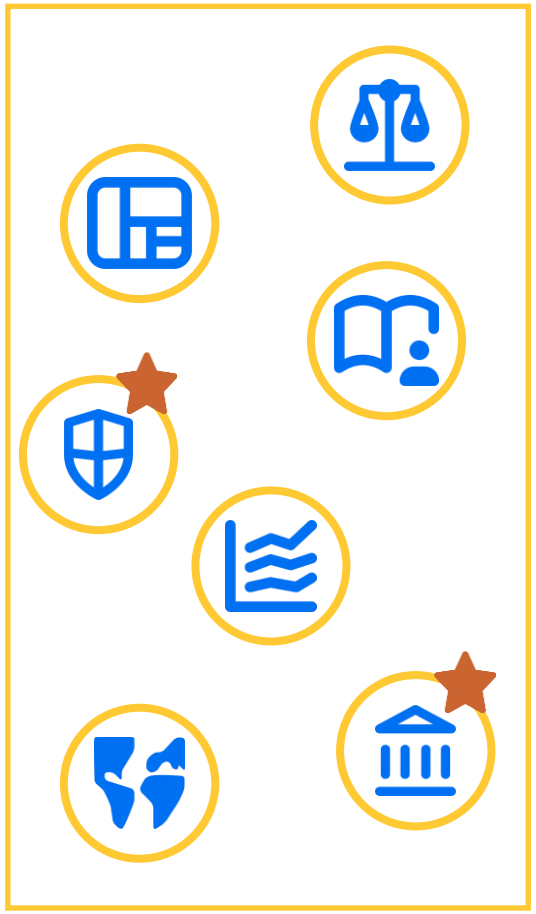
Guidelines

The Situation of a Development Team

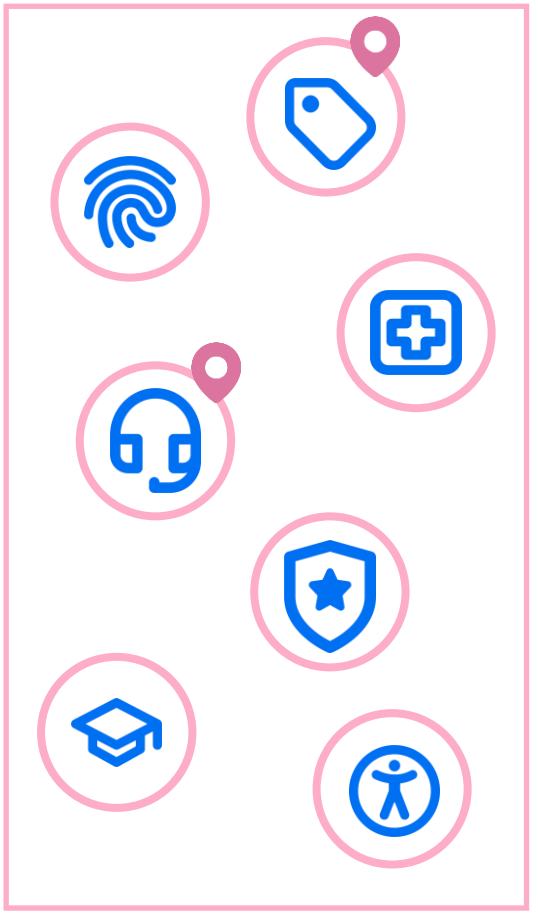
Development Team



Tools



Compliance



Guidelines

Platform Engineering

Platform Engineering – A Manifestation of DevOps

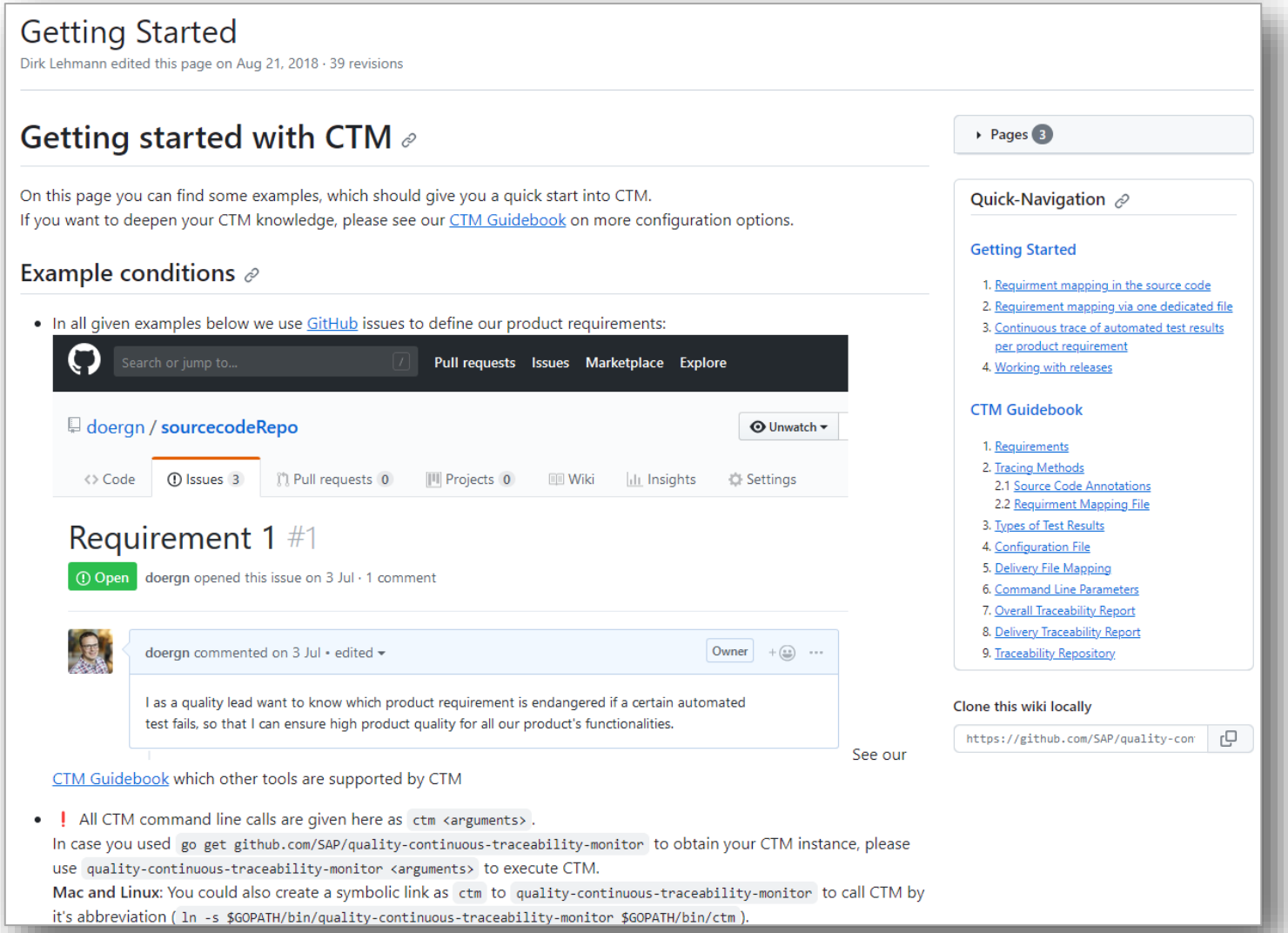
"The adoption of cloud and DevOps, while increasing the productivity of teams who can now move more quickly with reduced dependency on centralized operations teams and infrastructure, also has constrained teams who lack the skills to self-manage a full application and operations stack. Some organizations have tackled this challenge by creating **platform engineering product teams**. These teams operate an internal platform which **enables delivery teams to self-service deploy and operate systems with reduced lead time and stack complexity**. The emphasis here is on API-driven self-service and supporting tools, with delivery teams still responsible for supporting what they deploy onto the platform."

- Thoughtworks Technology Radar 2017

"**A digital platform** is a foundation of **self-service APIs, tools, services, knowledge and support** which are arranged as a **compelling internal product**. Autonomous delivery teams **can make use** of the platform to deliver product features at a higher pace, with reduced co-ordination."

- Evan Bottcher, Thoughtworks, 2018

Minimal Viable Platform



Getting Started

Dirk Lehmann edited this page on Aug 21, 2018 · 39 revisions

Getting started with CTM

On this page you can find some examples, which should give you a quick start into CTM.
If you want to deepen your CTM knowledge, please see our CTM Guidebook on more configuration options.

Example conditions

- In all given examples below we use GitHub issues to define our product requirements:

Search or jump to... / Pull requests Issues Marketplace Explore

doergn / sourcecodeRepo — Unwatch

Code | Issues 3 | Pull requests 0 | Projects 0 | Wiki | Insights | Settings

Requirement 1 #1

Open doergn opened this issue on 3 Jul · 1 comment

doergn commented on 3 Jul • edited — Owner

I as a quality lead want to know which product requirement is endangered if a certain automated test fails, so that I can ensure high product quality for all our product's functionalities.

See our CTM Guidebook which other tools are supported by CTM

- ❗ All CTM command line calls are given here as `ctm <arguments>`.
  In case you used `go get github.com/SAP/quality-continuous-traceability-monitor` to obtain your CTM instance, please use `quality-continuous-traceability-monitor <arguments>` to execute CTM.
  Mac and Linux: You could also create a symbolic link as `ctm` to `quality-continuous-traceability-monitor` to call CTM by it's abbreviation (`ln -s $GOPATH/bin/quality-continuous-traceability-monitor $GOPATH/bin/ctm`).

Pages 3

Quick-Navigation

Getting Started

1. Requirment mapping in the source code
2. Requirement mapping via one dedicated file
3. Continuous trace of automated test results per product requirement
4. Working with releases

CTM Guidebook

1. Requirements
2. Tracing Methods
   2.1 Source Code Annotations
   2.2 Requirment Mapping File
3. Types of Test Results
4. Configuration File
5. Delivery File Mapping
6. Command Line Parameters
7. Overall Traceability Report
8. Delivery Traceability Report
9. Traceability Repository

Clone this wiki locally

https://github.com/SAP/quality-con

Internal Developer Platform (IDP)

Internal
Targeting enterprise’s internal users only

Self service
Users access capabilities autonomously

User experience
Central Access point to documentation, support, tools, ...

Tool catalog
Comprehensive tools as part of aligned toolchains

Paved Roads (Golden Paths)
Drive standardization and leverage best practices

Compliance
Ensures rules and standards of the organization by default

APIs
Capabilities are composable (and optional)

How to Build a Internal Developer Platform

Treat the platform as a product

- Provides a “Holistic user experience”
- Provides support and great documentation

Strong collaboration with stream-aligned teams

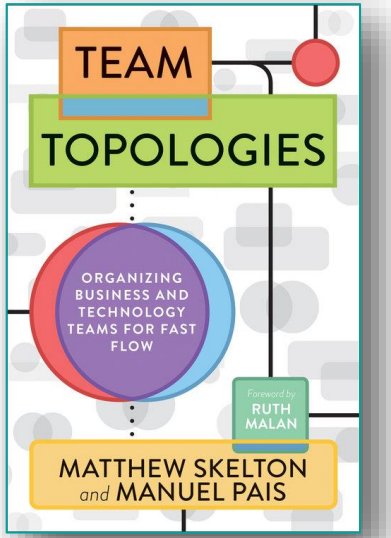
- Designed with users (internal teams) in mind with focus on UX and Developer Experience
- Measure customer satisfaction

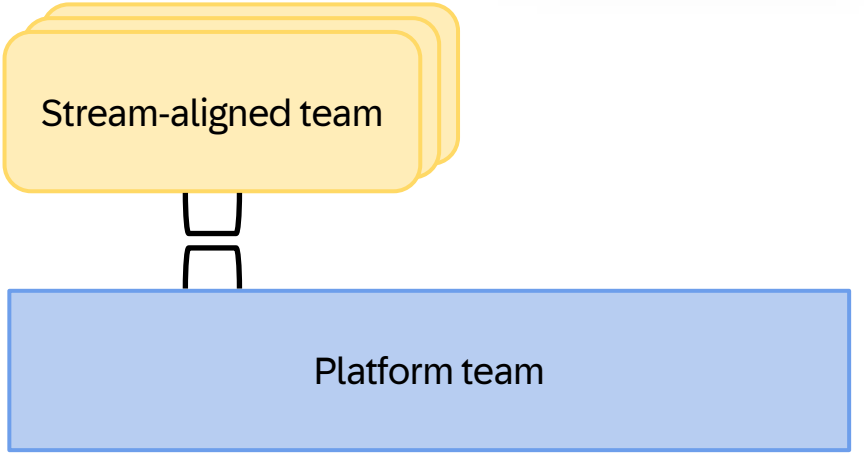
Focus on common uses cases

- Target ~80% of all Stream-aligned teams

Optional to use

- Platform lock-in decreases value
- Platform options increase value

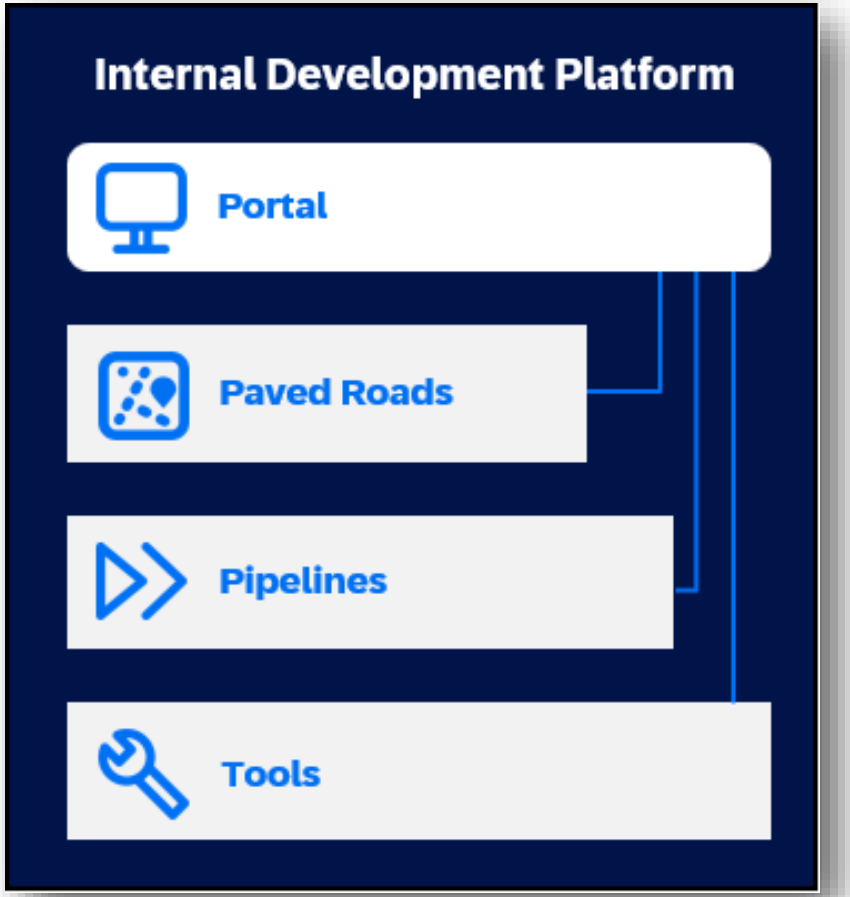




Hyperspace

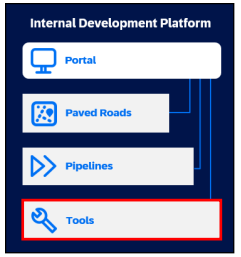
SAP's Internal Development Platform

Hyperspace - SAP's Internal Development Platform



Planning | Coding | Building | Validating | Deploying | Releasing | Cross

Tools



Tool catalog
Self-service inventory

Consolidate tools
One organization

Harmonize support channels
One support channel

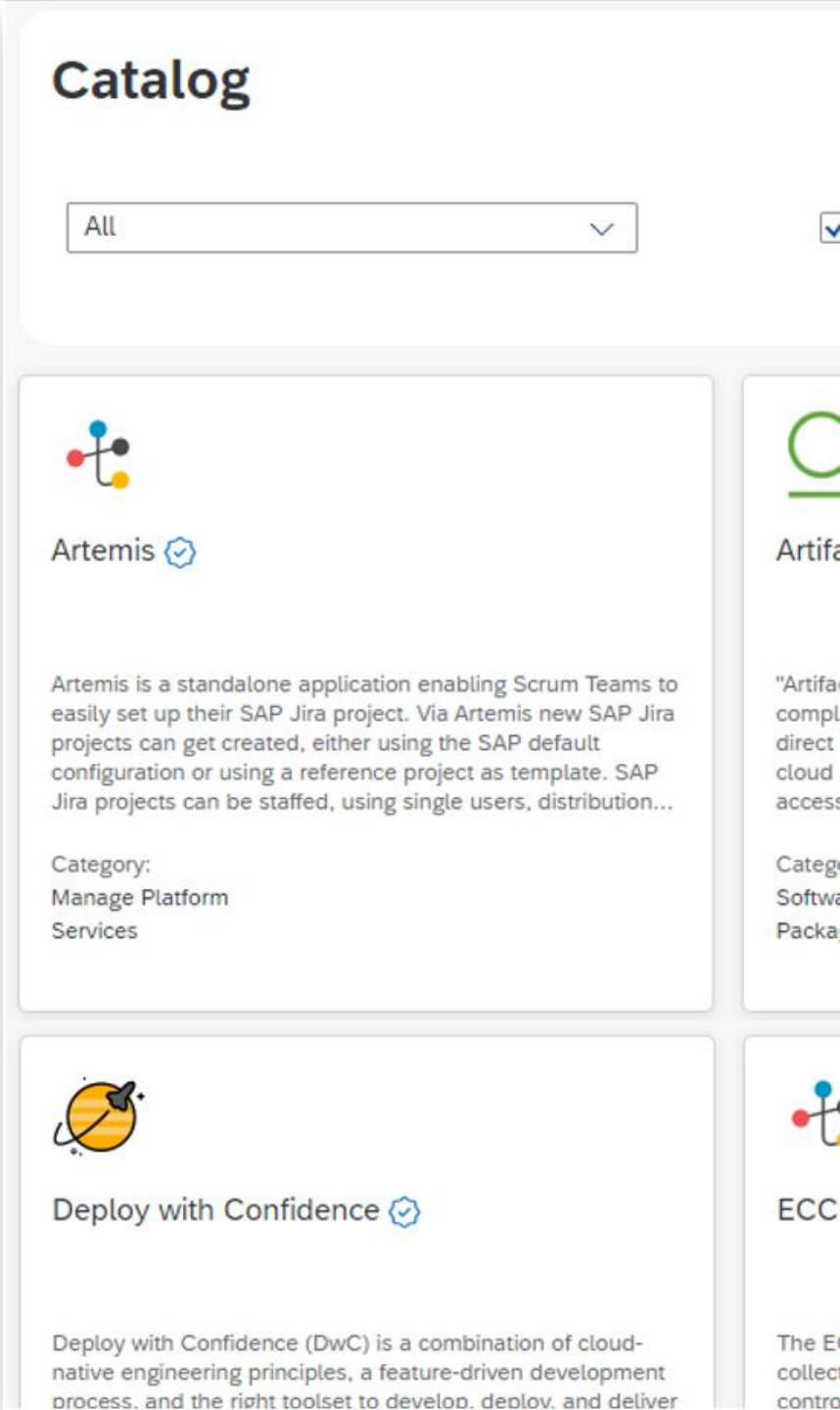
Integrate tools
Provide open APIs and align data models

Adjust SLAs
Communicate and measure SLAs

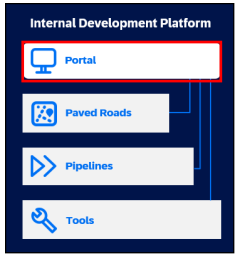
User management
One authorization and access management

Improve documentation
Structure documentation and terminology

Services
Replace individual tools with services



Portal



Self service
Users access capabilities autonomously

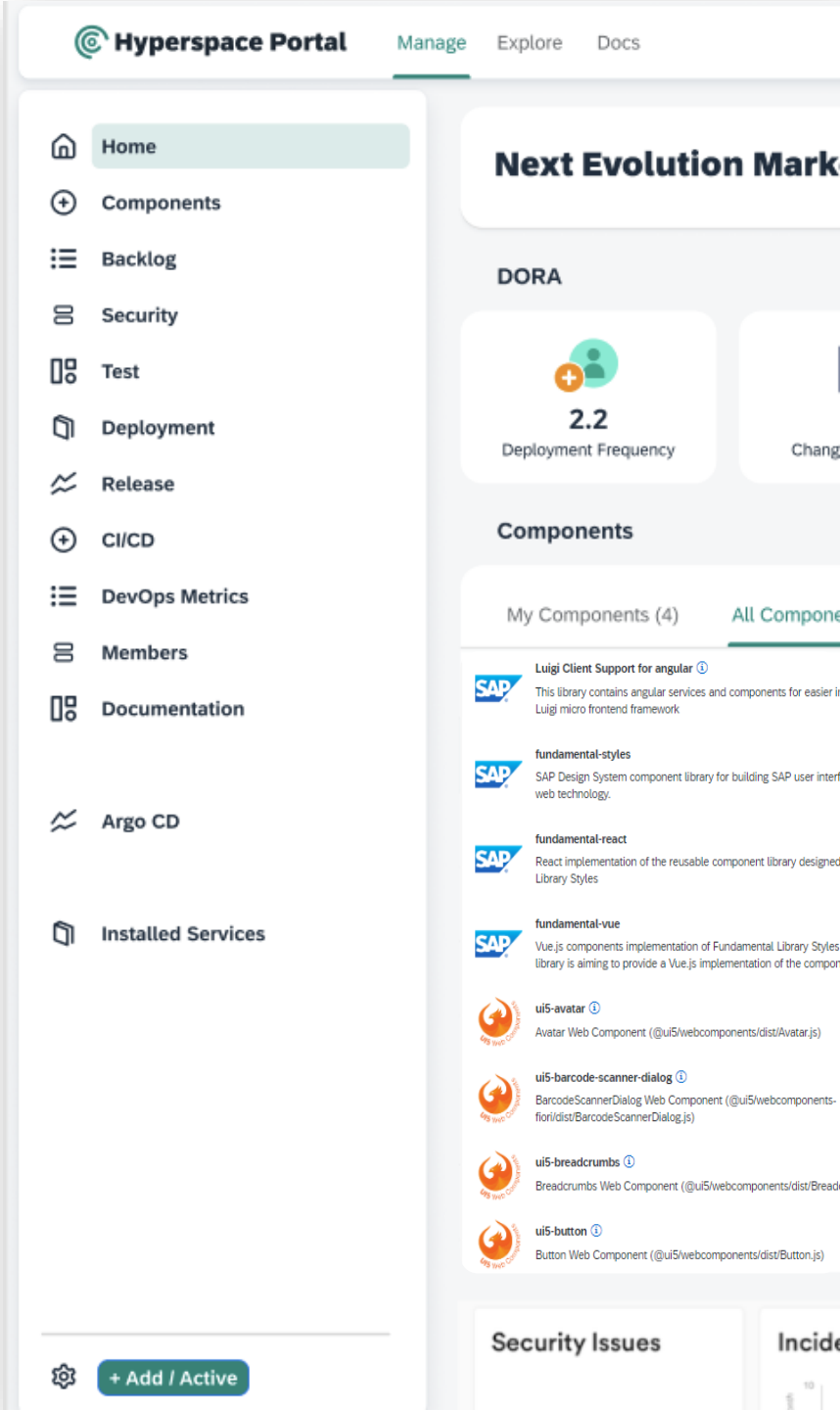
Support
Context enriched support

Central access point
Entry point for all platform capabilities

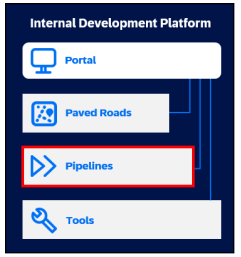
Documentation
Access to documentation

Tool catalog
Self-service inventory

Extensible
Open for contributions



Pipelines



Integrate tools
Provide open APIs and align data models

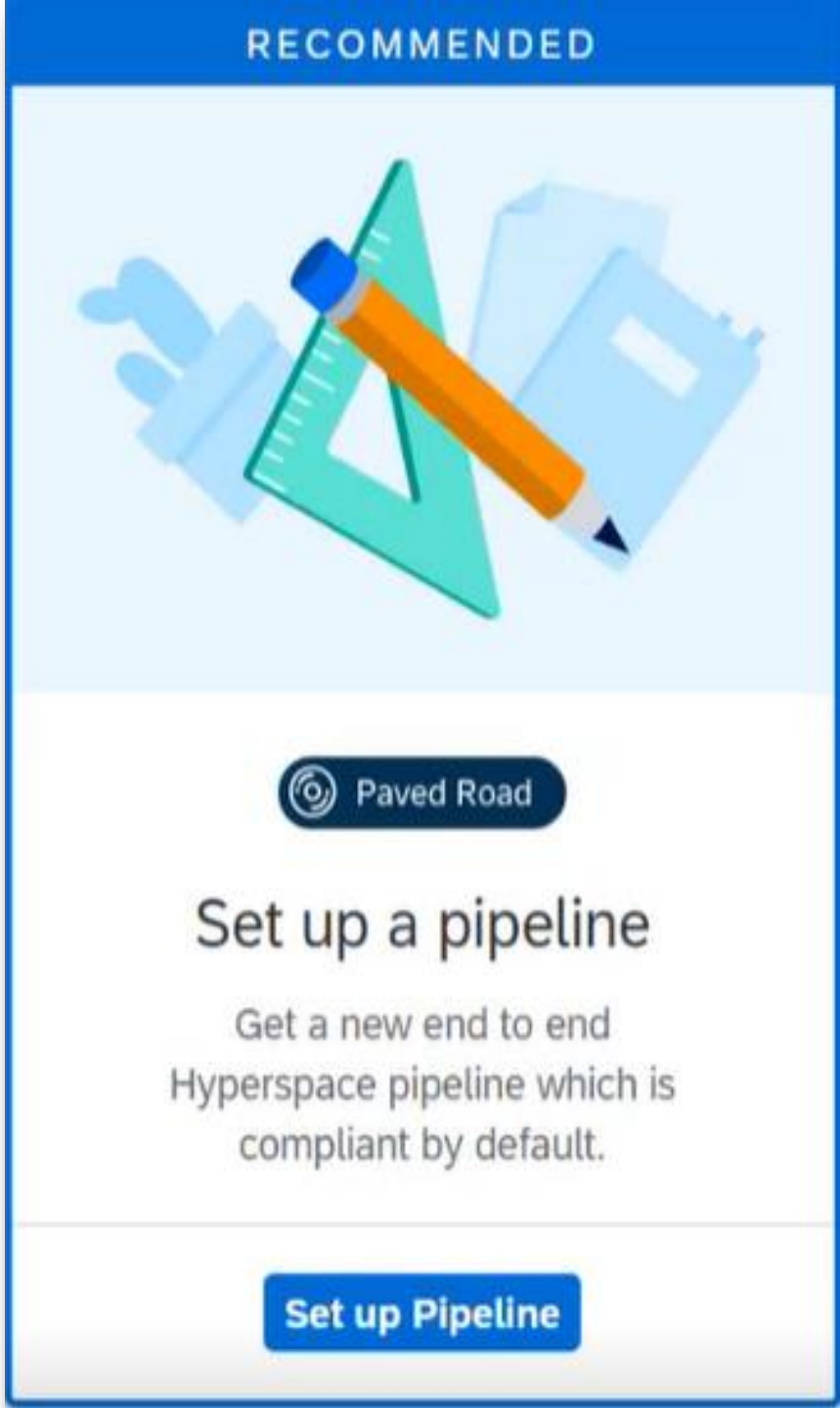
Grow on demand
Extend pipeline following team needs

Fast set up
Quick access to CI/CD pipelines

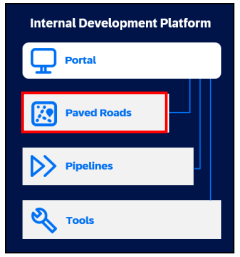
Extensible
Open for contributions

Compliance by default
Set up according to corporate rules and governance

APIs
Library for easy tool access

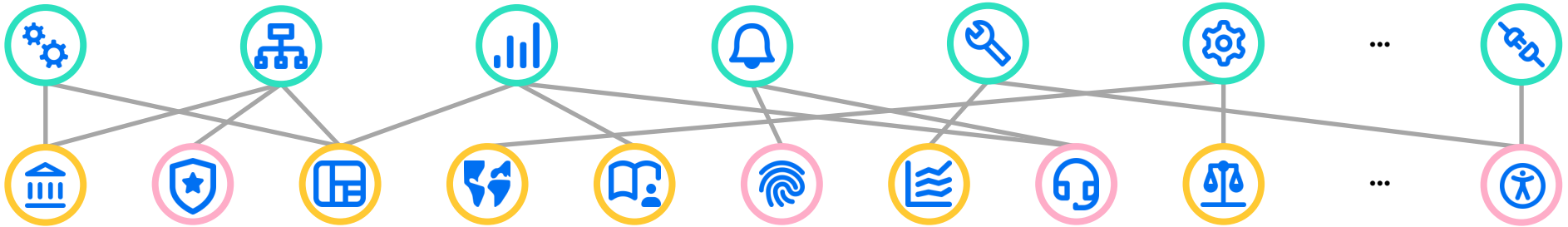


Paved Roads

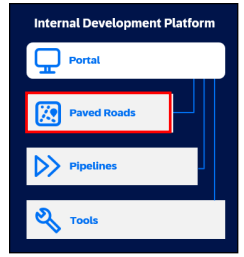


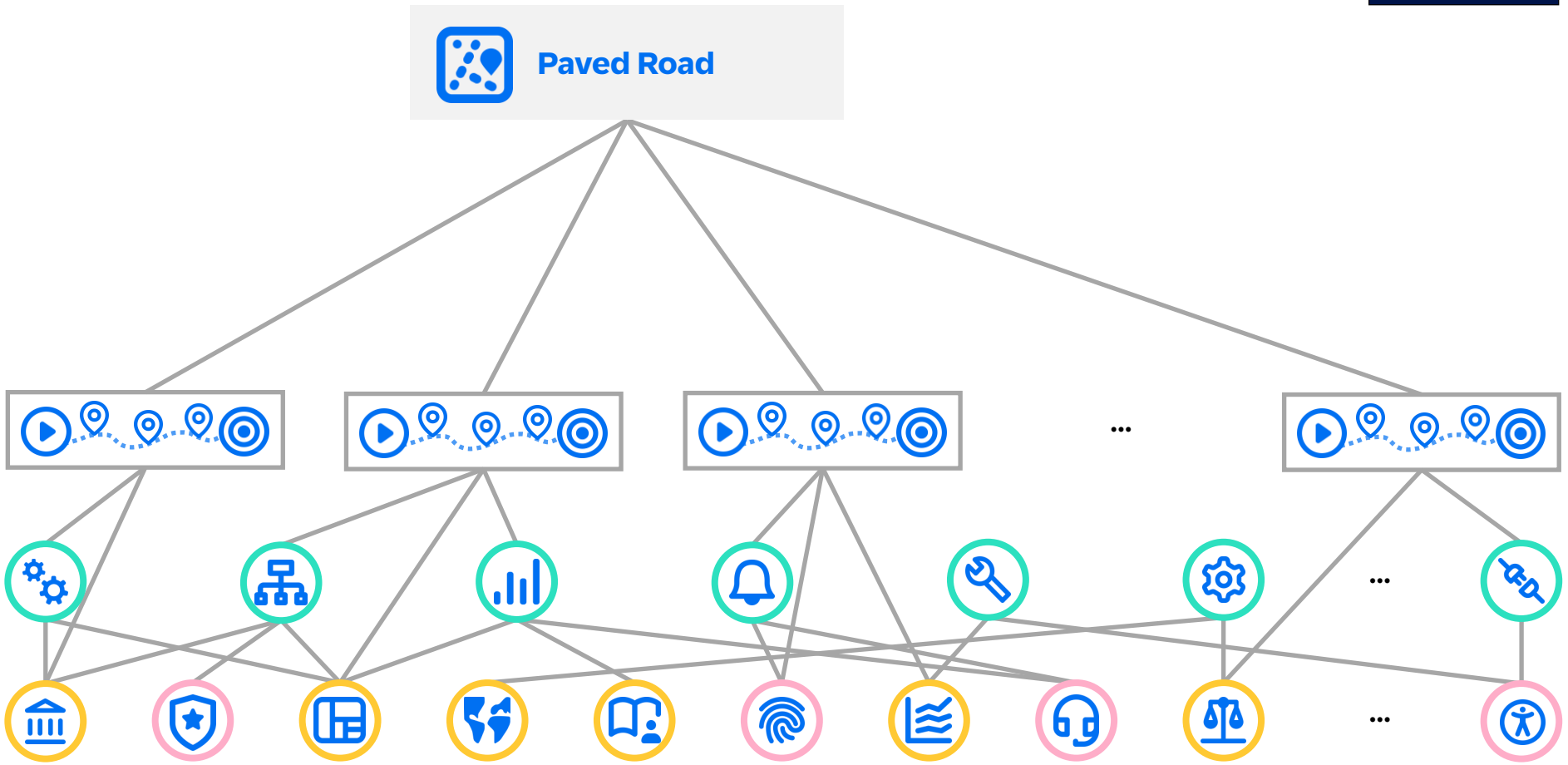
Tools

Compliance & Guidelines

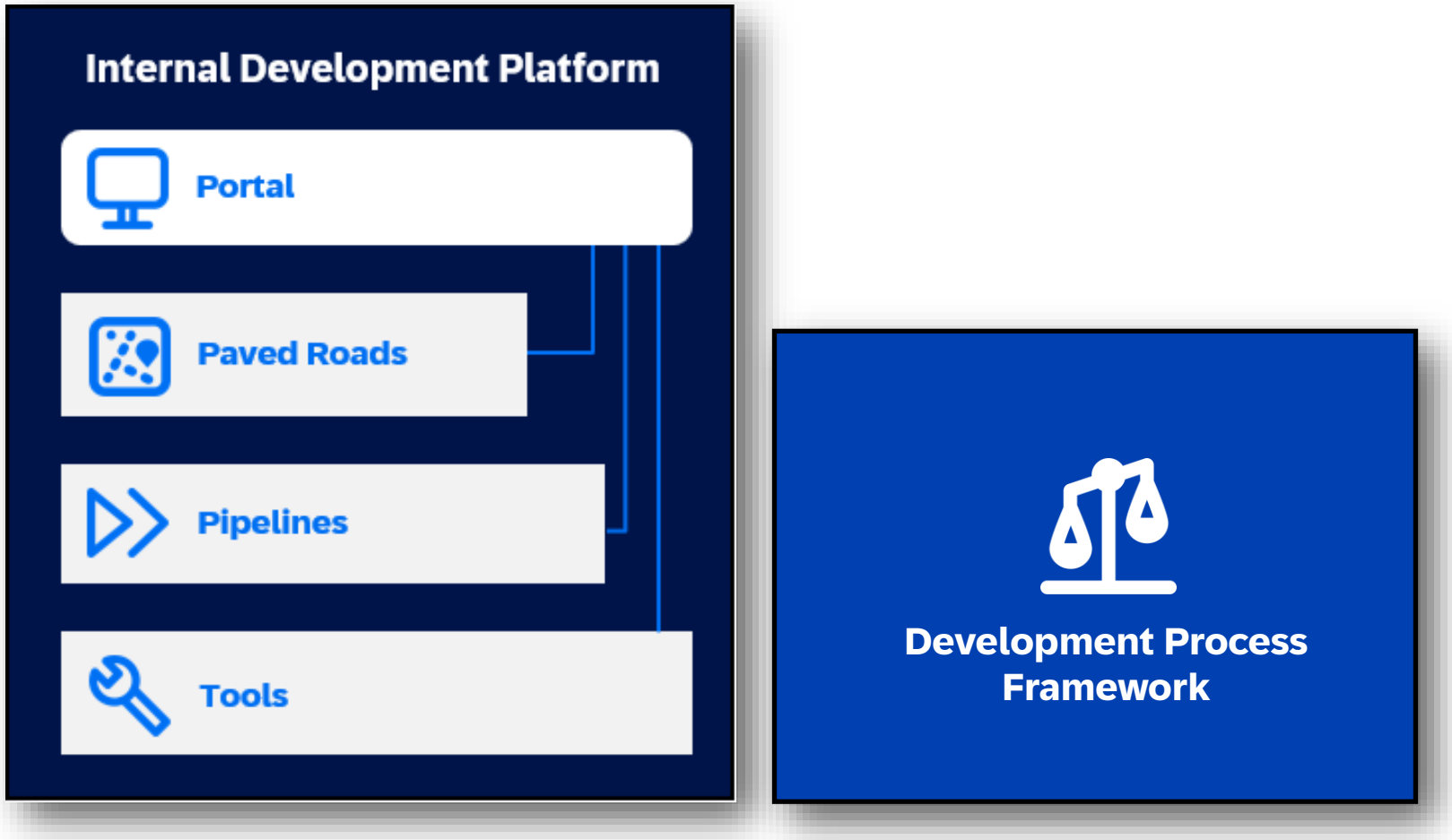


Paved Roads

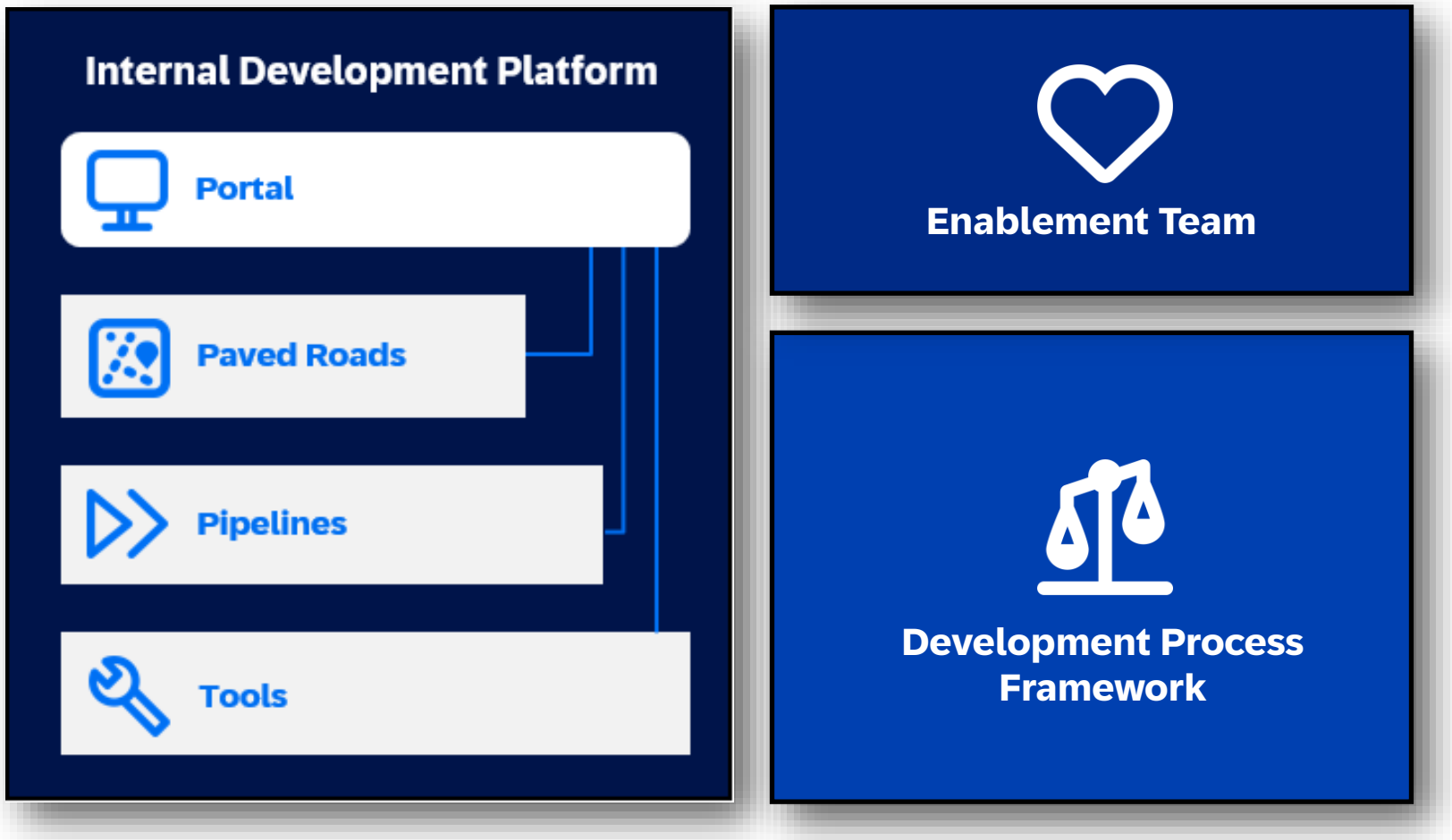

Internal Development Platform
Portal
Paved Roads
Pipelines
Tools


Paved Road
Development Procedures
Tools
Compliance & Guidelines

Development Process Framework



Enablement Team



Criticism on Platform Engineering

Technology Radar

An opinionated guide to today's technology landscape

"The adoption of cloud and DevOps, while increasing the productivity of teams who can now move more quickly with reduced dependency on centralized operations teams and infrastructure, also has constrained teams who lack the skills to self-manage a full application and operations stack. Some organizations have tackled this challenge by creating platform engineering product teams. These teams operate an internal platform which enables delivery teams to self-service deploy and operate systems with reduced lead time and stack complexity. The emphasis here is on API-driven self-service and supporting tools, with delivery teams still responsible for supporting what they deploy onto the platform.
Organizations that consider establishing such a platform team should be very cautious not to accidentally create a separate DevOps team, nor should they simply relabel their existing hosting and operations structure as a platform."

- Thoughtworks Technology Radar 2017

- One size (does not) fit all
- Stream aligned teams loose (to much) control
- "Eierlegende Wollmilchsau"

Learnings from SAP's Internal Development Platform (so far)

Put responsibility into one organization
Distributed setting causes lot of (unnecessary) friction

Paved Roads / Golden Paths are your friend
Help to get systemic view

It's a marathon
It takes times to change status quo

KPI reporting is difficult
Which KPI indicates platform success?

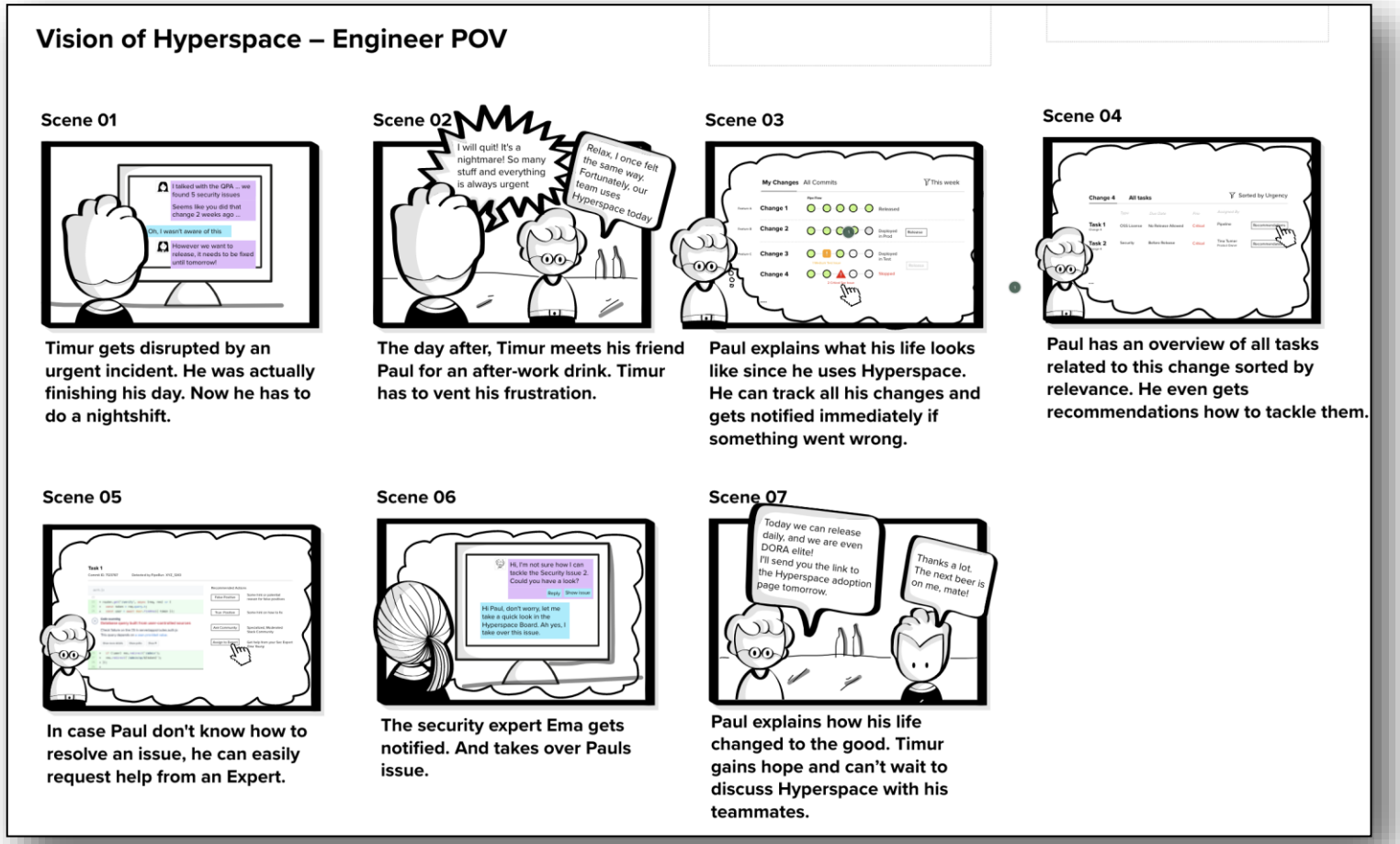
Requires proper staffing
Dedicated people to work on the platform.
Inner source won't solve the problem

Source: LinkedIn post by Matthew Skelton, Sep. 2023

Trust
Reliability and ability to execute is key

Learnings from SAP's Internal Development Platform (so far)

Share a vision
For customers and platform team members



Thank you.

Contact information:

Dirk Lehmann (he/him)
dirk.lehmann@sap.com

@doergn@hachyderm.io