

Introducing a **product mindset** in a platform team

Stéphane Di Cesare, Senior Platform Engineer

SRE Meetup Munich, 14.04.2025



SRE Meetup Munich

DKB
Das kann Bank

What is DKB

Not all banks are the same – and many banks don't even know what they stand for. But we do.

We invest in the things that matter: renewable energies, affordable housing, day-care centres, schools, hospitals. We support civic participation and are partners of the domestic agricultural sector.



Founded in 1990
Wholly owned subsidiary of BayernLB



5,3 million customers
Local authorities, companies, retail clients



EUR 121 billion balance sheet total
Among the top 20 banks in Germany



5000 employees



EUR 89.3 billion credit volume

Wir sind
#geldverbesserer

100 %
#geldverbesserer



Speaker presentation

Stéphane Di Cesare

DKB Standard Operations Platform, Platform Experience

Started at DKB in April 2023

Background in technology consulting, infrastructure automation, sales engineering, technical support, QA and software integration

Focusing on the **value** brought by technology

Member of **CNCF Platform Engineering Working Group**

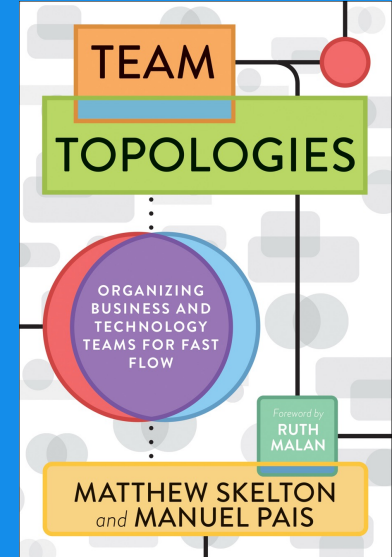
Interested in **linguistics and languages**



What is a platform?

“Platform team: a [team] that provides a compelling product to accelerate delivery by stream-aligned teams”

teamtopologies.com



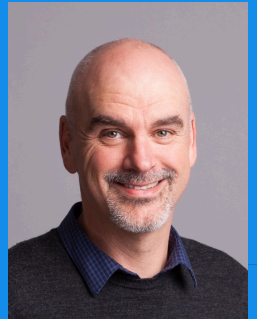
What is a platform?

“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 coordination.”

Evan Bottcher

<https://martinfowler.com/articles/talk-about-platforms.html>



Where we started

formed from different infrastructure-related teams

core: **container platform** using Crossplane

“focus on **Engineering**”

- workloads run!
- communication focused on infrastructure

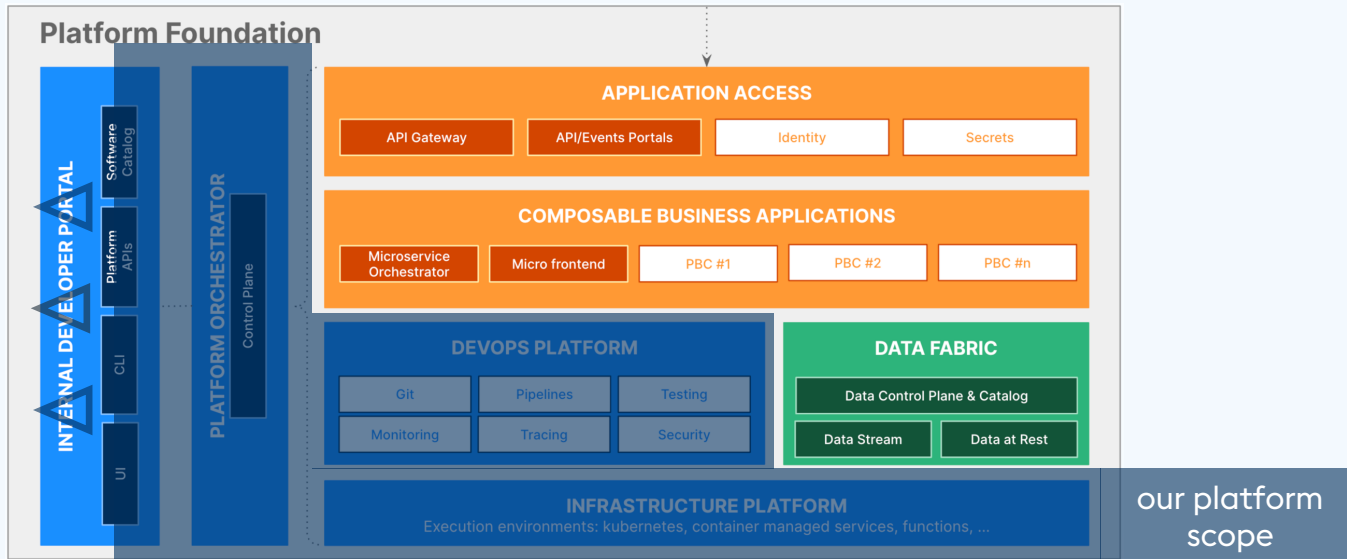


diagram courtesy of Giulio Roggero, mia Platform

Challenges

Users!

- ... don't know about the platform
- ... don't understand the scope of the platform
- ... use the platform wrong

But is the platform really solving **their problems**?

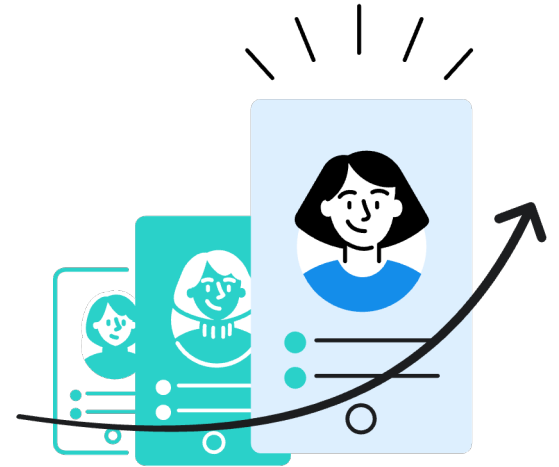


Goals

How can we **show the value** of the platform?

How can we show the **state** of the platform?
(and how it can improve)

How can we **share information** effectively about
the platform?



Important topics in our **platform journey**

1. **Product** mindset
2. **Defining** the platform
3. Clarifying **platform maturity**
4. **Information** – communication and documentation

1. Product mindset

Principles

Having a product mindset means focusing on **user value**.

user problems before solutions

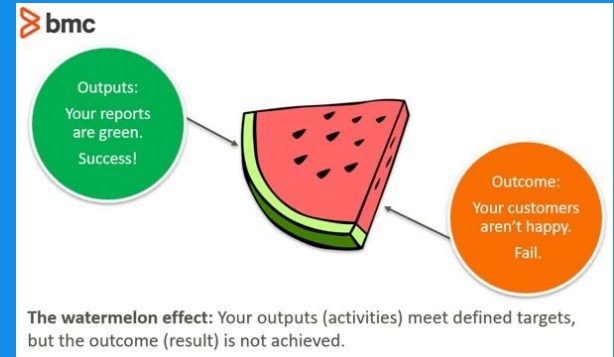
a team with a product mindset is tasked with problems to solve and decides how to solve them

outcome before output

a team with a product mindset is accountable for the outcome, not the output

products before projects

think about lifecycle and ownership



intro to Kubernetes as Product:

<https://www.youtube.com/live/lxUNUh9sWSI?t=4589s>

(Dominik Kress, Giant Swarm)



references:

Marty Cagan – Empowered + Transformed

Matthew Skelton & Manuel Pais – Team Topologies

Melissa Perri – Escaping the Build Trap

Product mindset

Some tips

Define a clear **vision** (how to provide value long-term)

Develop a **strategy** (how to realize value in the middle term)

Discover user problems – **design** the solution

Balance easy wins and long-term value

intro to Kubernetes as Product:

<https://www.youtube.com/live/lxUNUh9sWSI?t=4589s>

(Dominik Kress, Giant Swarm)



“We are making cloud simpler for developers”

“We will implement a cloud policy engine to ensure that workloads are compliant to current regulations”

Use shadowing to look how users are working, and think how the platform can make their work more effective



references:

Marty Cagan – Empowered + Transformed

Matthew Skelton & Manuel Pais – Team Topologies

Melissa Perri – Escaping the Build Trap

Product mindset

Challenges we encountered

unclear **ownership** (especially for product management)

who is the **user** and what is the **value**?

users try to **force the implementation of a specific requirement**

clarify the responsibility for **product management** activities (possibly with delegation)

users

- builders (external, internal)
- enablers
- regulatory
- “viewers”

value

- be more efficient
- avoid risks

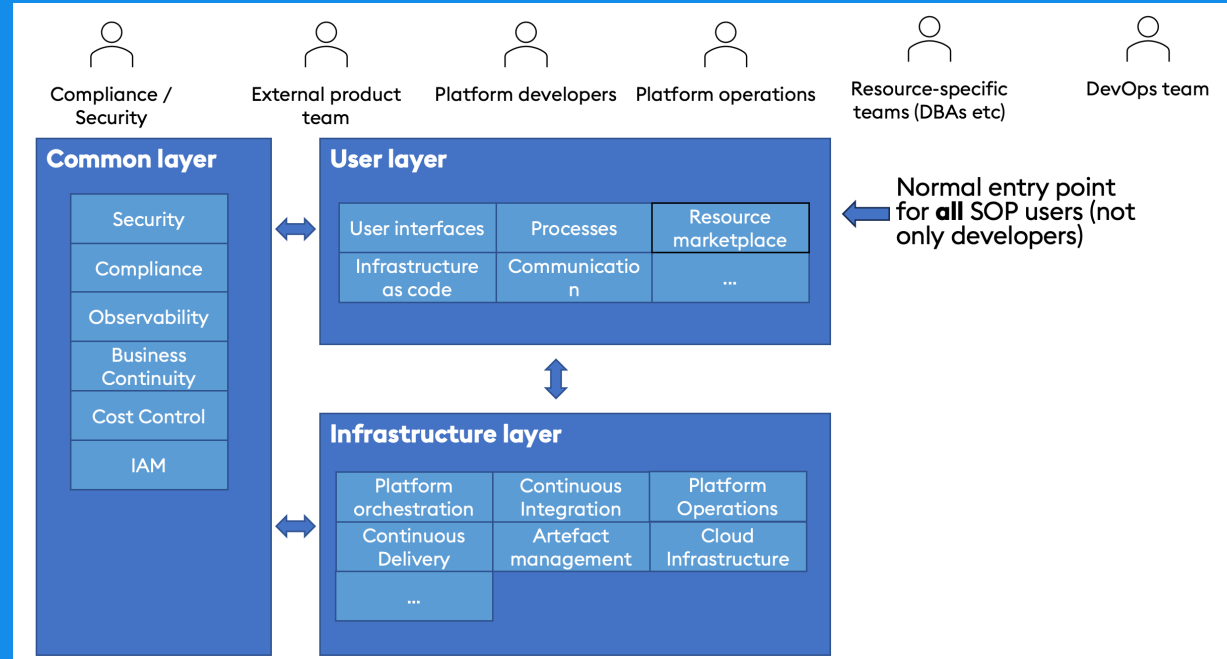
ensure that **decision stays at the platform level**

users must explain the problem behind their requirements, and cannot force a specific implementation

2. Defining the platform

Introduce a platform architecture / product view

The platform architecture breaks down the platform into capabilities



other reference platform architectures:

Humanitec: <https://humanitec.com/reference-architectures>

Gartner: <https://www.gartner.com/en/articles/what-is-platform-engineering>

(see also a variation: <https://www.syntasso.io/post/platform-engineering-orchestrating-applications-platforms-and-infrastructure>)

Defining the platform

Detail capabilities

Use cases corresponding to the capability are detailed (including actor and value)

Solution

Documentation (implementation and user documentation)

Responsibility

Status (in backlog, beta, productive, ...)

Reporter	Use Case (action + actor)	Value	Comments	Solution	Status	Documentation (solution + usage)	Responsibility
<i>John Doe</i>	developer wants to view the total cost of resources related to its application	enable cost visibility					
<i>Jane Doe</i>	cloud engineer wants to be able to push a GitOps cluster configuration change with a maximum of one MR requiring approval	being able to act quickly in case of an emergency					

3. Clarifying platform maturity

Where we started from

	Aspect	Provisional	Operational	Scalable	Optimizing
<u>Investment</u>	<i>How are staff and funds allocated to platform capabilities?</i>	Voluntary or temporary	Dedicated team	As product	Enabled ecosystem
<u>Adoption</u>	<i>Why and how do users discover and use internal platforms and platform capabilities?</i>	Erratic	Extrinsic push	Intrinsic pull	Participatory
<u>Interfaces</u>	<i>How do users interact with and consume platform capabilities?</i>	Custom processes	Standard tooling	Self-service solutions	Integrated services
<u>Operations</u>	<i>How are platforms and their capabilities planned, prioritized, developed and maintained?</i>	By request	Centrally tracked	Centrally enabled	Managed services
<u>Measurement</u>	<i>What is the process for gathering and incorporating feedback and learning?</i>	Ad hoc	Consistent collection	Insights	Quantitative and qualitative

standalone platform team
intention to work as product
value not concretely expressed

leadership committed to platform
adoption not always related to value

standard tooling and self-service for
many areas
documentation not centralized

no central view of capabilities

measurement tooling available
no central agreement on
measurement

from CNCF Platform Maturity Model <https://tag-app-delivery.cncf.io/whitepapers/platform-eng-maturity-model/>

Nicki Watt's talk at Platform Engineering Day 2024 <https://www.youtube.com/watch?v=MtYn60VWtJk>

3. Clarifying platform maturity

Where we wanted to go

	Aspect	Provisional	Operational	Scalable	Optimizing
<u>Investment</u>	<i>How are staff and funds allocated to platform capabilities?</i>	Voluntary or temporary	Dedicated team	As product	Enabled ecosystem
<u>Adoption</u>	<i>Why and how do users discover and use internal platforms and platform capabilities?</i>	Erratic	Extrinsic push	Intrinsic pull	Participatory
<u>Interfaces</u>	<i>How do users interact with and consume platform capabilities?</i>	Custom processes	Standard tooling	Self-service solutions	Integrated services
<u>Operations</u>	<i>How are platforms and their capabilities planned, prioritized, developed and maintained?</i>	By request	Centrally tracked	Centrally enabled	Managed services
<u>Measurement</u>	<i>What is the process for gathering and incorporating feedback and learning?</i>	Ad hoc	Consistent collection	Insights	Quantitative and qualitative

make value clear

communicate value

central view of documentation
self-service tooling

central view of capabilities

impact of capabilities is measured

from CNCF Platform Maturity Model <https://tag-app-delivery.cncf.io/whitepapers/platform-eng-maturity-model/>


Nicki Watt's talk at Platform Engineering Day 2024 <https://www.youtube.com/watch?v=MtYn60VWtJk>

4. Information



Matthew Skelton #BLM  
@matthewpskelton



What if the most important part of "platform engineering" is maintaining a high quality wiki with proven, empathic patterns for Stream-aligned teams to follow? 

#platformengineering #DevOps

Posted Oct 15, 2022 at 7:51AM

“[Documentation] is the single thing that will give you space to do everything else”

Sarah Wells

former Financial Times platform lead

4. Information

Communication channels

Communication is a **feature of the platform**

Communication should be **streamlined as much as possible** (but cover all use cases)

Communication is not only incidents or change requests!

Some communication use cases

- quick questions without guarantee of answer
- “platform consulting”
- communication from the platform (outages, new features, ...)
- emergency communication (critical incident, security emergency etc)



references:

Sumeet Gayathri Moghe – The Async-First Playbook
Matthew Skelton & Manuel Pais – Team Topologies

4. Information

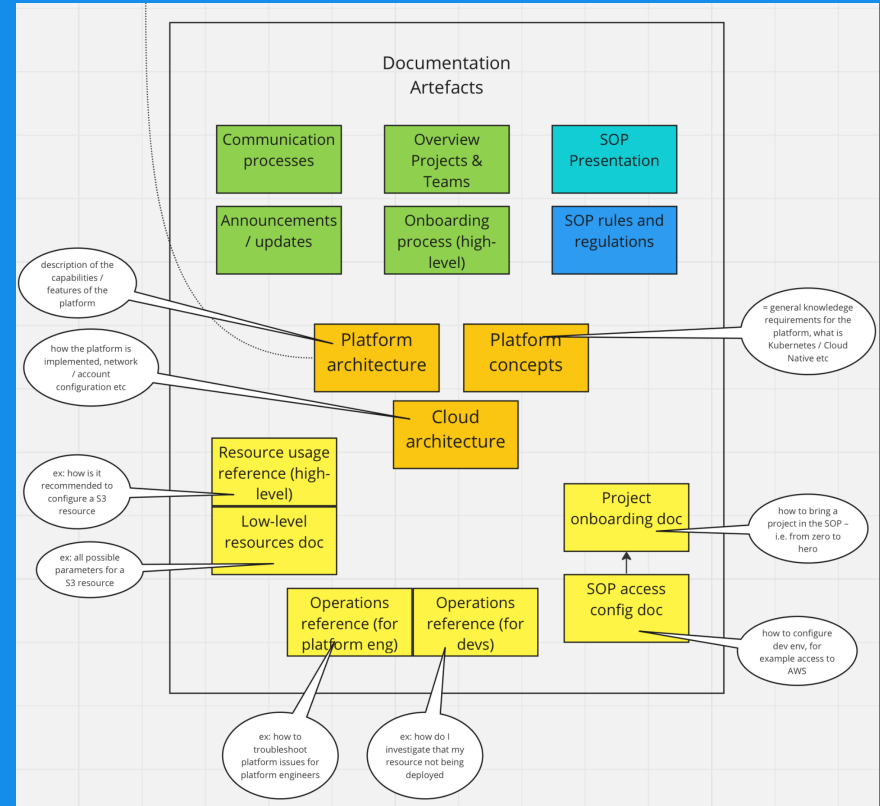
Documentation

Documentation is **not only** user documentation!

Documentation is **a feature of the platform**, with different persona and use cases

Different persona might need different media

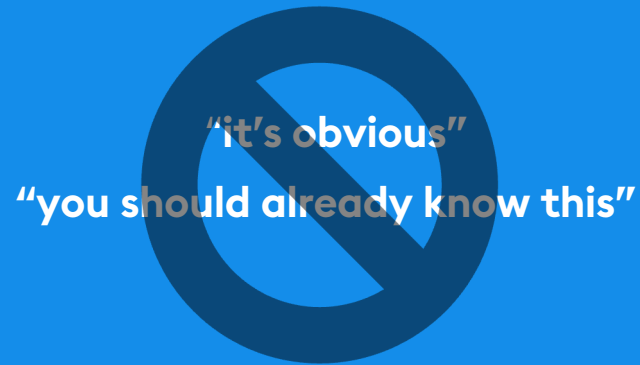
Avoid **shadow documentation**: the first priority is **keeping the overview**



4. Information

Psychological safety

Communication with end users is an important **product discovery** channel!



The future...

Next steps

Improve participation of **all stakeholders** while **keeping the overview**

Encourage **contact** between users and platform engineers, for example through shadowing

Encourage users and platform engineers to consider the **problem before the solution**

Contact me on **LinkedIn**: <https://linkedin.com/in/sdicesare>

CNCF Platform Engineering Working Group

<https://taq-app-delivery.cncf.io/was/platforms/>

<https://www.linkedin.com/company/platform-engineering-working-group>

We are hiring!

