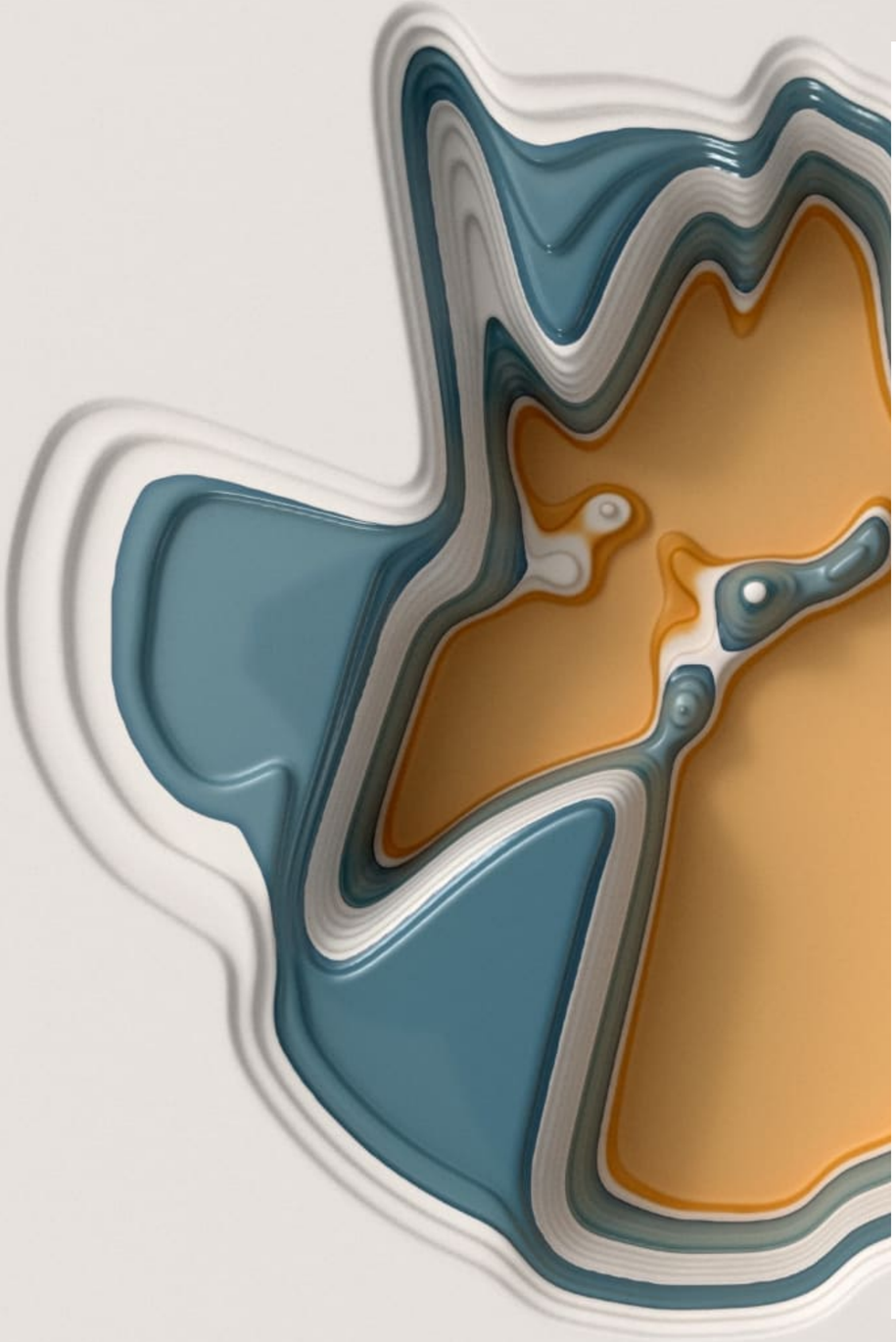


INNOQ Technology Day 2023

Architektur-Prinzipien, Makroarchitektur und The Paved Road

Die wenigen Entscheidungen, die 1000 Entscheidungen treffen

INNOQ

SVEN JOHANN

Welche wenigen Entscheidungen helfen, viele andere Entscheidungen zu treffen?

Wie skaliert man Entscheidungen?

Grundlegendes zu Entscheidungen

Typ 1 und Typ 2 Entscheidungen

Umstellung des Hosting Providers

Einführung einer neuen Programmiersprache im Unternehmen

Auswahl einer Open-Source-Bibliothek

Änderungen an der Public API

Anpassungen in der Datenbank

Erste Experimente mit einer neuen Programmiersprache

Irreversibel ⟷ **Reversibel**

- Globaler Konsens notwendig
- (Fast) Unmöglich zu ändernde Festlegungen
- Gut durchdachte Lösungen notwendig

- Lokale Entscheidung
- Niedrige Kosten beim Zurücknehmen einer Entscheidung
- Ad-Hoc-Entscheidungs-findung akzeptabel

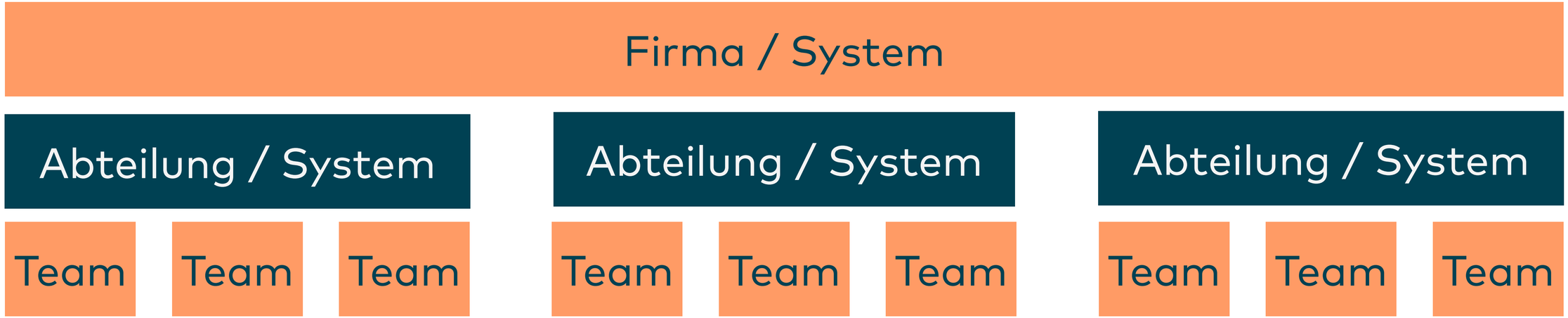
Von Sam Newman: Monoliths to Microservices

Anatomie einer Entscheidung

Autonomie und Entscheidungen

Autonomie bezieht sich auf die Fähigkeit einer Gruppe, unabhängige Entscheidungen zu treffen und Handlungen auszuführen, ohne dabei von externen Einflüssen oder Kontrollen stark eingeschränkt zu sein. (ChatGPT)

Wie unabhängig können wir sein?



Wie fördert man Autonomie/ Entscheidungen und schafft es gleichzeitig, dass nicht alle Teams in unterschiedliche Richtungen laufen?

Governance

to govern - was heißt das?

Regieren
Zwingen
Kontrollieren

(An-)leiten
Orientieren
Beeinflussen

https://speakerdeck.com/birgitta410/cultivating-architecture

Architekturprinzipien

Makroarchitektur

Follow the Trail

Architektur-Prinzipien

(anleiten, orientieren, beeinflussen)

Was ist ein Prinzip?

- “Grundsatz, den jemand seinem Handeln und Verhalten zugrunde legt”
- “allgemeingültige Regel, bestimmte Idee, bestimmte Grundlage, auf der etwas aufgebaut ist, nach der etwas abläuft”

Was ist ein Architekturprinzip?

“Eine deklarative Aussage, die mit der Absicht getroffen wird, architektonische Entscheidungen zu lenken, um eine oder mehrere Qualitäten eines Systems zu erreichen.”

Strategisches Ziel / Risiko

Anforderungen

Prinzip

Entscheidung

Orientierung

Produkt weltweit verfügbar (Policies, Regulation ggf. unbekannt)

Deployment on-prem und in der Cloud möglich

Nutze keine Cloud-spezifischen Services

Entspr. Entscheidungen auf Teamebene, z.B. "kein AWS SQS, sondern Kafka"

Begründung

“Real-World” Examples

STRATEGIC GOALS
Goals of the business side

REDUCE TIME TO MARKET
Establish fast feedback loops to learn, validate and improve. Remove friction, hand-offs and undifferentiated work.

SUPPORT DATA-DRIVEN DECISIONS
Provide relevant metrics and data for user and market insights. Validate hypothesis for problems worth solving.

MOBILE FIRST
Start small and use device capabilities.

BEST TALENT
Autonomy, Purpose and Mastery: We know why we do things, we decide how to approach them and deliberately practice our skills.

COST EFFICIENCY
Run your segment in the right balance of cost and value.

ONE SCOUT IT
Foster collaboration. Harmonize and standardize tools. Pull common capabilities into decoupled platform services.

ARCHITECTURAL PRINCIPLES
High-Level Principles

ORGANIZED AROUND BUSINESS CAPABILITIES
Build teams around products not projects. Follow the domain and respect bounded contexts. Make boundaries explicit. Inverse Conway Maneuver.

ELIMINATE ACCIDENTAL COMPLEXITY
Strive to keep it simple. Don't over-engineer. Focus on necessary domain complexity.

LOOSELY COUPLED
By default avoid sharing and tight coupling. No integration database. Don't create the next monolith.

MACRO AND MICRO ARCHITECTURE
Clear separation. Autonomous micro services within the rules and constraints of the macro architecture.

SECURITY, COMPLIANCE AND DATA PRIVACY
Build with least privilege and data privacy in mind. Know your threat model. Limit blast radius.

AWS FIRST
Favor AWS platform service over managed service, over self-hosted OSS, over self built solutions.

DESIGN AND DELIVERY PRINCIPLES
Tactical measures

YOU BUILT IT, YOU RUN IT
The team is responsible for shaping, building, running and maintaining its products. Fast feedback from live and customers helps us to continuously improve.

CROSS-FUNCTIONAL TEAMS
Engineers from all backgrounds work together in collaborative teams as engineers and share responsibilities. No silos.

AUTONOMOUS TEAMS
Make fast local decisions. Be responsible. Know your boundaries. Share findings.

BE BOLD
Go into production early. Value monitoring over tests. Fail fast, recover and learn. Optimize for MTTR not MTBF.

DATA-DRIVEN / METRIC-DRIVEN
Collect business and operational metrics. Analyze, alert and act on them.

INFRASTRUCTURE AS CODE
Automate everything: Reproducible, traceable, auditable and tested. Immutable servers.

SCOUT 24

Version 2.0

Icons made by Freepik from www.flaticon.com are licensed under CC BY 3.0

https://gotoams.nl/2017/sessions/219/enterprise-fast-lane-transforming-to-microservices-in-the-cloud

Minimize technology variation

We deliberately minimize the variety of technologies in our system to drive efficiency, and, at the same time, have a managed program of innovation. To achieve this, we monitor current technologies in use, gravitate to those that most appropriately address the given challenges and selectively experiment with new technologies to find those that will improve our products.

Motivating values

- Ease of people movement
- Company over team over individual

https://github.com/Scout24/scout24-engineering-values-and-principles/blob/master/2-engineering-principles.md

Independent Systems Architecture

PRINCIPLES FOR MICROSERVICES

https://isa-principles.org/ by INNOQ

1 | Modules

2 | Macro / Micro Architecture

3 | Container

4 | Integration & Communication

5 |Authentication & Metadata

6 | Independent Continuous Delivery Pipeline

7 | Standardized Operations

8 | Standards: Interface only

9 | Resilience

https://isa-principles.org/ by INNOQ

Orientierung

Hohe Kundenzufriedenheit

Wichtigste Use Cases > 99,95% Verfügbarkeit

Benutze Graceful Degradation wo möglich (Resilience)

Wir zeigen alle (Filme)***, wenn Subscription-Status unbekannt

Begründung

Vorgehen

Einprägsamer Titel

Handout / Flyer

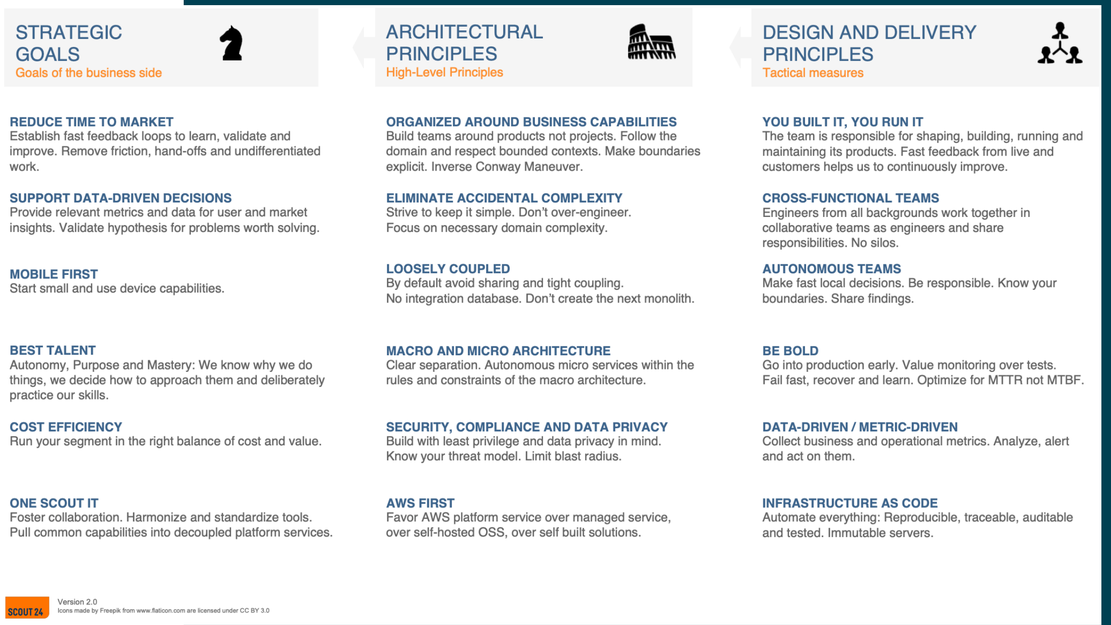
Separate Details zum nachlesen

(Code-)Beispiele

“Start small”

Institutionaliserte Prinzipien löschen

Prinzipien periodisch hinterfragen



STRATEGIC GOALS
Goals of the business side

REDUCE TIME TO MARKET
Establish fast feedback loops to learn, validate and improve. Remove friction, hand-offs and undifferentiated work.

SUPPORT DATA-DRIVEN DECISIONS
Provide relevant metrics and data for user and market insights. Validate hypothesis for problems worth solving.

MOBILE FIRST
Start small and use device capabilities.

BEST TALENT
Autonomy, Purpose and Mastery: We know why we do things, we decide how to approach them and deliberately practice our skills.

COST EFFICIENCY
Run your segment in the right balance of cost and value.

ONE SCOUT IT
Foster collaboration. Harmonize and standardize tools. Pull common capabilities into decoupled platform services.

ARCHITECTURAL PRINCIPLES
High-Level Principles

ORGANIZED AROUND BUSINESS CAPABILITIES
Build teams around products not projects. Follow the domain and respect bounded contexts. Make boundaries explicit. Inverse Conway Maneuver.

ELIMINATE ACCIDENTAL COMPLEXITY
Strive to keep it simple. Don't over-engineer. Focus on necessary domain complexity.

LOOSELY COUPLED
By default avoid sharing and tight coupling. No integration database. Don't create the next monolith.

MACRO AND MICRO ARCHITECTURE
Clear separation. Autonomous micro services within the rules and constraints of the macro architecture.

SECURITY, COMPLIANCE AND DATA PRIVACY
Build with least privilege and data privacy in mind. Know your threat model. Limit blast radius.

AWS FIRST
Favor AWS platform service over managed service, over self-hosted OSS, over self built solutions.

DESIGN AND DELIVERY PRINCIPLES
Tactical measures

YOU BUILT IT, YOU RUN IT
The team is responsible for shaping, building, running and maintaining its products. Fast feedback from live and customers helps us to continuously improve.

CROSS-FUNCTIONAL TEAMS
Engineers from all backgrounds work together in collaborative teams as engineers and share responsibilities. No silos.

AUTONOMOUS TEAMS
Make fast local decisions. Be responsible. Know your boundaries. Share findings.

BE BOLD
Go into production early. Value monitoring over tests. Fail fast, recover and learn. Optimize for MTTR not MTBF.

DATA-DRIVEN / METRIC-DRIVEN
Collect business and operational metrics. Analyze, alert and act on them.

INFRASTRUCTURE AS CODE
Automate everything: Reproducible, traceable, auditable and tested. Immutable servers.

SCOUT 24 Version 2.0
Icons made by Freepik from www.flaticon.com are licensed under CC BY 3.0

Birgitta Böckeler on Cultivating Architecture Principles

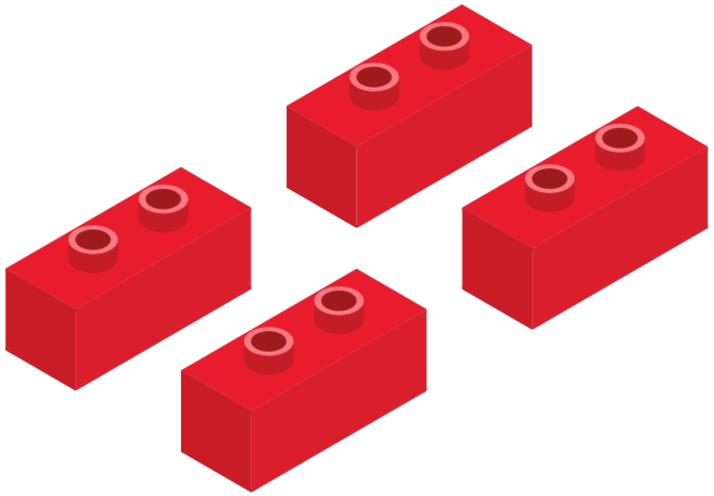
August 24, 2020

Birgitta Böckeler on Cultivating Architecture Principles
Sven Johann talks with Birgitta Böckeler about Cultivating Architecture …
00:00 1x

Download it: MP3 | AAC | OGG | OPUS

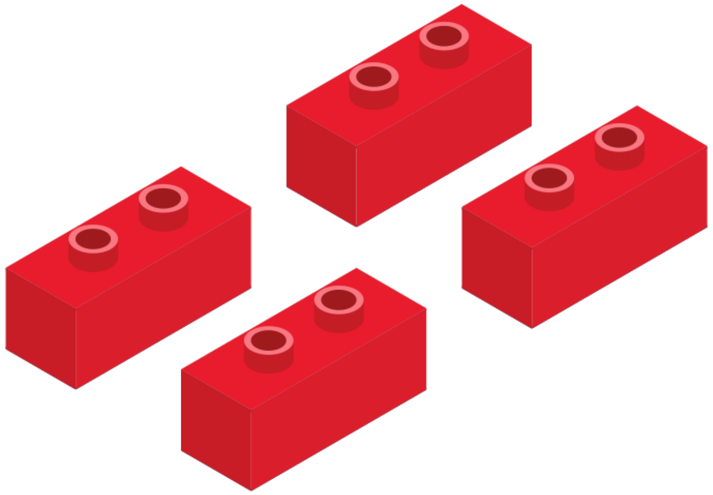
Makro-Architektur

Macro Architecture



> Decisions for all modules

Why Macro Architecture?



- All modules part of **one system**
- Goal: Minimal Macro Architecture
- Macro Architecture stable

Makro-Architektur

- Gilt zwingend für alle Services
- Stabil über langen Zeitraum
- Wirklich schwierig im Nachhinein zu ändern
- Limitiert Freiheiten der Teams

Beispiele

- Authentifizierung
- Frontend-Integration
- Programmiersprachen
- Container Runtime
- Service Discovery
- Log-Format

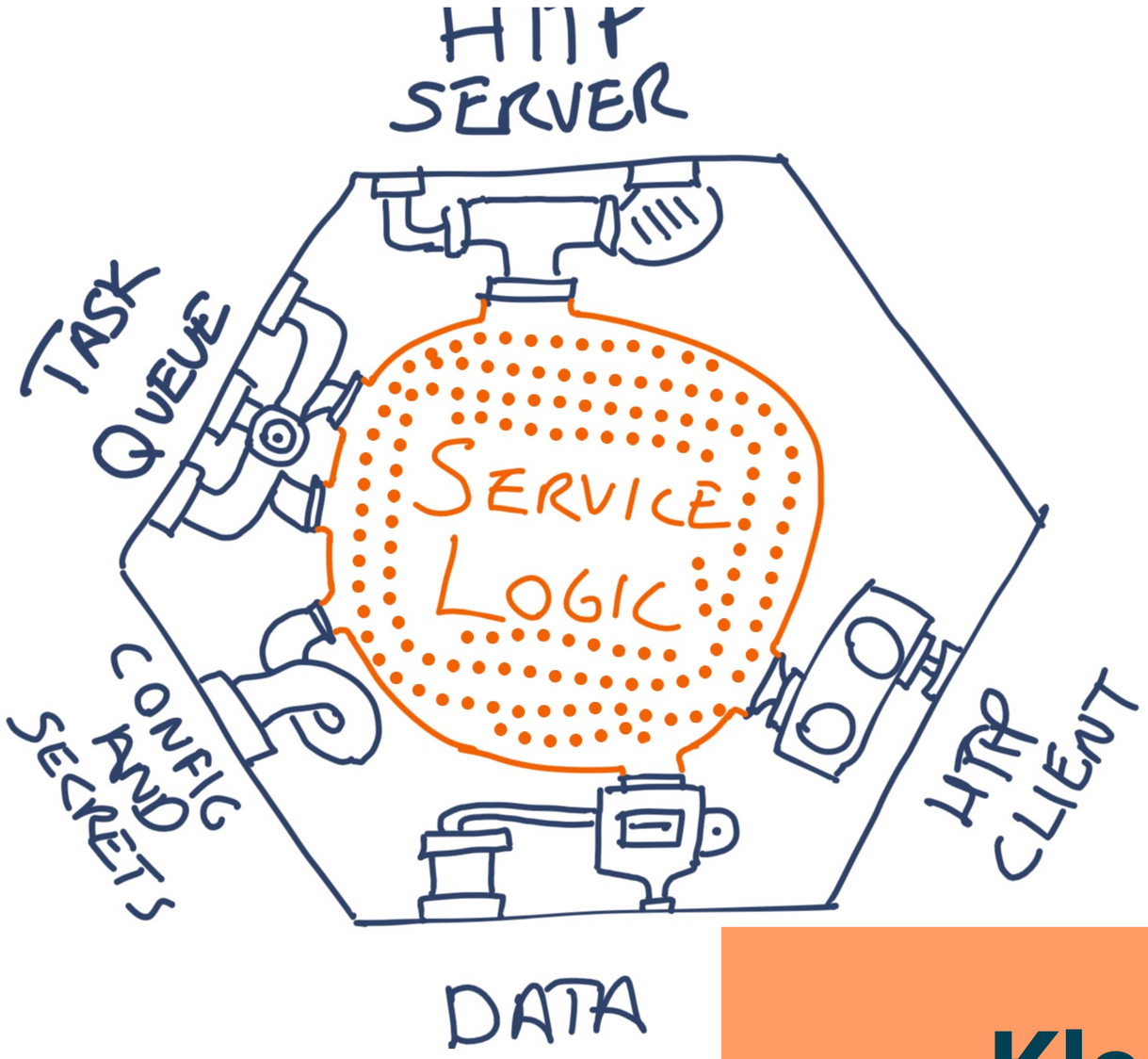
Prinzipien vs. Makroarchitektur

Follow the Trail

a.k.a Golden Path, The Paved Road, Service Templates / Chassis, Dancing Skeleton

Neuer Service/App: Was brauche ich?

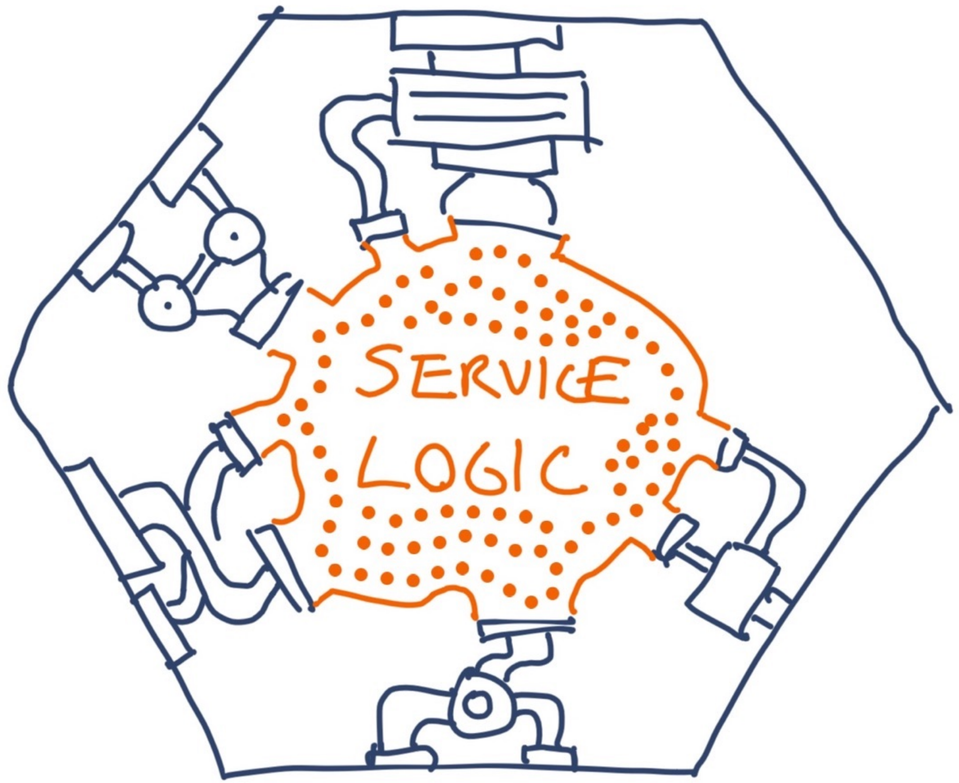
Programmiersprachen, Paketstruktur, Datenbank-Technologie, Persistenzimpl., Exception-Handling, Webframework, Template-Engine, I18N, Integration / Remote Process Communication, Build, Analyse, Deployment-Pipeline, Secret-Handling, Konfiguration, Testframeworks, Logging, Monitoring, Authentifizierung, Autorisierung, Feature Flags, Base-Image, Alerting, Service Discovery, Incident Management, Stream Processing, Tracing

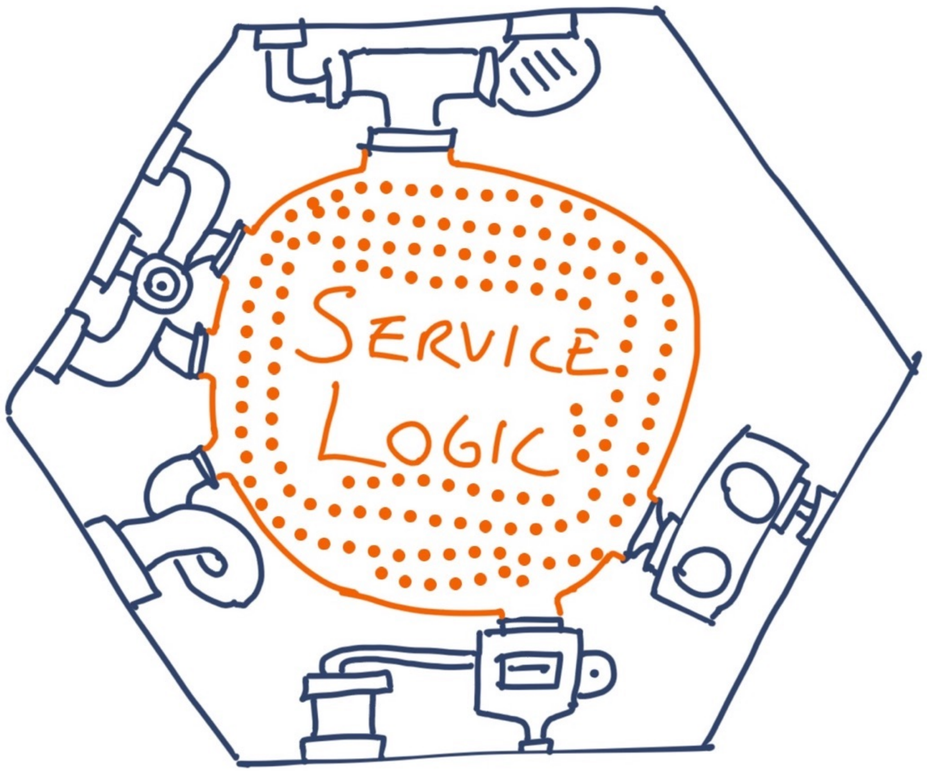


Klempnerei

https://blog.thepete.net/blog/2020/09/25/service-templates-service-chassis/

Alle Teams klempnern für sich selbst





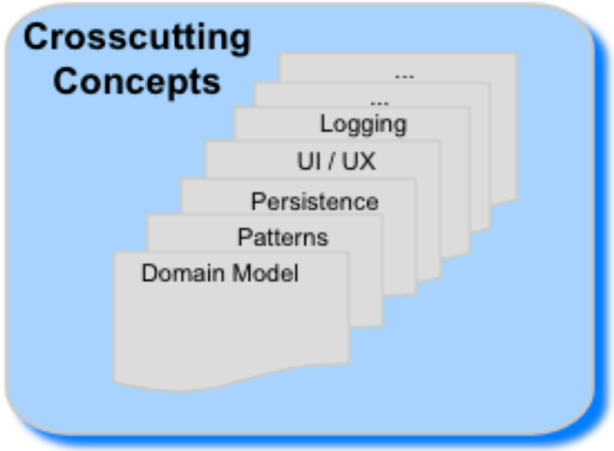
https://blog.thepete.net/blog/2020/09/25/service-templates-service-chassis/

Mögliche Lösung: arc42 Doku+Codeschnipsel

8. Querschnittliche Konzepte

Übergreifende, generelle Prinzipien und Lösungsansätze, die in vielen Teilen der Architektur einheitlich benutzt werden (→ cross-cutting). Konzepte beziehen sich oft auf **mehrere Bausteine**. Hier findet man Themen wie Domänenmodelle, Architekturmuster und -stile, Regeln zur Nutzung bestimmter Technologiestacks, etc.

mehr dazu ...



https://www.arc42.de/overview/

Sehr gut, aber...

https://unsplash.com/photos/green-trees-and-plants-during-daytime-3UayygJMXDI

Drift

- Kein Team bzw. Service wird dem Trail zu 100% folgen
- Der Trail entwickelt sich weiter, entweder durch den Anbieter des Trails oder die Teams (Nutzer)
- Anbieter: Wie bietet man immer den besten Trail an?
- Team: Wie weiß ich, dass es mittlerweile einen besseren Trail als den meinen gibt?

Produktdenke

Anzahl der Trails

- Biete ich nur Java mit Spring an?
- Biete ich Java, Go, Python und JavaScript mit unterschiedlichen Basisframeworks an?

Weitere Infos

- https://blog.thepete.net/blog/2020/09/25/service-templates-service-chassis/
- https://engineering.atspotify.com/2020/08/how-we-use-golden-paths-to-solve-fragmentation-in-our-software-ecosystem/
- https://www.case-podcast.org/49-chris-richardson-on-service-templates-and-service-chassis
- Kapitel “Governance and The Paved Road”, Sam Newman, Building Microservices, 2nd Edition

Fazit

“Autonomie” verlangt eigenständige Entscheidungen

Entscheidungen brauchen eine Orientierung, Prinzipien helfen

Makroarchitektur macht explizite Vorgaben (Freedom from Choice)

Follow the Trail nimmt uns jede Menge Entscheidungen ab

Wir können aber auch einen eigenen Weg gehen (Freedom of Choice)

Danke! Fragen?

Sven Johann
sven.johann@innoq.com

www.innoq.com

innoQ Deutschland GmbH

Krischerstr. 100
40789 Monheim
+49 2173 3366-0

Ohlauer Str. 43
10999 Berlin

Ludwigstr. 180E
63067 Offenbach

Kreuzstr. 16
80331 München

Hermannstrasse 13
20095 Hamburg

Erftstr. 15-17
50672 Köln

Königstorgraben 11
90402 Nürnberg