

# Progressives Projektsetup

## Erfolgreiche Entwicklung in komplexen Systemlandschaften

Die zunehmende Beschleunigung von Veränderungen in Unternehmen ist ein Fakt, welcher bekanntermassen immer bedeutenderen Einfluss auf den Erfolg und die Profitabilität von Unternehmen hat. Dabei spielt die immer stärkere Digitalisierung und Integrierung der Systeme eine tragende Rolle. Der Wunsch nach Individualisierung und der Möglichkeit kurzfristiger Applikationsanpassungen ist signifikant gestiegen.

Heute werden Softwarelösungen zunehmend in kleinere Komponenten unterteilt, so dass sich spezialisierte Teams stärker fokussieren können und somit eine grössere Expertise wie auch kürzere Reaktionszeit bei der Priorisierung neuer Anforderungen gewährleisten. Diese Produktteams werden in den meisten Organisationen in die bestehende Hierarchie eingegliedert. Ziel ist eine Verbesserung der Stabilität, Planbarkeit und Effizienz.

Durch das steigende Bedürfnis nach integrierten Systemen ist ein zunehmend komplexeres Zusammenspiel vieler Softwarekomponenten zu sehen, was in Softwareprojekten wiederum einen wachsenden Bedarf an Koordination aller involvierten Teams erfordert.

**Doch wie können in Zukunft Projekte erfolgreich realisiert werden, bei welchen das Zusammenspiel diverser Expertenteams stetig komplexer wird?**

In klassischen Projektorganisationen haben Projektleiter oft die ganzheitliche Verantwortung inne: Aufgaben zu planen und Lösungen zu realisieren. Dieser Ansatz funktioniert, solange die Projektorganisation die vollständige Lösung eigenständig bereitstellen kann.

Sobald weitere Produktteams kritische Komponenten für eine Lösung bereitstellen müssen, stellen sich jedoch neue Herausforderungen:

- Regelung der Verantwortlichkeiten
- Koordinativer (nicht produktiver) Aufwand
- Priorisierung der Aufgaben in Produktteams

Diesen Herausforderungen wird ein klassischer Projektansatz immer weniger gerecht. Ein Beispiel ist die Illusion, dass ein zentraler PL die Verantwortung für Lieferungen diverser Produktteams übernehmen kann.

Zentral kann eine koordinative Leitung übernommen werden. Die Priorisierung und Zusage zu einem Umsetzungszeitpunkt kann jedoch ausschliesslich aus den einzelnen Produktteams selbst erfolgen.

### Beispiel einer komplexen Systemlandschaft

Zur Illustration eines komplexen Projektsetups soll das folgende vereinfachte Beispiel einer Anbindung eines Drittsystems an zwei bestehende Backendsysteme dienen. Dabei sprechen wir von einem Programm mit drei Substreams für die Gesamtlösungsbereitstellung.

- Stream 1: Infrastrukturbereitstellung & Betrieb
- Stream 2: Softwareentwicklung (Integration)
- Stream 3: Migration der Daten

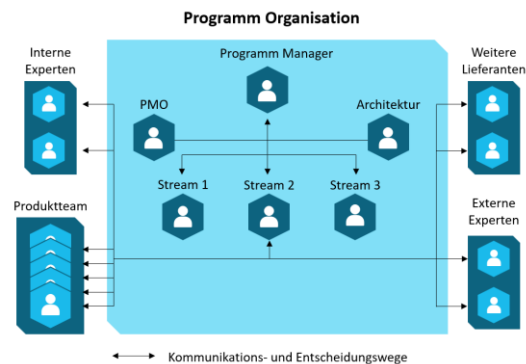


Abbildung 1: Programm-Organisation

Die Ziellösung kann vereinfacht folgendermassen dargestellt werden.

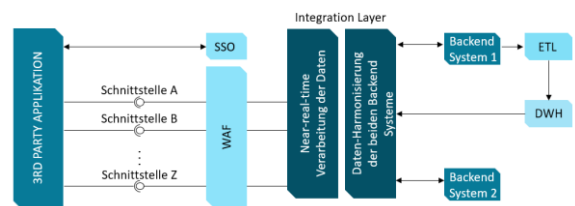


Abbildung 2: mögliches Zielbild der Softwarelösung

Im zentralen Team sind 6-7 Personen beteiligt, welche das Programm koordinieren (Programmleitung, Substream-Leads und Architektur). Weiter werden diverse Komponenten für die Lösungsbereitstellung benötigt, was zusätzlich 5-8 Produktteams umfasst.

Wie erwähnt, stellt die Koordination der diversen involvierten Produktteams eine der Kernherausforderungen komplexer Softwarelösungen dar. Aus dem zentralen Projektteam werden Anforderungen formuliert und deren Abhängigkeiten transparent nachgeführt. Die Produktteams ihrerseits kommunizieren mögliche Lösungen und deren Bereitstellungszeitpunkt zurück ans Projekt.

Um das Risiko eines «Flaschenhals» zu reduzieren (z.B. bei der Projektleitung), soll eine Verantwortungsaufteilung anhand von Rollen in drei Autoritätsbereiche realisiert werden:

- Governance Autorität
- Content Autorität (fachlicher Inhalt)
- Design Autorität (technische Lösung)

**Governance Autorität**

Diese Rollen übernehmen die Verantwortung für die koordinativen Aufgaben, das Projekt Reporting, sauberes Etablieren notwendiger Prozesse und das Beseitigen von Impediments (Hindernisse).

Zielgerichtete Strukturen und Reports auf Programm Level sind äusserst zentral und umfassen mindestens die folgende Informationen:

- Governance Chart und Projektplan (Milestones)
- Offene fachliche Themenbereiche (Features)
- Offene erwartete/geschätzte Zeitaufwände
- Angelaufene Kosten / Aufwände
- Abhängigkeiten und Risiken
- Testreports (automatisiert und periodisch)

**Content Autorität**

Diese Rollen übernehmen die Verantwortung über die inhaltlichen Anforderungen. Sie beschreiben, was für fachliche Funktionalität bereitgestellt werden soll, welche Kriterien diese zu erfüllen haben, priorisieren die Aufgaben und übernehmen die fachliche Abnahme. Diese Rolle erfüllt typischerweise der Product Owner (PO) oder der Product Manager (Skalierungsrolle).

**Design Autorität**

Diese Rollen übernehmen die Verantwortung, wie die technische Lösung realisiert werden soll. Darunter fallen unter anderem Architektur-, Technologie- und Framework-Fragestellungen. Die gebaute Lösung muss nicht nur die fachlichen Anforderungen, sondern auch die nicht-funktionalen Anforderungen z.B. an Sicherheit und Reaktionszeit erfüllen und für den anschliessenden Betrieb wie auch für die Weiterentwicklung die notwendigen Voraussetzungen abdecken.

**Über diese Separierung der drei Autoritäten erzielt man einen Rollenfokus** und schafft eine Situation, in der sich die unterschiedlichen Rollen gegenseitig in die Pflicht nehmen. Dabei erfüllt die transparente Informationsbereitstellung der Governance-Rollen eine zentrale Aufgabe und unterstützt die Rollen der beiden anderen Autoritäten bei der Erfüllung ihres Auftrags.

Im Weiteren gehen wir darauf ein, wie die drei Autoritäten die Koordination der Produktteams aus dem Projektteam heraus signifikant vereinfachen können, und welche Vorteile damit erzielt werden.

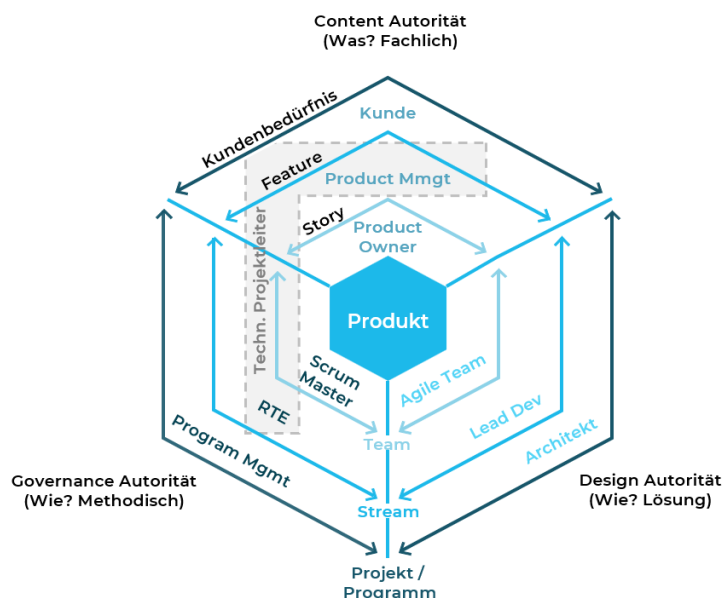


Abbildung 3: Separierung der Verantwortlichkeiten anhand der Autoritäten

Gehen wir nun gemäss Abbildung 1 davon aus, dass die Projektorganisation die Programmleitung (Governance) sowie Substream-Leads umfasst (verantworten als «Techn. Projektleiter» mindestens die Governance-Rolle, oft auch die Content-Rolle «Product Manager», siehe Abbildung 3). Die Architektur sowie ein erfahrener Lead-Entwickler werden spezifisch hinzugezogen für das Lösungsdesign sowie die Qualitätssicherstellung der ganzheitlichen, technischen Lösung.

Des Weiteren bestehen alle Produktteams aus einem Scrum Master (SM), Product Owner (PO) und dem Entwicklungsteam (Dev Team). Wie erwähnt, sind diese Produktteams in der Regel in einer hierarchischen Organisation eingebunden. Diese Ausgangslage erschwert ein reibungsloses Zusammenspiel zwischen Projekt und Produktteams, da die Produktteams in den Konflikt geraten,

1. langfristige Planungssicherheit, Effizienz und hohe Qualität bezüglich Anforderungen aus der Linienorganisation sicher zu stellen, vs.
2. kurzfristige Anpassungen für ein laufendes Kundenprojekt zu realisieren.

### Progressives Projektsetup

Im Folgenden ein Vorgehensansatz, welcher die benannten Herausforderungen adressiert.

#### 1. Projektvision und Backlog

Bei Projektstart wird eine Projektvision erarbeitet, sowie ein Produkt-Backlog mit dem bekannten Projektumfang. Dieser soll frühzeitig mit dem Kunden abgestimmt werden, um den Projektumfang einschätzen zu können.

#### 2. Value-Stream Identifikation

Der Produkt-Backlog dient als Basis für die anschliessende Value-Stream<sup>1</sup> Analyse. Dabei werden alle fachlichen Anforderungen aufgeführt, sowie eine Zuordnung auf die dazu benötigten Software-Komponenten erstellt. Anschliessend wird eruiert, welche Produktteams für diese Komponenten verantwortlich sind, sowie ob weitere Teams wie z.B. externe Lieferanten einbezogen werden müssen.

#### 3. Projektteam Setup

Das Projektteam ist eine Gruppe von Personen, welche Rollen aus den Bereichen Governance und Content übernehmen. Sie definieren das Projektvorgehen und dokumentieren, strukturieren und priorisieren die Anforderungen des Kunden.

Das Projektteam agiert dabei nicht aus der Linienorganisation heraus, sondern als losgelöstes Netzwerk. Dies erlaubt eine möglicherweise in Silos separierte Organisation zentral zusammenzuführen.

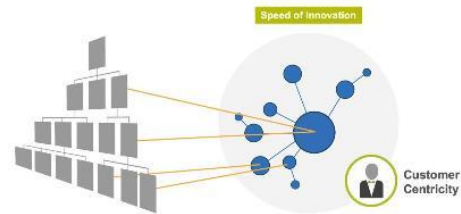


Abbildung 4: Projektteam als Netzwerk (Quelle: XLR8, Kotter)

#### 4. Produktteams

Die Produktteams werden aus dem Projektteam heraus einbezogen und damit zentral vernetzt. Sie stellen einen PO als Repräsentanten der Komponente, welcher die Content Verantwortung für die notwendigen Anpassungen seines Teilbereichs übernimmt. Neue Anforderungen müssen aus dem Projektteam heraus aktiv in die Programm-Inkrementen (PIs) und die Sprintplanung der Produktteams eingebracht werden. Dies kann anhand eines wöchentlichen Meetings stattfinden, in welchem alle POs und die Projektleitung sich zu offenen Aufgaben austauschen sowie Risiken und Abhängigkeiten aufgezeigt werden. Die POs übernehmen aus diesem Meeting die Verantwortung, dass ihre Komponente(n) die notwendigen Anforderungen des Projekts erfüllen, oder weisen auf erkannte Risiken aktiv hin. Damit kann sichergestellt werden, dass Abhängigkeiten, Risiken und Verzögerungen frühzeitig erkannt, adressiert und blockierende Situationen im besten Fall vermieden werden können.

#### Die Vorteile zusammengefasst

In diesem Setup nutzen wir die Vorteile des Projekt-Netzwerks, **um fokussiert auf spezifische Kundenwünsche eingehen zu können, während Produktteams aus möglichen Organisationssilos zentral eingebunden und koordiniert werden.** Die Kombination aus Effizienz, Planbarkeit und Qualität der Produktteams (z.B. PI- und Sprint-Kadenz) mit der Flexibilität und Dynamik der Projektorganisation ermöglicht, Verantwortlichkeiten besser zwischen POs und Projekt aufzuteilen. So können die POs nicht nur die Verantwortung ihrer Erweiterungen, sondern das Zusammenspiel ihrer Module mit der Gesamtlösung übernehmen. Zentral ist ein inkrementelles Bereitstellen von Erweiterungen, wobei nach jeder Iteration lauffähige Software dem Kunden zur Verfügung gestellt werden soll.

<sup>1</sup> Value-Streams stellen eine Reihe von Abläufen über diverse Teams hinweg dar, anhand welcher eine Organisation eine Lösungen realisiert.