Agile Architecture
in a Nutshell

No more spaghetti code & Big Ball of Mud

No more isolated & Big Upfront Design

Architecture as a Structure
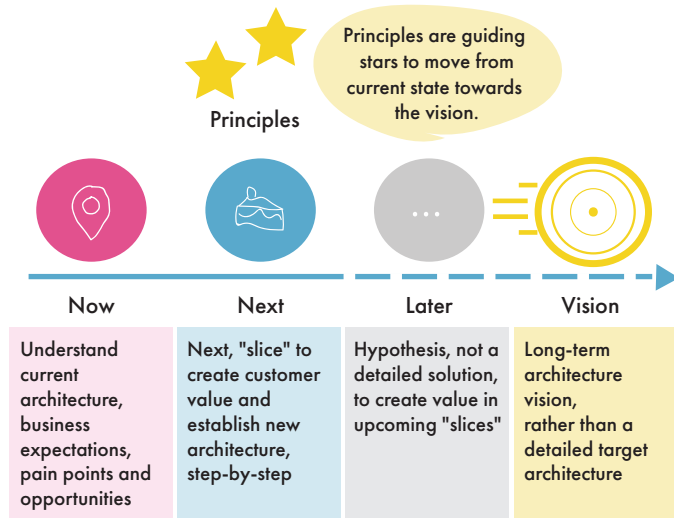The organization of a system, its sub-systems and their relations.

Agile Architecture
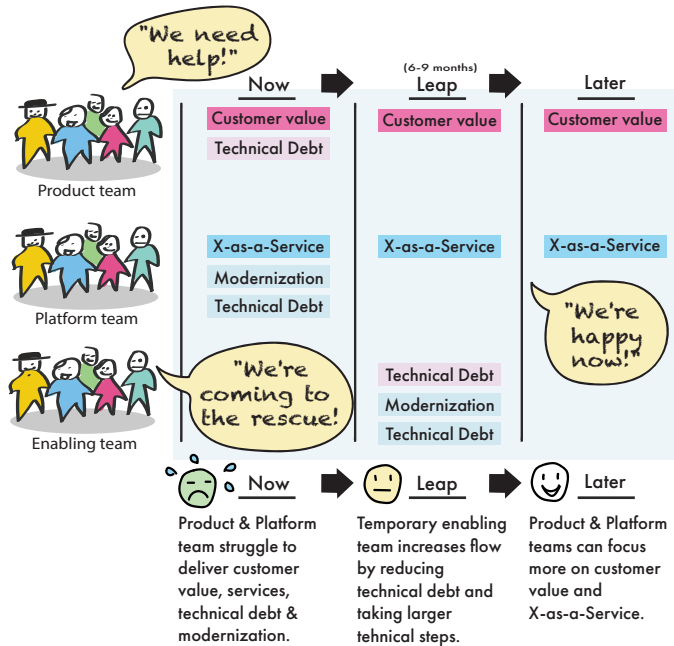Flexible structure created just in time to enable value flow.

DANDY PEOPLE

To Architect
In a step-by-step approach

Principles are guiding stars to move from current state towards the vision.

Principles

Now — Next — Later — Vision

Now: Understand current architecture, business expectations, pain points and opportunities

Next, "slice" to create customer value and establish new architecture, step-by-step

Later: Hypothesis, not a detailed solution, to create value in upcoming "slices"

Vision: Long-term architecture vision, rather than a detailed target architecture

Larger technical leaps

"We need help!"

Now → Leap (6-9 months) → Later

Product team
Now: Customer value / Technical Debt
Leap: Customer value
Later: Customer value

Platform team
Now: X-as-a-Service / Modernization / Technical Debt
Leap: X-as-a-Service
Later: X-as-a-Service

"We're coming to the rescue!"

Enabling team
Leap: Technical Debt / Modernization / Technical Debt

"We're happy now!"

Now → Leap → Later

Now: Product & Platform team struggle to deliver customer value, services, technical debt & modernization.

Leap: Temporary enabling team increases flow by reducing technical debt and taking larger tehnical steps.

Later: Product & Platform teams can focus more on customer value and X-as-a-Service.

Alignment

Team Assessment
Teams assess themselves based on shared values, principles & standards.

Coaching
Architect works close to the teams and educates and coaches them to increase flow through technology.

Self-service
The teams assess by using predefined questionnaires and get support from architect when needed.

Architecture Review Board
Try to avoid. Only for complex and unexplored architectural challenges. Must be solution oriented.

"We build green, not red!"

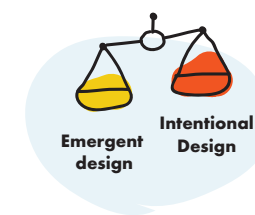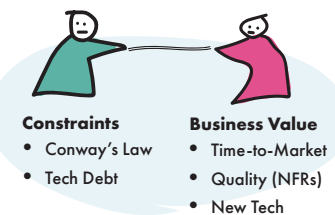C4 Model by Simon Brown • The BAPO Model by Jan Bosh

---

Architect to Enable Value Flow

Architecture is an art of continually improving value flow through technology, by balancing emergent and intentional design, and realizing business value while overcoming constraints in the current situation.

Without agile architecture, there will be no business agility.

"Organizations, who design systems, are constrained to produce designs which are copies of the communication structures of these organizations."

Constraints
• Conway's Law
• Tech Debt

Business Value
• Time-to-Market
• Quality (NFRs)
• New Tech

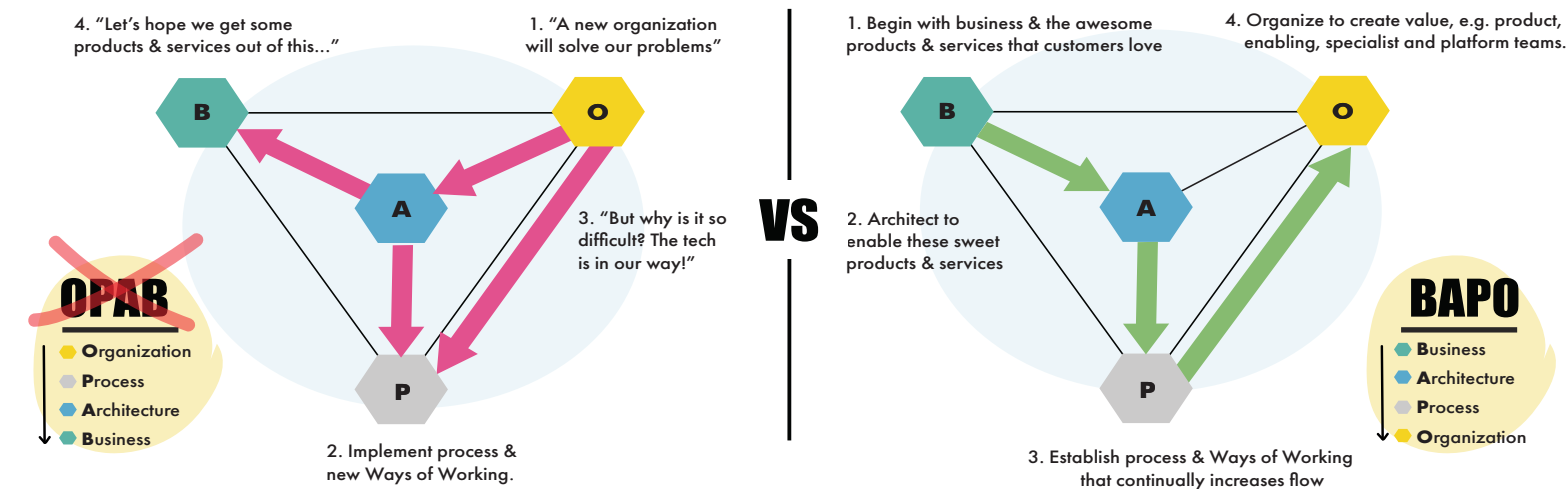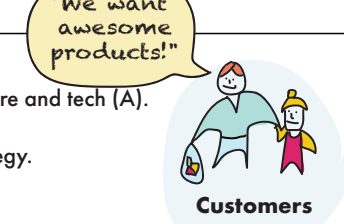Emergent design / Intentional Design

Emergent design can lead to short lead times initially, but disastrous consequences due to poor quality in the long run.

Intentional design can lead to long lead times and design detached from reality, due to too much upfront design.

Take a Business & Product First Approach

"We must re-organize!"

Managers

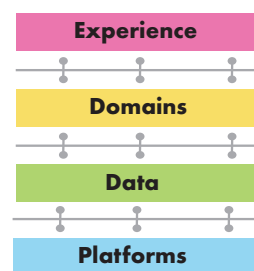"We want awesome products!"

Customers

The BAPO Model states that the business & product strategy (B) should drive architecture and tech (A). These should in turn drive process, ways of working & tooling (P). These should be used to define the organization (O) to realize the product and tech strategy.

4. "Let's hope we get some products & services out of this…"
1. "A new organization will solve our problems"
3. "But why is it so difficult? The tech is in our way!"

OPAB
• Organization
• Process
• Architecture
• Business

2. Implement process & new Ways of Working.

VS

1. Begin with business & the awesome products & services that customers love
4. Organize to create value, e.g. product, enabling, specialist and platform teams.
2. Architect to enable these sweet products & services

BAPO
• Business
• Architecture
• Process
• Organization

3. Establish process & Ways of Working that continually increases flow
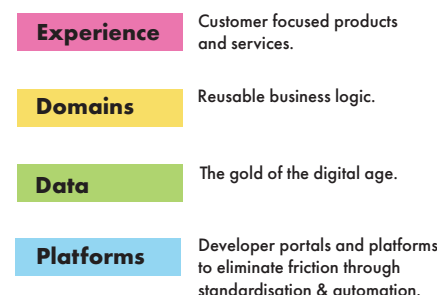
Adapt a Pace Layer Architecture

A Pace Layer Architecture is a technical stack that enables efficient development of digital products and services.

The layers encapsulates code that belongs together and is losely coupled to underlying complexity to ensure correct, efficient, reusable, configurable development.
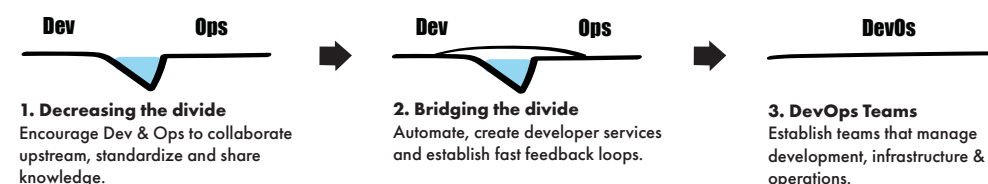
Experience — Domains — Data — Platforms

External APIs to develop digital products and services in an ecosystem.

Internal APIs to simplify data exchange internally and an enabler to be data driven

Experience: Customer focused products and services.

Domains: Reusable business logic.

Data: The gold of the digital age.

Platforms: Developer portals and platforms to eliminate friction through standardisation & automation.
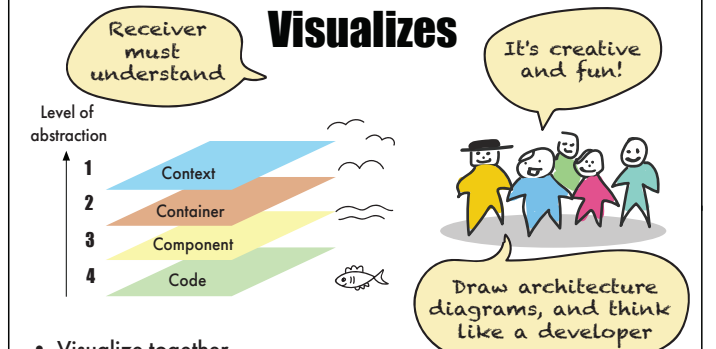
Encourage a DevOps Culture

An architect is an evangelist for a DevOps culture and continually contributes to decreasing the divide and friction between development and operations, by exploring new Ways of Working, standardization, automation and establishing DevOps teams.

Dev / Ops
1. Decreasing the divide
Encourage Dev & Ops to collaborate upstream, standardize and share knowledge.

Dev / Ops
2. Bridging the divide
Automate, create developer services and establish fast feedback loops.

DevOs
3. DevOps Teams
Establish teams that manage development, infrastructure & operations.

The Architect
Visualizes

Receiver must understand

It's creative and fun!

Level of abstraction
1 Context
2 Container
3 Component
4 Code

Draw architecture diagrams, and think like a developer

• Visualize together
• Create maps with different levels of details depending on the target group
• Use a whiteboard and a common notation, e.g. C4 Model
• Visualize "pain points" that inhibit flow first.

Creates a Toolbox

Design the toolbox based on the needs of the organisation and the teams.
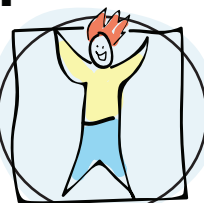
Principles	Shared rules and guidelines
Standards	Preferred tech, e.g. Tech Radar
Reference Models	Common language and concepts, e.g. C4 Model
Templates	Decisions, principles, visuals, etc.
Patterns	Pre-tested design and tech combinations
Implementation guides	Developer manuals (when needed)
Reusable components	Repository of services, components and source code, e.g. Design system

Is a Trusted Advisor

An architect is a trusted advisor for managers, product owners and developers in architecture.

A storyteller connecting the dots from business value to components and code, who:

• Is part of early stages of business strategy and planning.
• Coaches teams to make them able to architect themselves.
• Is a tech advisor for urgent and complex architecture challenges.
• Fosters an agile mindset and encourages continually improving flow, e.g. through DevOps.
• Has deep insight into business, products, cloud, tech, development and operations.
• Takes necessary decisions.

DANDY PEOPLE