

ZUSAMMENFASSUNG

Lernziele

- Sie verstehen es Projekte erfolgreich zu starten, durchzuführen und abzuschliessen
- Sie kennen unterschiedliche Vorgehen von agil bis zum klassischen Wasserfall
- Sie beherrschen die wichtigsten Führungsinstrumente eines Projektleiters
- Sie besitzen grundlegende Kenntnisse für Management und Sicherung von Qualität in Projekten und Unternehmen
- Sie erlangen die weltweit anerkannte Zertifizierung «IPMA Level D» (freiwillig)

Lerninhalte

Diese Vorlesung ist ein Muss für alle Ingenieurinnen und Ingenieure, welche sich das Rüstzeug für Projektmanagement holen möchten. Im Vordergrund steht die Methodik zur Abwicklung von zeitlich begrenzen Vorhaben. In den praxisorientierten Übungsstunden lernen Sie Beispiele und Techniken aus der Industrie kenne. Folgender Inhalt im Detail:

- Was versteht man unter einem Projekt
- Projekttypen und deren interne Organisation
- Projektkontrolle
- ITIL als Qualitätsinstrument
- Klassische Projektvorgehensmodelle
- Agile Projektvorgehensmodelle
- Finanzierung von Projekten
- Umgang mit Projektrisiken
- Qualitätsmanagement/-sicherung
- Kommunikation mit wichtigen Anspruchsgruppen
- Konfigurationsmanagement
- Methoden zur Lösungsfindung

PROJEKT UND QUALITÄT	7
ZIELE DES KAPITELS	7
INNOVATION IST TEAMWORK	7
WAS IST EIN PROJEKT	7
<i>Typische Merkmale eines Projektes</i>	7
<i>Das magische Dreieck</i>	7
<i>Typische Projektarten in der IT</i>	8
<i>Projektgrösse</i>	8
<i>Make or Buy</i>	9
WAS BEDEUTET QUALITÄT	9
<i>Begriff «Qualität»</i>	9
<i>Begriff «Qualitätsmanagement»</i>	9
<i>Qualitätskreislauf nach Deming</i>	9
<i>Beispiel eines Qualitätsplans in einem Projekt</i>	10
PROJEKTE INITIALISIEREN, DURCHFÜHREN UND ABSCHLIESSEN	11
ZIELE DES KAPITELS	11
UNTERNEHMENSZIELE	11
PROJEKTTYPEN	11
WIE WERDEN PROJEKTE GESTARTET, WIE BEENDET	12
<i>Der Projektstart</i>	12
<i>Vorgehen bei der Planung</i>	13
<i>Projekteende/Projektabschluss</i>	15
PROJEKTEAM / ORGANISATION	15
<i>Mögliche Organisationen</i>	15
<i>AKV definieren</i>	15
<i>Gruppendynamik positiv umsetzen</i>	15
REQUIREMENTS ENGINEERING	16
<i>Agil oder Wasserfall – Auswirkungen auf die Anforderungen</i>	16
PROJEKTKONTROLLE	17
ZIELE DES KAPITELS	17
WER KONTROLLIERT DIE PROJEKTE	17
BERICHTSWESEN	17
<i>Welches Projektcontrollsyste mnehme ich?</i>	17
<i>Messwert und Vermessen</i>	18
<i>Effektiver Verlauf in den Projekten</i>	18
<i>Wie messe ich den genauen Fortschritt?</i>	18
<i>ISO 9126 – Sicherstellung der Softwarequalität</i>	18
<i>Das Ampel-Prinzip</i>	19
PROJEKTOCKPIT	19
STANDARDS FÜR DIE PROJEKTKONTROLLE	19
EARNED-VALUE-ANALYSE	20
<i>Wie berechne ich den Earned Value (Fertigstellungsgrad)?</i>	20
<i>Beispiel einer EVA an einem Studentenprojekt</i>	20
MEILENSTEIN-TRENDANALYSE	21
KLASSISCHES PROJEKTMANAGEMENT	21
ZIELE DES KAPITELS	21

VORGEHEN	21
<i>Vorgehensmodelle</i>	21
<i>Das Vorgehen</i>	21
<i>Klassische Vorgehensmodelle</i>	22
<i>Phasen am Beispiel RUP</i>	22
<i>CYNEFIN – Wann ein Phasenmodell</i>	22
HERMES, VORGEHENSMODELL DES BUNDES	23
<i>Phasen</i>	24
<i>Rollenverteilung bei HERMES</i>	24
<i>Aufgaben der Projektsteuerung</i>	24
<i>Aufgaben der Projektführung</i>	24
<i>Alle Projektteilnehmer nach HERMES (standardisiert)</i>	24
<i>Pro Projektteilnehmer ein definiertes AK(F)</i>	25
<i>Jede Aufgabe ist bis ins Detail beschrieben</i>	25
<i>Vor- und Nachteile von HERMES</i>	25
PROTOTYPENMODELL	25
<i>Merkmale</i>	25
SPIRALMODELL	26
DISKUSSION	26
FINANZIERUNG, KOSTEN UND BESCHAFFUNG	27
<i>ZIELE DES KAPITELS</i>	27
<i>KOSTEN</i>	27
<i>NUTZEN</i>	28
<i>Nutzelemente von Projekten</i>	28
<i>ROI</i>	28
<i>ROI Beispiel – Entwicklung einer neuen App</i>	29
<i>ROI Beispiel – WLAN Access Points ersetzen</i>	29
<i>Diskontierung – Was ist das?</i>	30
<i>ROI Beispiel – Online Shop einer Handelskette</i>	30
<i>BUSINESS CASES</i>	30
<i>FINANZIERUNG UND LIQUIDITÄT</i>	31
<i>Projektliquidität</i>	31
<i>BESCHAFFUNG UND VERTRÄGE</i>	32
<i>Was ist ein Vertrag?</i>	32
<i>Unterscheidung der Verträge für Dienstleistungen</i>	32
IT SERVICE MANAGEMENT NACH ITIL V3	33
<i>EINLEITUNG</i>	33
<i>Motivation</i>	33
<i>Historie</i>	33
<i>Ziele von IT Service Management</i>	33
<i>ITIL Übersicht v3</i>	33
<i>SERVICE DESIGN</i>	34
<i>Business Nutzen</i>	34
<i>Service Management</i>	34
<i>Key Performance Indikatoren</i>	34
<i>Capacity Management</i>	35
<i>Availability Management</i>	35
<i>SERVICE TRANSITION</i>	35
<i>Transition Planning & Support</i>	35

<i>Change Management</i>	35
<i>Release & Deployment Management</i>	35
<i>Service Asset & Configuration Management</i>	36
<i>Knowledge Management</i>	36
SERVICE OPERATION	36
TIPPS AUS DER PRAXIS	37
<i>Für konventionelle Projekte</i>	37
<i>Für agile DevOps Teams</i>	37
RISIKOMANAGEMENT	38
ZIELE DES KAPITELS	38
ANALYSE	38
<i>Beispiele aus der Schweiz</i>	38
<i>Identifizieren</i>	38
<i>Einschätzen</i>	39
<i>Einschätzungsbeispiel «Serverausfall»</i>	39
BEWERTUNG	39
<i>Risikoliste und Risikograph</i>	40
<i>20 Fehler nach Lindecker (Software Projects, 1989)</i>	40
BEWÄLTIGUNG	40
<i>Beispiel von Massnahmen</i>	40
RISIKEN AUS DER PRAXIS	41
<i>Top-Ten Risiken einer Software-Entwicklung</i>	41
<i>Risikobasiertes Vorgehen für ein nicht konfiguriertes Produkt</i>	41
<i>Risikobasiertes Vorgehen für ein konfiguriertes Produkt</i>	41
<i>Risikobewertung für eine kundenspezifische Applikation</i>	41
CHANCEN	41
AGILES PROJEKTMANAGEMENT I	42
ZIELE DES KAPITELS	42
MOTIVATION FÜR AGILITÄT	42
AGILES MANIFEST	43
<i>Zwölf Prinzipien</i>	43
BESTANDTEILE EINES AGILEN VORGEHENS	44
<i>Inspect & Adapt</i>	44
<i>Transparenz</i>	44
<i>Selbstorganisation</i>	44
<i>Timeboxing</i>	44
HEUTE VERWENDETE AGILE METHODEN	45
AGILES PROJEKTMANAGEMENT II	46
ZIELE DES KAPITELS	46
SCRUM FRAMEWORK IN DER TIEFE	46
<i>Die Elemente</i>	46
<i>Fundament von Scrum</i>	46
<i>Scrum ist eine empirische Methode</i>	47
<i>Unterscheidungen</i>	47
<i>«Verunreinigung» von Scrum</i>	48
SEMI-AGIL MIT KANBAN	48
<i>Was ist Kanban (in der IT)?</i>	48
<i>Nutzen</i>	48

<i>Woher kommt Kanban?</i>	48
<i>Kommentar.....</i>	49
<i>Weitere agile Methoden.....</i>	49
QUALITÄTSMANAGEMENT	50
ZIELE DES KAPITELS	50
QS IM PROJEKT	50
<i>Das V-Modell im Bestreben einer Früherkennung</i>	50
<i>Verfahren zur Qualitätssicherung in Softwareprojekten.....</i>	50
<i>Beispiel Ad Absurdum.....</i>	51
CHANGE-MANAGEMENT IM PROJEKT	51
<i>Kategorien von Changes Request (I)</i>	51
<i>Voraussetzungen eines Changes (II).....</i>	52
<i>Beschreibung eines Changes (III)</i>	52
<i>Auswirkungen und Risiken (IV)</i>	52
<i>Change quantifizieren (V).....</i>	52
<i>Vorgehen.....</i>	52
QS IM UNTERNEHMEN MIT SIXSIGMA	53
<i>Vergleich klassisches QS mit SixSigma.....</i>	53
<i>Was bedeutet Sigma genau.....</i>	53
<i>SixSigma – 5 stetige Verbesserungsschritte.....</i>	53
KONFIGURATIONSMANAGEMENT	54
<i>Wozu braucht es Konfigurationsmanagement?.....</i>	54
<i>Releases und ihre Versionsbezeichnung.....</i>	54
<i>Vorteile eines guten Tools.....</i>	55
<i>ITIL.....</i>	56
<i>Berechtigungen in SCM.....</i>	56
KOMMUNIKATION UND DOKUMENTATION.....	57
ZIELE DES KAPITELS	57
STAKEHOLDER.....	57
<i>Sechs Fragen um Stakeholder zu finden</i>	57
<i>Gehört das Projektteam zu den Stakeholdern?</i>	57
FÜHRUNG / RESSOURCENMANAGEMENT	58
<i>Führungsrythmus</i>	59
<i>Ressourcenmanagement.....</i>	59
KOMMUNIKATION	59
<i>Projekt- und Organisationsanalyse</i>	59
<i>Qualität vor Quantität.....</i>	60
<i>Kommunikation nach Schulz von Thun.....</i>	60
PROJEKTMARKETING	60
<i>Der Projektleiter als Botschafter</i>	60
DOKUMENTATION	61
LÖSUNGSFINDUNG UND TEAMARBEIT	62
ZIELE DES KAPITELS	62
VERSCHIEDENE METHODEN ZUR LÖSUNGSFINDUNG	62
<i>Finden Brainstorming.....</i>	62
<i>Finden Methode 653.....</i>	63
<i>Finden Morphologischer Kasten</i>	63
<i>Bewerten Nutzwertanalyse</i>	64

<i>Bewerten Entscheidungsbaum.....</i>	64
REVIEWTECHNIK	65
SITZUNGSTECHNIK.....	65
<i>So agieren Sie als Sitzungsleiter.....</i>	66
<i>Die Teilnehmer einer Sitzung</i>	66
<i>Führungsrythmus</i>	66
VERHANDLUNGSTECHNIK	67
<i>Verhandlungskommunikation</i>	67
<i>Die Harvard-Methode 5 Grundsätze.....</i>	67
<i>Kompromiss vs. Konsens.....</i>	67
<i>Konflikt-Eskalation nach Friedrich Glasl.....</i>	67
PROJEKTE IN DER PRAXIS.....	68
ZIELE DES KAPITELS	68
PROJEKTZIELE/-ABGRENZUNG	68
PROJEKTANTRAG/-AUFTRAG.....	68
PROJEKTSTRUKTUR (OBJEKT-,AKTIVITÄT, PHASENORIENTIERUNG).....	69
GANTT-/NETZPLAN	69
PHASEN / SPRINTS.....	70
MEILENSTEINE.....	71
RESSOURCENPLANUNG / AKV.....	71
KICKOFF VERANSTALTUNG	72
<i>Beispiele</i>	72
<i>Zeitplanung typisches Kickoff.....</i>	72
<i>Wortschatz Kickoff.....</i>	72
TYPISCHE LIEFERERGEBNISSE	72
EINFÜHRUNG, BIG BANG	73

Projekt und Qualität

Ziele des Kapitels

Lerninhalte	Ziele: Die Studierenden...
<ul style="list-style-type: none"> - Was ist ein Projekt - Was bedeutet Qualität 	<ul style="list-style-type: none"> - erarbeiten sich selbstständig Beispiele von Innovation - erkennen ein Projekt aufgrund von Merkmalen - verstehen was Qualitätsmanagement ist

Innovation ist Teamwork



Die Schweiz ist gemäss dem WEF das Wettbewerb fähigste und gemäss GII das innovativste Land der Welt. Viel Innovation entsteht aus Projekten.

Aber was genau ist Innovation?
Lösungen, welche neue durch Organisation und neue Technologien getrieben sind.

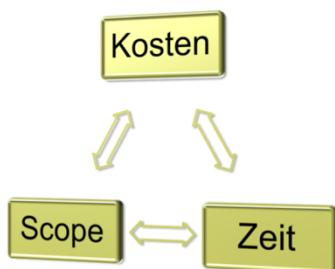
Beispiele dafür sind: Facetune App, Zipline Dronen, ...

Was ist ein Projekt

Typische Merkmale eines Projektes

- Zeitlich begrenzt
- Beschränkte Ressourcen (Zeit, Geld, etc.)
- Neue Problemstellung (Neue Technologie, Neue organisatorische Aspekte, Beide in Kombination = Innovation)
- Ungewissheit aufgrund komplexer Herausforderungen
- Benötigt Personen aus verschiedenen Organisationseinheiten

Das magische Dreieck



Das magische Dreieck zeigt die Abhängigkeiten in einem Projekt. Kosten beeinflussen Zeit und Scope, Scope wiederum Kosten und Zeit und die Zeit Kosten und Scope. Folgend werden ein paar Beispiele von Konflikten genannt.

- **Zeit ↔ Kosten**
 - o Ein fester Endtermin für ein Entwicklungsprojekt kann unter Umständen nicht mehr gehalten werden (Möglichkeit: Mehr Entwickler einsetzen um den Termin zu halten, was aber mehr Kosten verursacht)

- Kosten ⇔ Scope

- Das Budget ist fast aufgebraucht und eigentlich besteht noch der Wunsch das Webfrontend mit Logo und CO des Kunden auszustatten. (Möglichkeit: Webfrontend im Standard stehenlassen um Geld zu sparen)

- Zeit ⇔ Scope

- Die Applikation ist fast abgeschlossen und der Projektleiter würde aber gerne noch automatische Performance Tests über zwei Woche sowie einige Tuning Arbeiten ausführen (Möglichkeit: Je nach Interesse Performance Tests durchführen und dafür Verspätung in Kauf nehmen oder darauf verzichten)

Typische Projektarten in der IT

Projekt Neuentwicklung, Projekt Weiterentwicklung, Integration einer gekauften Software, Aufbau einer Infrastruktur, Forschungsprojekt, Organisationsprojekt.

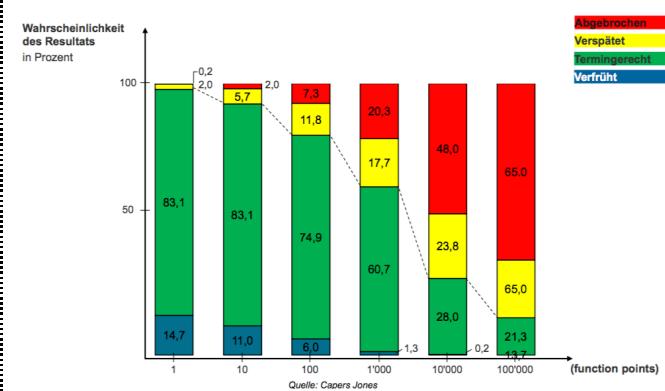
Projektgrösse

Die Grösse eines Projekts lässt sich über verschiedene Faktoren messen. Kosten, Zeit oder Arbeitsstunden. Auch möglich ist der akademische Ansatz von Capers Jones. Die Berechnung (<http://groups.engin.umd.umich.edu/CIS/course.des/cis375/projects/fp99/main.html>) beruht dort auf Functions Points.

Diese Abbildung zeigt einige Beispiele von Projektgrössen marktüblicher Software (angegeben in Functions Points).

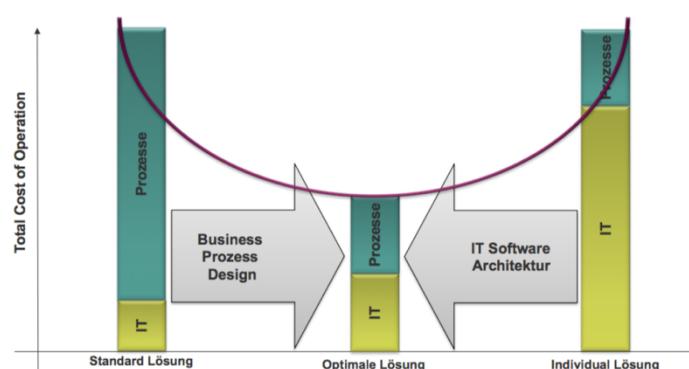
1. Oracle	229,434
2. Windows 7 (all features)	202,150
3. Microsoft Windows XP	66,238
4. Google docs	47,668
5. Microsoft Office 2003	33,736
6. F15 avionics/weapons	23,109
7. VA medical records	19,819
8. Apple I Phone	19,366
9. IBM IMS data base	18,558
10. Google search engine	18,640
11. Linux	17,505
12. ITT System 12 switching	17,002
13. Denver Airport luggage (original)	16,661
14. Child Support Payments (state)	12,546
15. Facebook	8,404
16. MapQuest	3,793
17. Microsoft Project	1,963
18. Android OS (original version)	1,858
19. Microsoft Excel	1,578

Je grösser das Projekt, je geringer die Wahrscheinlichkeit eines guten Resultats.



Make or Buy

In Projekten muss man irgendwann immer einmal entscheiden, ob man es selbst macht oder kauft. Bei einer Standardlösung müssen meist die Prozesse angepasst werden, während bei der individuellen Lösung der Aufwand an IT höher liegt. Die optimale Lösung wäre eine ausgeglichene Variante in der Mitte.



Resultat

Wie die Entscheidung fällt ist je nach Person abhängig für einen Hersteller von Velo-Computer würde ich sagen, dass diese Firma alle folgenden Sachen kaufen/outsourcen soll: Webseite, Zeiterfassungssoftware, ERP, CRM, Schliesssystem Gebäude.

Was bedeutet Qualität

Begriff «Qualität»

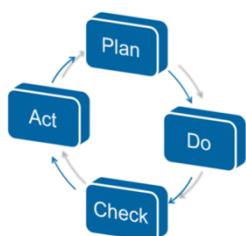
Die Herkunft stammt aus dem Lateinischen und zwar Qualitas. Der Begriff Qualität beinhaltet nicht zwingend eine Wertung. Im Sprachgebrauch ergänzt man daher oft auch Wörter wie gute oder schlechte Qualität.

Begriff «Qualitätsmanagement»

Qualitätsmanagement ist ein stetiger Prozess, um nachhaltig den Output zu steigern. QM ist je nach Branche gar nicht oder auch sehr stark vorgegeben. Im Textil-Bereich ist dies sehr gering, während QM in der Luftfahrt sehr wichtig ist. QM selbst möchte verbessern, um den dafür investierten Aufwand mehr als zurückzugewinnen.

Viele Akteure haben Qualitätsrelevante Interessen. Die Umwelt, Gesetze, der Kunde (Funktion, Ergonomie, Betrieb, Lebendauer, Kontinuität, Service, Support und Preis) sowie der Produzent (Mitarbeiter, Produktionsmittel, Material, Organisation, Entwicklung, Zulieferanten, Energie, Kosten und Termine).

Qualitätskreislauf nach Deming



Ein Qualitätskreislauf definiert nach Deming lautet wie folgt.

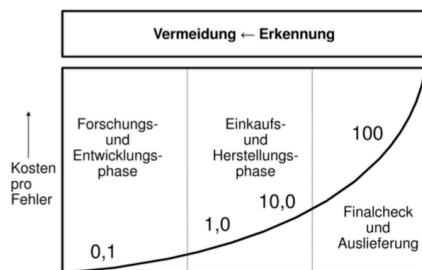
- | | |
|--------------|------------------------------------|
| Plan | Output bestimmen und planen |
| Do | Notwendige Arbeiten ausführen |
| Check | Output messen |
| Act | Massnahmen zur Korrektur bestimmen |

Plan – Think Smart

- Output mit SMART Ansatz definieren
 - o Spezifisch
 - o Messbar
 - o Akzeptiert
 - o Realistisch
 - o Terminiert
- Strategie der Umsetzung
 - o Maximal
 - o Minimal
 - o Optimiert
- Risiken beachten und Reserve einbauen

Check – Please mind the gap

- Überlege wie du checkst
- Finde Abweichungen früh

**Do – Do it right**

- Mach es OTOBOS
 - o On time
 - o On budget
 - o On scope
- Wenn der Output nicht klar ist, nach einem Prototyp
 - o Eine Zusatzrunde mit Check – Act – (Re)Plan

Act – Let's go

- Fixe die Abweichung
- Mach dein Produkt reif und bring es an die Front
- Definiere Massnahmen für den nächsten Zyklus

Beispiel eines Qualitätsplans in einem Projekt

- **2 Qualitätsziele und Risiken im Projekt**
 - 2.1 Qualitätsziele für Produkte und Prozesse
 - 2.2 Qualitätsrisiken
 - 2.3 Massnahmen auf Grund der Qualitätsziele und –risiken
- **3 QS-Massnahmen gemäss Kritikalität und IT-Sicherheit**
 - 3.1 Verwendete Richtlinien oder Normen
 - 3.2 Einstufungsbedingte QS-Massnahmen
- **4 Entwicklungsbegleitende Qualitätssicherung**
 - 4.1 Zu prüfende Produkte
 - 4.2 Zu prüfende Aktivitäten
- **5 Spezifische Kontrollmassnahmen**
 - 5.1 Eingangskontrolle von Fertigprodukten
 - 5.2 Kontrolle von Unterauftragnehmern
 - 5.3 Ausgangskontrolle der Software-Bausteine
 - 5.4 Änderungskontrolle
 - 5.5 Kontrolle von Bearbeitungskompetenzen
 - 5.6 Kontrolle des Konfigurationsmanagements

Projekte initialisieren, durchführen und abschliessen

Ziele des Kapitels

Lerninhalte	Ziele: Die Studierenden...
<ul style="list-style-type: none"> - Unternehmensziele - Projekttypen - Wie werden Projekte gestartet, wie beendet - Projektteam / Organisation - Requirements Engineering 	<ul style="list-style-type: none"> - kennen die Stolpersteine wenn Projekte nicht zu Unternehmen passen - vermögen nach der Vorlesung ein Projekt sauber zu starten und zu beenden - sehen die Möglichkeiten, wie man ein Projekt organisiert - erhalten einen Fresh-Up zu Requirements Engineering - diskutieren wann Anforderungen noch ändern dürfen

Unternehmensziele

Projektziele müssen möglichst mit Unternehmenszielen vereinbar sein, ansonsten riskieren Sie ein Scheitern wegen zu viel Widerstand. Unternehmensziele sind langfristig (grösser 5 Jahre), mittelfristig (3-5 Jahre) und kurzfristig (bis 3 Jahre) ausgelegt.

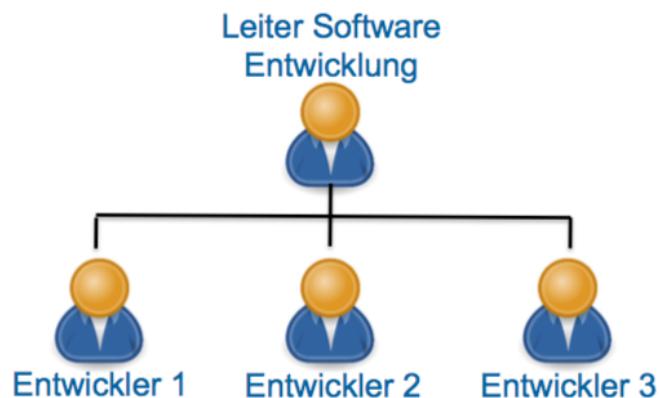
In dieser Tabelle ein paar Beispiele, wenn es nicht passt.

Unternehmensziel	Projektziel
Der Verwaltungsrat hat beschlossen Produkte ausschliesslich über indirekte Kanäle zu verkaufen.	Das Produktmanagement möchte mit Ihnen ein Projekt lancieren, um Velo-Computer direkt übers Internet zu verkaufen.
In der Zeitung lesen Sie: «Handelsgeschäft steht zum Verkauf, mehrere ausländische Unternehmen interessiert»	Sie bewerben sich als Projektleiter für den Aufbau einer neuen Handelsplattform.
Innert 5 Jahren sollen alle Standorte ausserhalb der Deutschschweiz geschlossen werden.	Ihr Chef möchte Sie als Projektleiter damit beauftragen eine mehrsprachige Lernplattform zu entwickeln.
Für «bahnfremde» Kommunikationsdienste gilt eine Outsourcing-Strategie an etablierte Telefongesellschaften.	Sie versuchen als Jungunternehmer einen Bahnbetreiber ein Projekt für WLAN im Zug zu verkaufen.

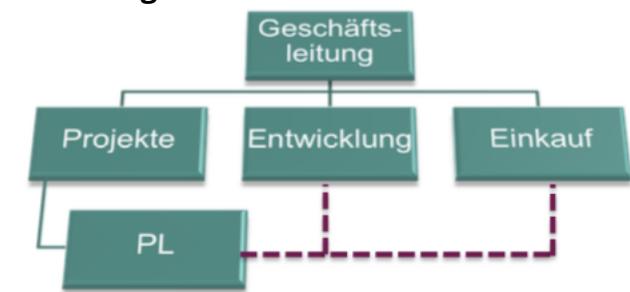
Projekttypen

Es gibt keine einheitliche Typendefinition. In der Literatur von Thomas Grossner findet man folgende vier Quadrate, welche eine gute Definition sind.

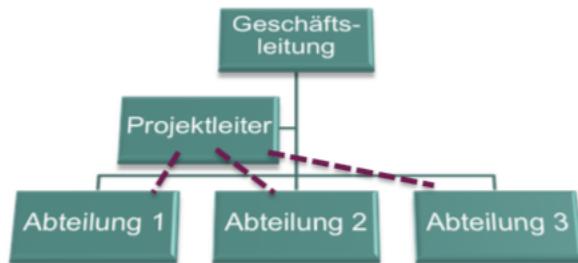
Linienprojekt	Stablinienorganisation
Matrixorganisation	Reine Projektorganisation

Liniensprojekt

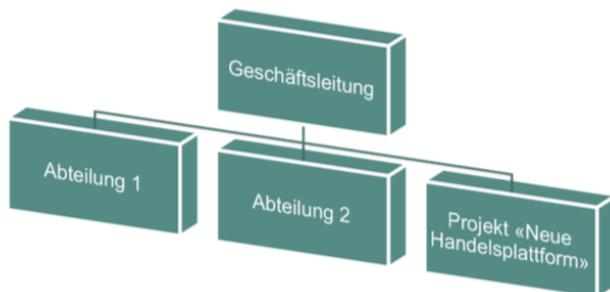
Alle wichtigen Akteure sind in der gleichen Einheit, ihrer Stammorganisation. Der Chef der Einheit ist gleichzeitig Projektleiter. Vereinzelt werden Personen aus anderen Einheiten beigezogen. Für kleine Projekte ohne grossen Anspruch an interdisziplinärem Mitwirken oder für Start-Up Unternehmen bestens geeignet.

Matrixorganisation

Die Akteure sind auf verschiedene Einheiten verteilt. Der PL ist einem Pool von Projektleitern zugeordnet und führt vielleicht mehrere Projekte parallel durch. Mit den Stammesorganisationen sind die Einsatzpläne abzustimmen. Der PL ist «normaler Mitarbeiter», die Unterstellung muss also geregelt werden. Für grosse Unternehmen geeignet, welche Bedarf für einen eigenen Pool an PL haben.

Stablinienorganisation

Die Akteure sind im Unternehmen in verschiedenen Stammesorganisationen verteilt. Ein PL agiert aus einer Stabsstelle heraus. Der Projektleiter kann einmal eingesetzt und anschliessend wieder verschwinden (z.B. externe). Flexible Lösung für mittelgrosse Projekte, da der PL nur die wirklich notwendige Ressource benötigt.

Reine Projektorganisation

Alle wichtigen Akteure werden zu 100 % für ein Projekt in der gleichen Einheit zusammengezogen oder sogar angestellt. Der Chef der Einheit ist gleichzeitig Projektleiter. Vereinzelt werden Personen aus anderen Einheiten beigezogen. Für grosse Projekte mit langfristigem Fokus. Mitarbeiter müssen nach dem Projekt entlassen oder reorganisiert werden.

Wie werden Projekte gestartet, wie beendet**Der Projektstart**

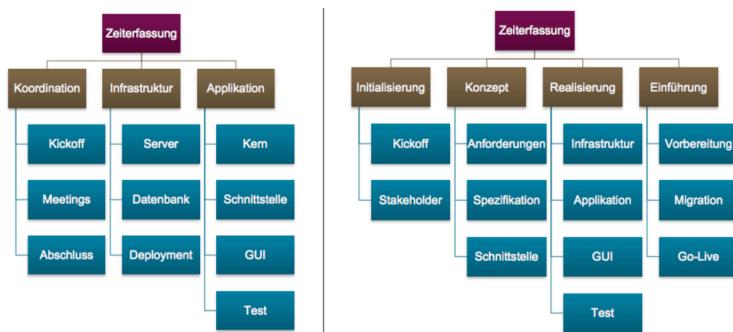
Häufige Fehler beim Projektantrag

- Die Argumentation des Zweckes ist nicht schlüssig, es fehlt ein roter Faden und die Gründe lassen sich nicht nachvollziehen
- Der Projektantrag widerspiegelt die Wünsche des Auftragsgebers nicht oder nur teilweise
- Der Antrag basiert auf einer teilweisen falschen Ausgangslage
- Die definierten Ziele sind nicht messbar
- Der Nutzen des Projekts wird viel zu optimistisch definiert
- Bedeutende Risiken werden nicht erkannt

Vorgehen bei der Planung

1. Projektstrukturplan

- Top-Down oder Bottom-Up Zerlegung des Projekts
- Mehrstufige Strukturierung von Arbeitspaketen



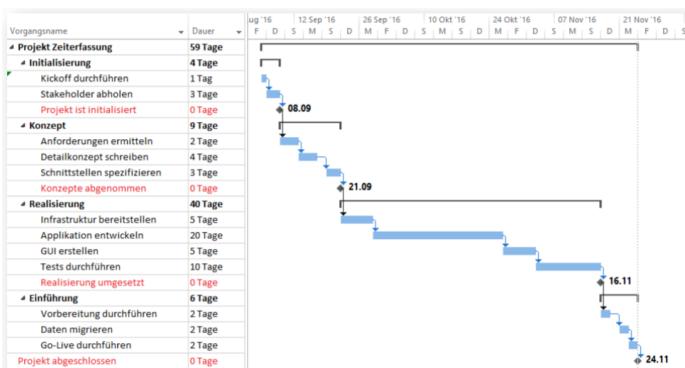
Strukturpläne können Sie nach Thema, Projektablauf, Funktion, etc. strukturieren.

2. Gantt Diagramm

- Erstellen Sie den Strukturplan nie ganz alleine, holen Sie Input und andere Meinungen ein
- Identifizieren Sie die Arbeitspakete systematisch Top-Down oder Bottom-Up
 - o **Top-Down:** Projekt zuerst grob strukturieren und dann immer detaillierter werden
 - o **Bottom-Up:** Sammeln aller realen Arbeitspakete und nachträglich die Gliederung definieren
- Der Strukturplan ist das Fundament, wenn Sie zu viel oder zu wenig detailliert sind, können Sie das Projekt nicht mehr vernünftig steuern.
- Trennen Sie nur Dinge voneinander ab, welche wenig gegenseitige Abhängigkeiten haben

2. Gantt Diagramm

- Aufwand und Zeitschätzung für jedes Arbeitspaket
- Abhängigkeiten ermitteln und in Phasen aufteilen



- Bilden Sie im Gantt Diagramm entweder drei Stufen:

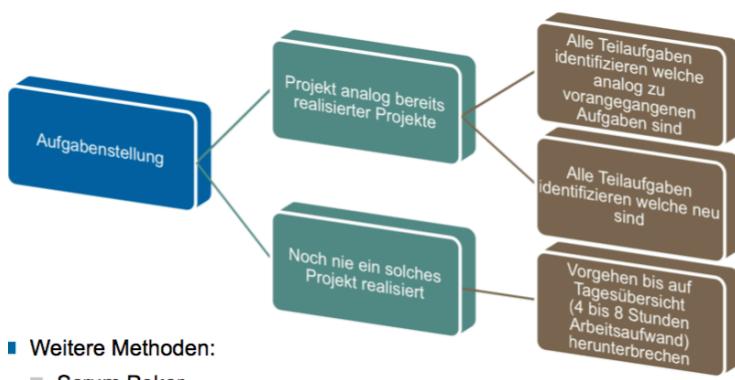
- Phase
- Arbeitspaket
- Task

- Oder bilden Sie zwei Stufen (Seite vorher) ab:

- Phase
- Task

Zu beachten beim Gantt-Diagramm

- Beim Gantt-Diagramm scheiden sich Anfänger und Profis, investieren Sie etwas Zeit um diesen korrekt und übersichtlich zu gestalten.
- Projekte die mehrheitlich agil sind, benötigen diesen Plan nicht zwingend
- Strukturierung in Phasen, Arbeitspakete und Tasks unbedingt beachten
- Phasen und Arbeitspakete benennen Sie mit einem Nomen
- Verwenden Sie für Tasks eine aktive Formulierung «Nomen + Verb» mit welcher Sie genau sagen, was zu tun ist (erstellen, spezifizieren, begleiten, etc.) und für Meilensteine «Nomen + Partizip II»
- Pro Phase definieren Sie mindestens einen Meilenstein, bevorzugt immer am Ende einer Phase um den Übergang (Gate) sauber zu steuern.

Analogien nutzen für die Zeitschaufschätzung

Weitere Methoden: Scrum Poker, Gauss-Annäherung ($1x$ Worst Case) + ($4x$ Normal Case) + ($1x$ Best Case) durch 6, White Elephant

3. Ressourcenplanung

- Zuweisung der personellen und materiellen Ressourcen
- Kostenbetrachtung für die eingesetzten Ressourcen

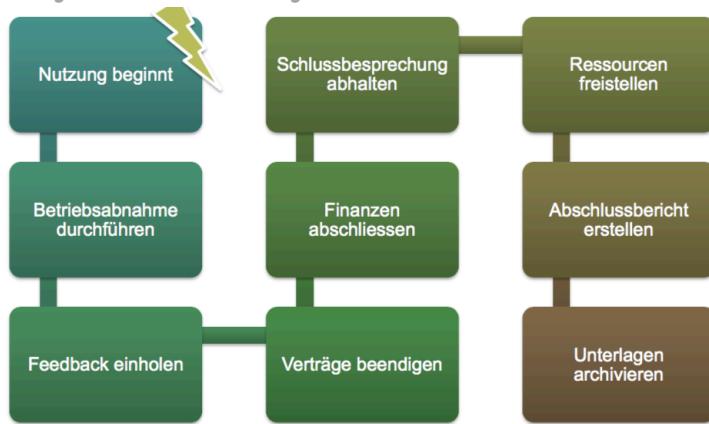
Tragen Sie die notwendigen Ressourcen für die Erfüllung wie Puzzle-Teile zusammen. Durch die Zuweisung dieser Ressourcen an die einzelnen Arbeitspakete erhält man einen geeigneten Überblick. Hinter den Ressourcen sind Kosten zu hinterlegen. Reserven von etwa 10% hinterlegen Sie entweder auf Ressourcen oder im Gantt Plan, nicht aber an beiden Orten.

4. Arbeitspakete

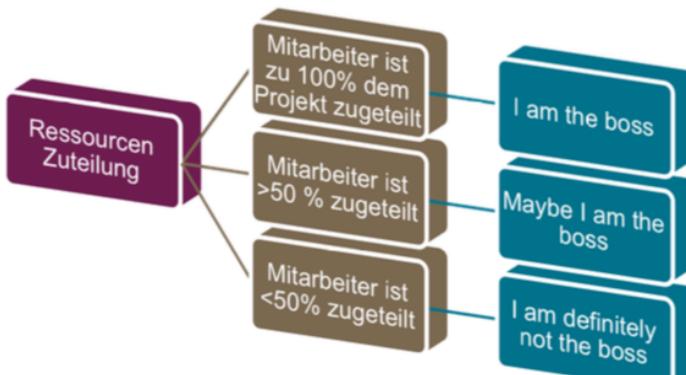
- Input und Output pro Arbeitspaket festlegen
- Arbeitspakete detailliert beschreiben

Arbeitspakete werden mit folgenden Parametern definiert. Name und eindeutige Nummer/Code, Verantwortliche Person, Ziel, Inout, Beschreibung, Ressourcen, Zeitplan und Output (Lieferergebnisse).

Die Beschreibung eines Arbeitspakets weicht nicht wesentlich von einem Projektauftrag ab, ein Arbeitspaket ist also ein kleines Projekt im Projekt.

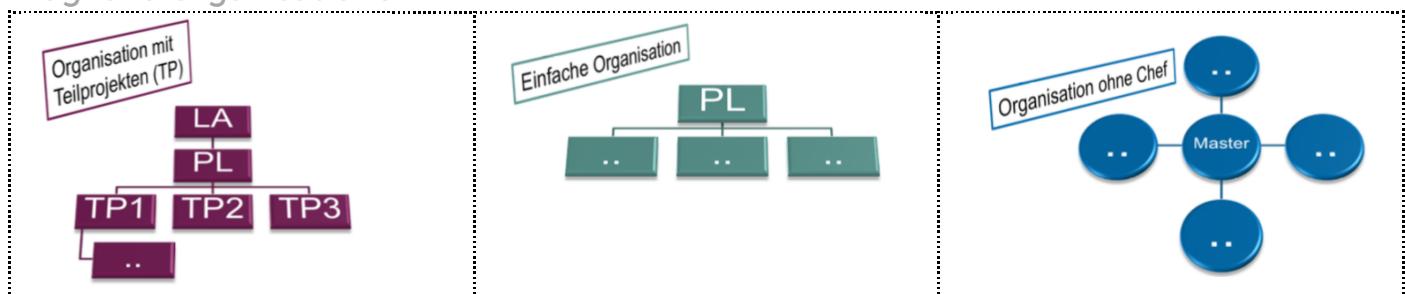


Projektteam / Organisation



Ab zwei Personen ist mit Koordinationsaufwand zu rechnen. Nehmen Sie sich bei der Zusammenstellung der Teams Zeit und legen Sie die Organisation fest, zeichnen Sie das Organigramm. Handeln Sie mit den Stammorganisationen aus, wer wie viel im Projekt beiträgt.

Mögliche Organisationen



AKV definieren

Definieren Sie ein AKV pro Rolle im Projekt.

	Aufgabe	Kompetenz	Verantwortung
Projektleiter (PL)	<ul style="list-style-type: none"> Leitet das Projekt Führt die Sitzungen Koordiniert die Mitarbeitenden 	<ul style="list-style-type: none"> Entscheidet innerhalb Projektbudget Definiert die Einsätze 	<ul style="list-style-type: none"> Hält das Projekt OTOBOS
Projektmitarbeiter	<ul style="list-style-type: none"> Leitet seine Arbeitspakete Arbeitet im Projekt mit 	<ul style="list-style-type: none"> Entscheidet innerhalb Arbeitspaket 	<ul style="list-style-type: none"> Verantwortet die Arbeitspakete
Lenkungsausschuss (LA)	<ul style="list-style-type: none"> Bewilligt Changes Kontrolliert das Projekt 	<ul style="list-style-type: none"> Entscheidet über Veränderungen bei Zeitplan, Budget und Scope 	<ul style="list-style-type: none"> Zeichnet sich für die Erreichung der Projektziele verantwortlich

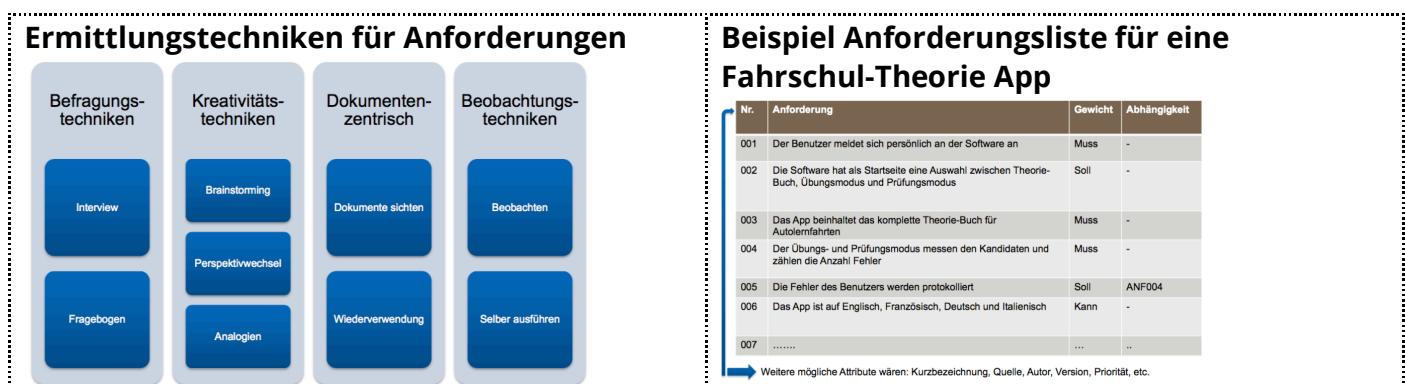
Gruppendynamik positiv umsetzen

Sprechregeln vereinbaren, zuhören nicht abschweifen; Ich-Botschaften anstelle von Man-Formeln; Ansichten sachlich diskutieren, Kritikkultur fördern; Zusammengehörigkeit fördern; Konflikte früh angehen und im Keim ersticken; Pünktlich und zuverlässig; Soziale Events planen; Motivieren; Jeder ist ein Vorbild.

Das Erheben von Anforderungen wurde gemäss den vorliegenden Unterlagen bereits im Modul SE1 grundlegend behandelt.

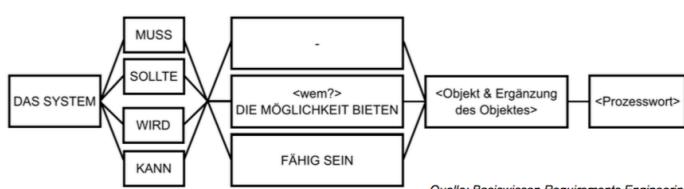
Ein kleines Fresh-Up

- Anforderungen sind der Grundstein für ein IT-Projekt
- Im agilen Vorgehen «User Story» genannt
- Vom Kunden erwarten wir funktionale Anforderungen, die beschreiben «**Was** wird gewünscht»
- Meistens kann der Kunde nicht strukturieren, wir müssen dies tun:
 - o Wie genau lautet die Anforderung?
 - o Abhängigkeiten zu anderen Anforderungen?
 - o Wie wichtig ist die Anforderung?
- Bilder zeigen mehr als tausend Worte, modellieren Sie die Anforderungen und verwenden Sie dazu eine normierte Sprache wie BPMN oder UML.



Achtung vor Modalverben

In der Theorie werden oft Modalverben verwendet, z.B. «Das System muss die Fehler des Benutzers protokollieren.»



Dies bringt den Nachteil, dass bei einer Änderung der Gewichtung diese Verben wieder ausgetauscht werden müssen, einfacher ist m. E. daher: «Das System protokolliert die Fehler des Benutzers».

Agil oder Wasserfall – Auswirkungen auf die Anforderungen



Projektkontrolle

Ziele des Kapitels

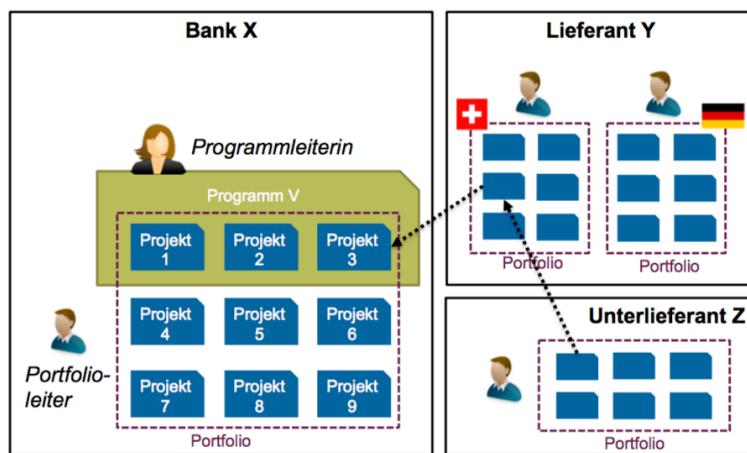
Lerninhalte

- Wer kontrolliert die Projekte
- Berichtswesen
- Projekt Cockpit
- Standards für die Projektkontrolle
- Earned-Value-Analyse
- Meilenstein-Trendanalyse

Ziele: Die Studierenden...

- kennen die Stolpersteine wenn Projekte nicht zu Unternehmen passen
- verstehen, dass Berichte nur ein Teil einer Kontrolle sind
- kennen Standard zur Projektkontrolle
- wenden zwei Analysen an

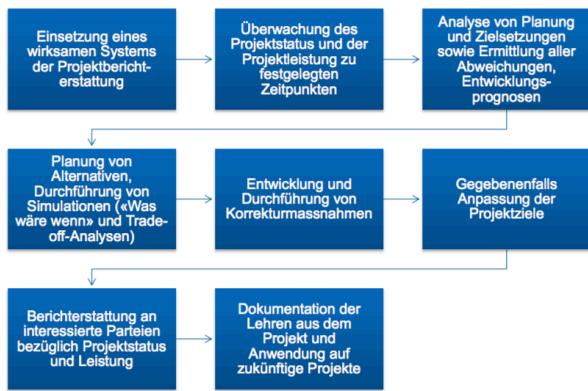
Wer kontrolliert die Projekte



Sie selber als Projektleiter steuern grundsätzlich selber innerhalb Budget, Zeitplan und Scope. Der Auftraggeber oder der Kunde kontrolliert Sie. Weiter verfügen Projekte über einen Lenkungsausschuss, der wichtige Entscheide fällt. Je nach Projektvorgenen gibt es weitere regelmässige Kontrollen durch das Finanzwesen und Audits durch den Qualitätsbeauftragten. Arbeiten mehrere Projekte einem Ziel zu, ordnet man diese

einem Programm (Multi-Projekt) unter. Der Leiter des Programms überprüft dann die ihm unterstellten Projekte zusätzlich. Unternehmen mit mehreren Projekten verfügen über einen Portfolio Manager. Diese kontrollieren die Projekte in ihrem Portfolio. Das Portfolio Management ist eine stetige Aufgabe und zeitlich unbefristet.

Berichtswesen



Projekte «Insieme» zeigen das ein Berichtswesen nötig. Ist Ein Aufbau einer Kontrolle nach NCB sieht wie folgt aus.

Welches Projektkontrollsysteem nehme ich?

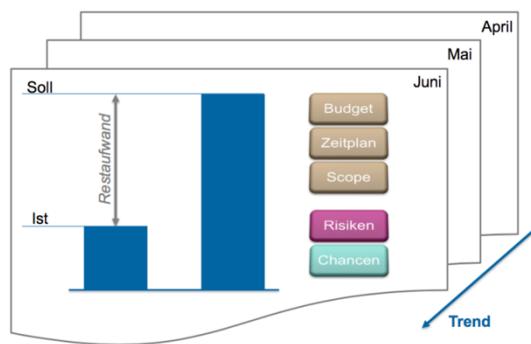
Nehmen Sie grundsätzlich immer jenes, welches bereits im Unternehmen vorhanden ist. Falls keines vorhanden ist, nehmen Sie ein anderes oder bauen Sie selber ein Berichtswesen unter Beachtung folgender Punkte auf :

- Status von Budget, Zeitplan und Scope
- Qualitative Beschreibung des Fortschritts oder

Restaufwandschätzung

- Risiken und Chancen im Projekt
- Issues
- Weitere Analyse-Charts
- Möglichkeit für Kommentare

Achtung! Ein Berichtswesen ist lediglich eine formale Meldung und noch eine vollständige Kontrolle. Kontrolle heisst auch Steuern.



Aufgabe des Projektleiters ist es geeignete Mess- und Überprüfungsverfahren zu Beginn des Projekts einzuführen. Aufgabe des Controllers ist es zu überprüfen, ob das gemacht wurde und auch sinnvoll ist. Je nach Standardisierung von Projekten ist das einfacher oder schwieriger. Der Controller sollte rechtzeitig mit einbezogen werden.

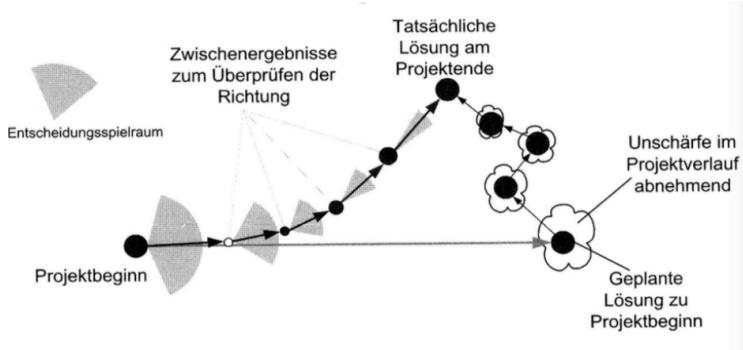
Bei Abweichungen zum Plan muss gehandelt werden.

Folgende Massnahmen zum Beispiel

- Änderung in der Planung, Serielle in parallele Tätigkeiten ändern
- Überzeit anordnen
- Coaching oder zusätzliche Unterstützung anfordern

Bringen die Massnahmen keine Verbesserung, muss ein Change-Request gestellt werden.

Effektiver Verlauf in den Projekten

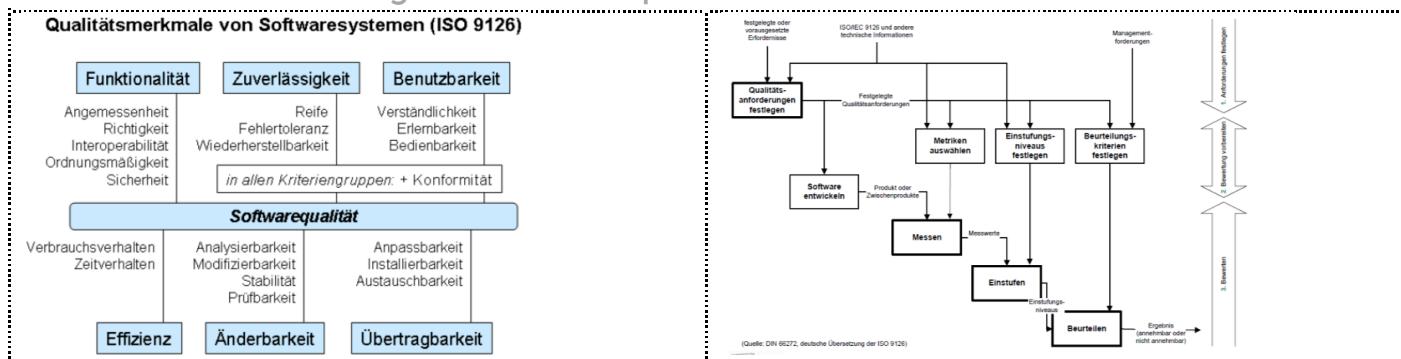


Wie messe ich den genauen Fortschritt?

Faktisch kaum möglich, Unschärfe vorprogrammiert. Am besten Produkt betrachten und Fertigkeitsgrad ermitteln. Fragen Sie einen Entwickler lieber, wie viel Zeit er noch benötigt und nicht wie weit er prozentual liegt. Legen sie die Metriken für die Kommunikation mit dem Auftraggeber fest, z.B. 80 Prozent bedeutet fertig aber noch nicht getestet (Definition of Done). Nun ein Beispiel einer Definition of Done in einem Projekt.

Wert	Kriterium
0%	Nicht begonnen
10%	Gestartet
10%-80%	In Arbeit
80%	Erledigt
100%	Review/Modultest ok

ISO 9126 – Sicherstellung der Softwarequalität



Das Ampel-Prinzip

Das Ampel-Prinzip hilft, Fakten sehr transparent und klar zu kommunizieren. Jedes Unternehmen hat eigene Definitionen von den Farben, bei IPG gilt:

- **ROT** – Abweichung grösser als 5 % → Eskalation
- **GELB** – Abweichung 0 – 5 % → Beobachtung
- **GRÜN** – Alles läuft nach Plan

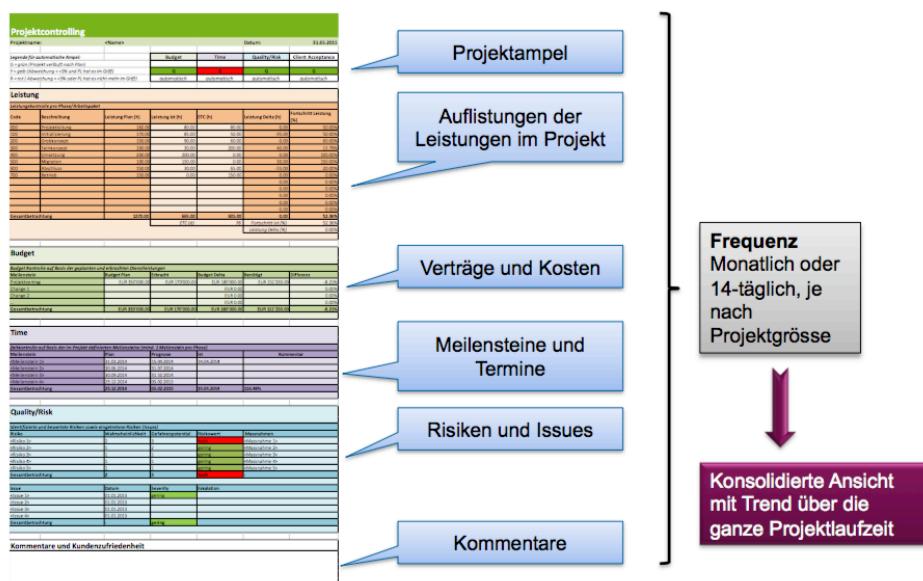
Die drei Stufen zwingen den Projektleiter klar Stellung zu nehmen. Portfoliomanager können sich schnell einen Überblick verschaffen.

Ampeln aggregieren

Innerhalb eines Projektes gilt generell die Aggregation nach dem Maximum Prinzip, Ausnahmen bestätigen die Regel. Wenn also Budget GELD, Zeitplan GRÜN und Scope GRÜN dann ist das Gesamtprojekt GELD.

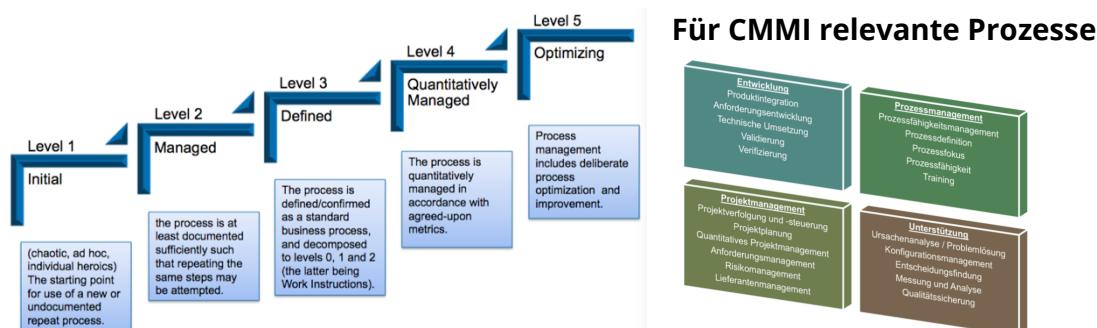
Projektcockpit

Bedeutet eigentlich Hahnengrube, ist aber mit einem Flugzeugcockpit zu vergleichen. Ein Auszug eines Projektcockpits (Controlling) könnte wie folgt aussehen.



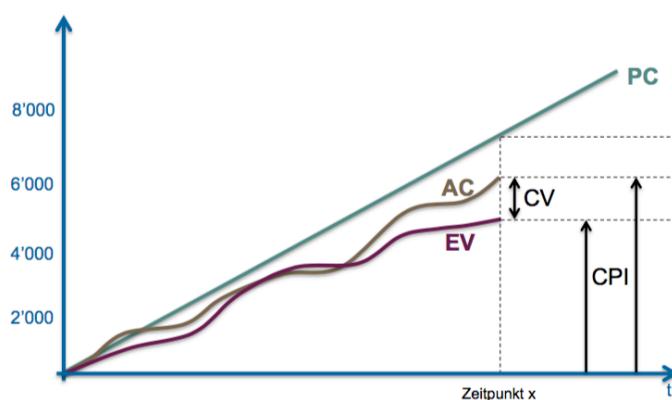
Standards für die Projektkontrolle

Qualitäts- und Reifegradstandards helfen Ihnen um das gleiche Verständnis von Qualität oder Reife zu schaffen. Als Beispiel CMMI zur allgemeinverständlichen Definition von Reife. Es wurde 1987 entwickelt vom SEI (Software Engineering Institute). Es beschreibt fünf Reifegrade von Softwareentwicklungsorganisationen. Enthält keine Anleitung wie, sondern legt Ziele fest, welche eine Software, Organisation oder Abteilung erreichen muss. Prüfung erfolgt durch Assessments.



Earned-Value-Analyse

Die Earned-Value-Analyse (EVA) ist die meist genannte und bekannteste Analyse um den Projektfortschritt zu analysieren und vor allem darzustellen. In Büchern und Artikeln omnipräsent, in der Praxis meist in abgewandelter Form anzutreffen. Die Analyse steht und fällt mit der Bestimmung des Fertigstellungswerts (Earned Value). Ziel der Analyse ist eine Ermittlung der Kosteneffizienz, also ob die verbrauchten Ressourcen im Verhältnis zum Fortschritt stehen. Angestrebgt wird eine Kosteneffizienz von >1.



Planned Cost (PC)

Die zu Beginn geplanten Kosten/Aufwände für das Projekt

Actual Cost (AC)

Die bis jetzt ausgegebenen Kosten /Aufwände für das Projekt

Earned Value (EV)

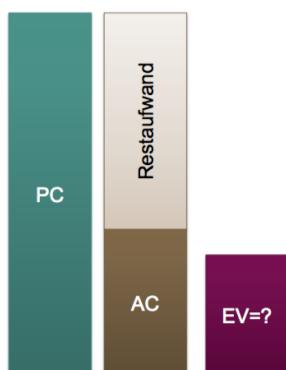
Der Wert der bis jetzt erbrachten Arbeit

Cost Varaince (CV) Wird berechnet mittels $EV - AC$

Cost Performance Index (CPI) Wird berechnet mittels EV/AC

Wie berechne ich den Earned Value (Fertigstellungsgrad)?

Zeitpunkt x



Variante «Strikt»

Nur realisierte Werte haben einen Nutzen, daher halbfertige Software = 0 CHF.

Variante «Zwischenresultate»

Die Summe aller brauchbaren Komponenten definiert den EV

Variante «Restaufwand»

$$EV = \frac{PC}{AC+REST} * AC$$

Beispiel einer EVA an einem Studentenprojekt

Sie arbeiten in einem Projekt und rechen mit Kosten von CHF 100.- pro Stunde. Im Projekt sind 300h vorgesehen. In der Mitte des Projekts (nach 150h aufgewandter Arbeit) machen sie eine Restzeitschätzung und realisieren, dass Sie noch 225h für die Fertigstellung benötigen. Wie hoch sind zu diesem Zeitpunkt folgende Kennzahlen?

Earned Value

$$PC = 300 * 100, AC = 150 * 100, REST = 225 * 100 \rightarrow EV = 12000$$

Cost Variance

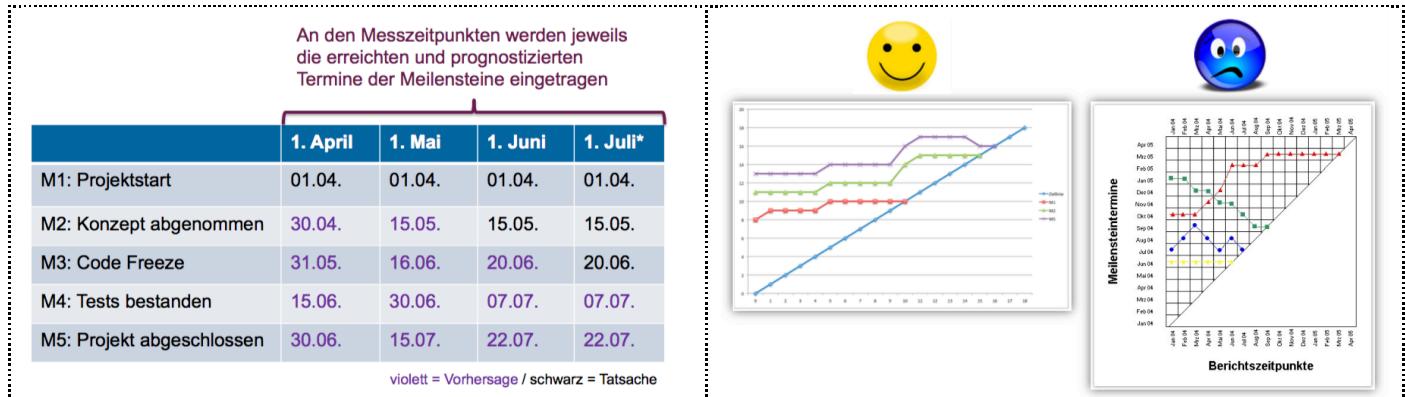
$$EV = 12000, AC = 15000 \rightarrow CV = -3000$$

Cost Performance Index

$$EV = 12000, AC = 15000 \rightarrow CPI = 0.8$$

Meilenstein-Trendanalyse

Die Meilenstein-Trendanalyse (MTA) ist eine rückwärtsbetrachtete und sehr eindrucksvolle Analyse, da sie den Verlauf des Projekts schonungslos aufdeckt. Die Analyse basiert darauf, dass Sie mit klug gewählten Meilensteinen arbeiten. Die Meilensteine müssen möglichst wichtige Ereignisse (z.B. Phasenende, Erreichung gewisser Funktionalität, etc.) terminlich fixieren. Die MTA zeichnet den Verlauf der Meilensteine auf. Angestrebt werden möglichst horizontale Linien.



Klassisches Projektmanagement

Ziele des Kapitels

Lerninhalte

- Vorgehen
- HERMES, Vorgehensmodell des Bundes
- Prototypenmodell
- Spiralmodell

Ziele: Die Studierenden...

- kennen mehrere Vorgehensmodelle und können diese abwägen
- verstehen klassisches Projektmanagement auf Phasen basierend
- verstehen was HERMES beinhaltet
- wissen wann klassische Projektmethoden eingesetzt werden

Vorgehen

Vorgehensmodelle

In den Modulen SE1 und SE2 hatten Sie bereits einige Vorgehensmodelle in den Vorlesungen behandelt. Unter anderem RUP, V-Modell und Scrum. Vorgehensmodelle werden in der Praxis frei kombiniert, dies ist ein legitimer Ansatz zur Bewältigung von Herausforderungen. Folgende weitere Modelle werden in diesem Kapitel ebenfalls noch erläutert: **HERMES, Prototypenmodell** und **Spiralmodell**.

Das Vorgehen

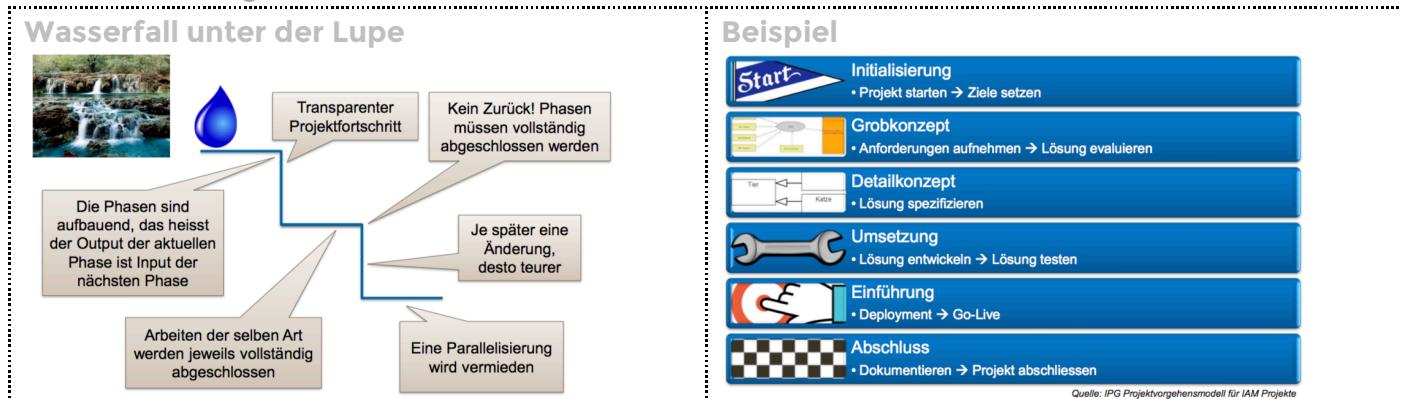
Das Vorgehen wird zu Beginn des Projekts definiert, eine Änderung ist nur schwer möglich (aber nicht unmöglich). Es wird zwischen agil und klassisch unterschieden

- Agile Vorgehensmethoden basieren dank interaktivem Vorgehen auf schlanken Prozessen im Projekt, damit weniger Zeit für administratives aufgewendet wird.

- Klassische Vorgehensmethoden sind formeller und strikter, was klare Aussagen zu Fortschritt und Inhalt zulässt, wiederrum aber mehr administrativer Aufwand bedeutet.

Ein «projekttechnisch» guter Projektleiter kennt mehrere Methoden und hat diese auch erfolgreich (oder nicht erfolgreich) angewendet. Klassische Modelle benötigen ein gutes Change Management.

Klassische Vorgehensmodelle



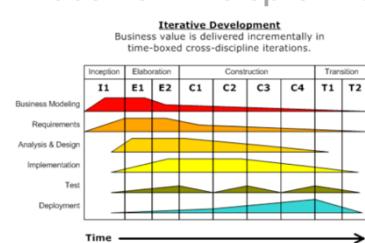
Definition einer Phase

Im klassischen Projektmanagement werden die Arbeitspakete oder Tasks in logische Abschnitte, sprich Phasen unterteilt. Alles was von der Tätigkeit her zeitlich und inhaltlich zusammenhängend ist, wird in eine Phase gepackt, zum Beispiel



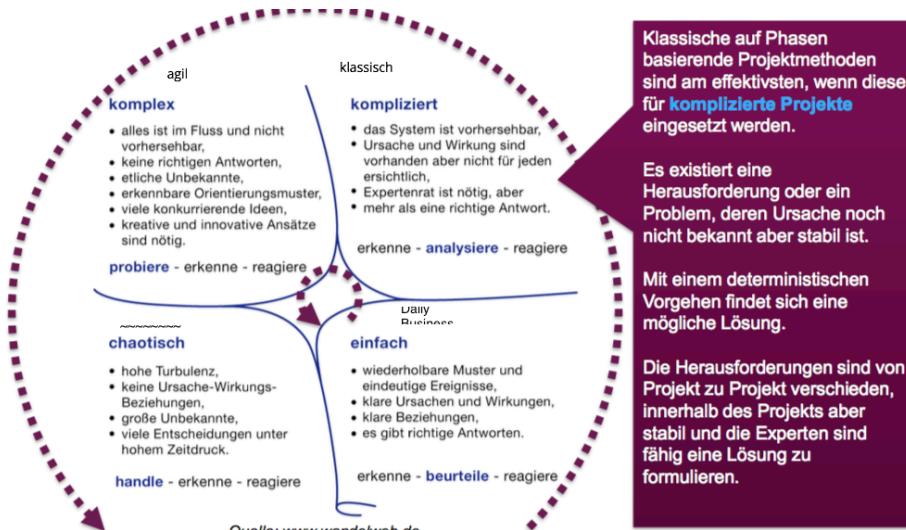
In der Regel terminiert ein Meilenstein eine Phase (schwarzes Viereck). Phasenübergänge werden auch «Gates» genannt. Eine Parallelisierung von Phasen ist nicht vorgesehen, kommt jedoch in der Praxis vor.

Phasen am Beispiel RUP



RUP den die vier Phasen Inception, Elaboration, Construction und Transition. RUP arbeitet sequentiell. Die fachlichen Zusammenhänge sind jedoch nicht strikt gegeben, da RUP davon ausgeht, dass zum Beispiel noch bis in die Transition-Phase Tasks in der Disziplin «Analysis & Design» erledigt werden müssen.

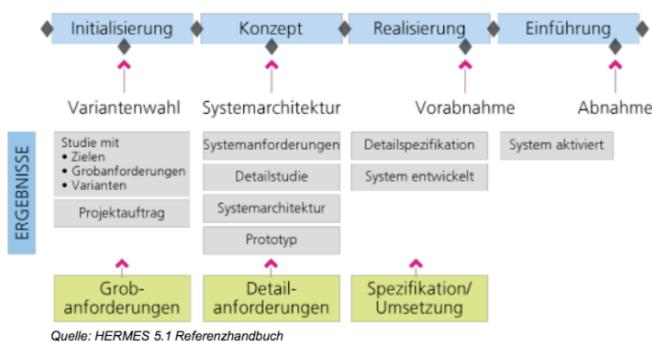
CYNEFIN - Wann ein Phasenmodell



HERMES, Vorgehensmodell des Bundes

HERMES steht für Handbuch der Elektronischen Rechenzentren des Bundes, eine Methode zur Entwicklung von Systemen (hermes.admin.ch). Aktuell gibt es die Version 5.1. Es gibt genau vier Phasen, welche klar vorgegeben sind.

Phasen



Rollenverteilung bei HERMES



Aufgaben der Projektsteuerung

	Initialisierung	Konzept	Realisierung	Einführung
Projektsteuerung	Initialisierung beauftragen und steuern	Projekt steuern Entscheid zur Phasenfreigabe treffen	Projekt steuern Entscheid zur Phasenfreigabe treffen	Projekt steuern Entscheid zum Projektabschluss treffen
	Entscheid zur Projektfreigabe treffen			

Quelle: HERMES 5.1 Referenzhandbuch

Wichtig: Bei HERMES bedürfen Phasenübergänge (Gates) einer Freigabe durch die Lenkungsausschuss. Dies bedeutet, dass man als Projektleiter nur innerhalb einer Phase unabhängig arbeiten kann.

Aufgaben der Projektführung

	Initialisierung	Konzept	Realisierung	Einführung
Projektführung	Initialisierung führen und kontrollieren Entscheid zur Variantenauswahl treffen Projektauftrag erarbeiten	Projekt führen und kontrollieren Leistungen vereinbaren und steuern Probleme behandeln und Erfahrungen nutzen Stakeholdermanagement und Kommunikation führen Qualitätssicherung führen Risiken managen Änderungsmanagement führen Phasenfreigabe vorbereiten	Projekt führen und kontrollieren Leistungen vereinbaren und steuern Probleme behandeln und Erfahrungen nutzen Stakeholdermanagement und Kommunikation führen Qualitätssicherung führen Risiken managen Änderungsmanagement führen Phasenfreigabe vorbereiten	Projekt führen und kontrollieren Leistungen vereinbaren und steuern Probleme behandeln und Erfahrungen nutzen Stakeholdermanagement und Kommunikation führen Qualitätssicherung führen Risiken managen Änderungsmanagement führen Projektabschluss vorbereiten

Quelle: HERMES 5.1 Referenzhandbuch

Alle Projektteilnehmer nach HERMES (standardisiert)

Hierarchie-Ebene	Rolle	Anwender	Ersteller	Betreiber
Steuerung	Auftraggeber	X		
	Projektausschussmitglied	X	X	X
	Qualitäts- und Risikomanager	X		
Führung	Projektleiter	X		
	Teilprojektleiter	X	X	X
	Projektunterstützung	X	X	
	Fachausschussmitglied	X	X	X
Ausführung	Fachspezialisten			
	Anwendervertreter	X		
	Anwendungsverantwortlicher	X		
	Betriebsverantwortlicher			X
	Business Analyst	X	X	
	Entwickler		X	
	Geschäftsprozessverantwortlicher	X		
	IT-Architekt	X	X	X
	ISDS-Verantwortlicher	X		
	Testverantwortlicher	X	X	X
	Tester	X	X	X

Quelle: HERMES 5.1 Referenzhandbuch

Pro Projektteilnehmer ein definiertes AK(F)

Entwickler

Die Rolle Entwickler ist umfassend und bezeichnet den Produktentwickler und den IT-Entwickler. Der Entwickler entwirft, gestaltet und erstellt das Produkt bzw. das IT-System gemäss den Anforderungen unter der Führung des Projektleiters. Er integriert das Produkt bzw. das IT-System in die Umgebung des Betreibers.

Verantwortung

- Verantwortet die Erarbeitung der Entwicklungsergebnisse unter Einhaltung der gesetzten Termine und Kosten

Kompetenzen

- Kann auf alle benötigten Informationen zugreifen

Fähigkeiten

- Vertiefte Kenntnisse im Spezialgebiet Produkt- oder Softwareentwicklung
- Vertiefte Kenntnisse der Methoden und Praktiken für Design, Spezifikation, Entwicklung, Test und Integration
- Kenntnisse von HERMES
- Teamfähigkeit, Kommunikationsfähigkeit und Konfliktlösungsfähigkeit

Quelle: HERMES 5.1 Referenzhandbuch

Jede Aufgabe ist bis ins Detail beschrieben

Migrationskonzept erarbeiten

Das Migrationskonzept legt die Grundlage für die Überführung des alten in das neue System und die Ausserbetriebssetzung des Altsystems.

Grundidee

Der Schwerpunkt der Migration von IT-Systemen ist die Migration von Daten. Migrationen können technisch (maschinell) oder organisatorisch (manuell) erfolgen. Das Migrationskonzept berücksichtigt die Mengen, Häufigkeiten und Qualität der Daten im Altsystem und ihre Integration ins Zielsystem. Mögliche Migrationszonen werden analysiert und beurteilt, sodass die geeigneten Migrationsverfahren bestimmt werden können.

Bei den Migrationsüberlegungen fließen Aspekte der Machbarkeit, der Wirtschaftlichkeit, der Qualität und des zeitlichen Ablaufs einer Migration ein. Mit der Migration der Daten müssen auch die Fragen der Archivierung von Altdaten und dem Systemabbau beantwortet werden. Die Aspekte der Datensicherheit und des Datenschutzes werden berücksichtigt.

HERMES Spezifisch

Die Einführungsstrategie im Einführungskonzept bestimmt die Migrationsstrategie (eine stufenweise Einführung erfordert z.B. eine stufenweise Migration).

Aktivitäten

- IT-Systeme und Datenanalyse durchführen
- Migrationskonzept auf der Grundlage des Einführungskonzepts erarbeiten
- Auswirkung auf das Einführungskonzept überprüfen
- Abbau des Altsystems konzipieren und bei Bedarf Datenarchivierung klären
- Machbarkeit überprüfen
- Migrationskonzept mit den Stakeholdern abstimmen

Quelle: HERMES 5.1 Referenzhandbuch

Vor- und Nachteile von HERMES

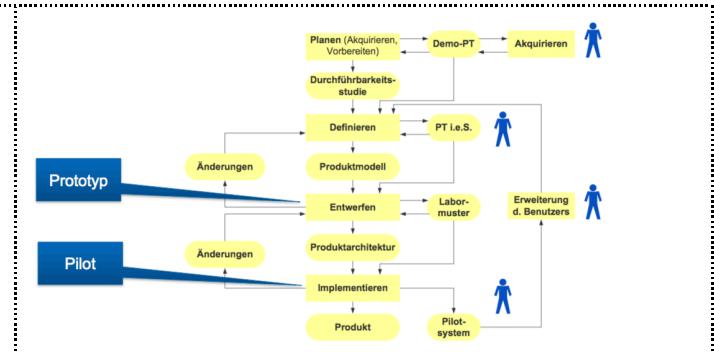
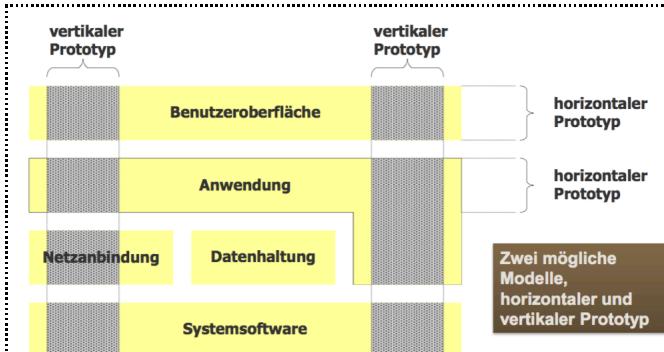
Vorteile

- Hohe Standardisierung mit klar definierten Phasen, Rollen, Aufgaben, Ergebnissen, etc.
- Eine Zertifizierung erlaubt die Mitarbeit in Bundesprojekten
- Einbettung von Scrum ist klar definiert
- Passt zu Institutionen, welche ihre Aufträge öffentlich ausschreiben oder Dienstleister welche an den Ausschreibungen teilnehmen.

Nachteile

- Sehr starke Vorgaben mit wenig Spielraum
- Vier Phasen sind etwas knapp bemessen
- In der Privatwirtschaft und sowieso ausserhalb der Schweiz kaum relevant

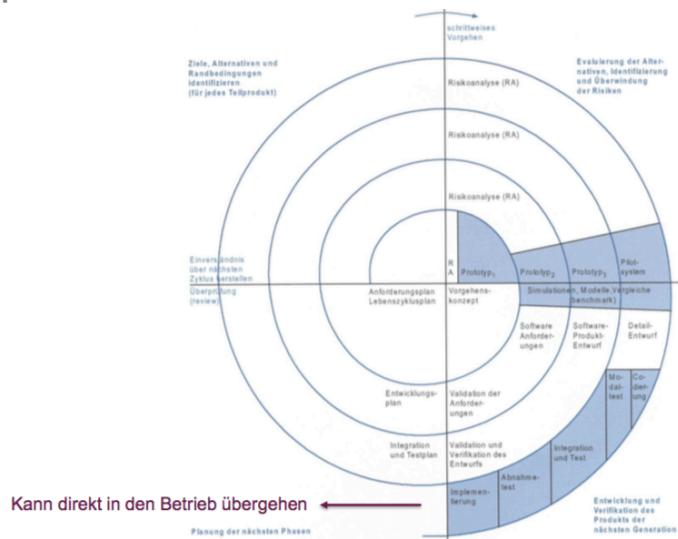
Prototypenmodell



Jeder Prototyp selbst muss definiert, entworfen und implementiert werden. Beispiele aus der Praxis sind Aeromobil oder Rollable Display.

Merkmale

- Auftraggeber und Endbenutzer sind oft nicht in der Lage, die Anforderungen zu beschreiben
- Während der Entwicklung ist oft eine Absprache mit dem Anwender notwendig
- Kann mit anderen Modellen wie Wasserfall, V-, RUP-Modellen kombiniert werden
- Realisierbarkeit von Anforderungen lässt sich manchmal nicht garantieren
- Unterstützt die Akquisitionsphase, indem der Anwender mit Prototypen von speziellen Funktionen überzeugt werden kann
- Es gibt verschiedene Prototypen (Demo, Muster, Pilot) aber diese sind in der Regel einzigartig



Risikogetriebenes Modell, bei dem oberstes Ziel die Minimierung des Risikos ist. Jede Spirale stellt einen iterativen Zyklus durch dieselben Schritte dar. Die Ziele für jeden Zyklus werden aus den Ergebnissen des letzten Zyklus abgeleitet. Keine Trennung der Entwicklung und Wartung. Das Ziel ist es beim Kleinen zu beginnen und damit die Spirale so eng wie möglich zu halten und erreiche somit die Entwicklungsziele mit minimalen Kosten. Bei der Zielbestimmung werden auch Qualitätsziele aufgeführt. Für jede

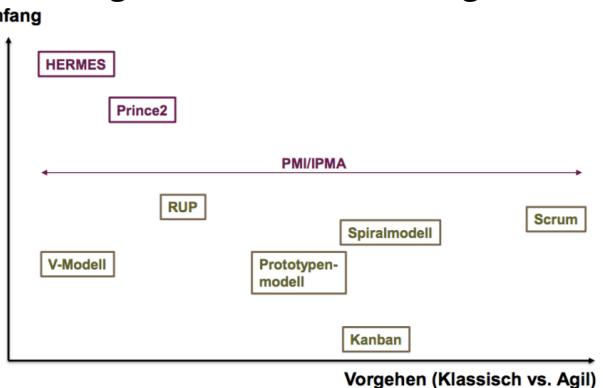
Aktivität und jeden Ressourcenverbrauch wird gefragt «Wieviel ist genug?». Dadurch wird ein «Overengineering» vermieden.

Diskussion

Projektstandards/Zertifizierungen und ihre Modelle

- PMI → **Vorgehen:** Keine Vorgaben
→ **Umfang:** Kein Projektstandard im Detail sondern Zertifizierung
- Prince2 → **Vorgehen:** Impliziert Phasenmodell
→ **Umfang:** Vorgegebener Standard und Zertifizierung
- IPMA → **Vorgehen:** Keine Vorgaben
→ **Umfang:** Grober Standard und Zertifizierung
- HERMES → **Vorgehen:** Vorgehensmodell bis ins Detail festgelegt
→ **Umfang:** Detailliert vorgegebener Standard und Zertifizierung

Einordnung der Standards und Vorgehen



Finanzierung, Kosten und Beschaffung

Ziele des Kapitels

Lerninhalte

- Kosten
- Nutzen
- ROI
- Business Case
- Finanzierung und Liquidität
- Beschaffung und Verträge

Ziele: Die Studierenden...

- ermitteln Kosten/Nutzen/ROI eines Projekts
- wissen was ein Business Case beinhaltet
- erkennen die Schlüsselpunkte bei der Finanzierung
- kennen die Stolpersteine bei der Beschaffung und die verschiedenen Vertragsarten

Kosten

Die Kosten in einem Projekt können sich aus folgenden Aufwänden zusammensetzen.

- Personalaufwand (intern/extern)
- Material (intern/extern)
- Software (Kauf/Miete)
- Spezielle Ausbildungen (intern/extern)
- Spesen (Hotel, Transport, Essen, etc.)
- Finanzierung (Zins, Versicherungen)
- Diverses (Bücher, Normen)
- Rückstellungen (Reverse)
- Risiken (Risiko Positionen)

Beispiel Projekt-Kosten-Kalkulation

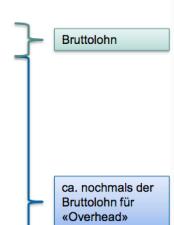
	Arbeits-	Aufwand in	Stundensatz	Kosten
	paket	Stunden		
Investitionskosten				
Hardware (Server, Netzwerk etc.)				SFr. 20'000.00
Software				SFr. 20'000.00
Entwicklung/Test Infrastruktur (SW,HW,Lizenzen etc.)				SFr. 10'000.00
Kosten Firma A				SFr. 15'000.00
Kosten Firma B				SFr. 15'000.00
Allgemeine Kosten				
Ausbildung (Kurskosten)				SFr. 4'000.00
Reisespesen (Unterkunft, Reise, Verpflegung)				SFr. 3'000.00
Sonstiges (Material, Bücher, etc.)				SFr. 2'000.00
Personalkosten (Projektaufwand)				
Projektmanagement	AP 01	200	SFr. 200.00	SFr. 40'000.00
Qualitätsmanagement	AP 02	100	SFr. 180.00	SFr. 18'000.00
Spezielle Projekt Ausbildung	AP 03	40	SFr. 120.00	SFr. 4'800.00
Business Engineering	AP 04	100	SFr. 160.00	SFr. 16'000.00
System Architecture	AP 05	100	SFr. 160.00	SFr. 16'000.00
Software Entwicklung 1	AP 06	150	SFr. 120.00	SFr. 18'000.00
Software Entwicklung 2	AP 07	150	SFr. 120.00	SFr. 18'000.00
Systemtest	AP 08	200	SFr. 140.00	SFr. 28'000.00
Validationstest	AP 09	200	SFr. 140.00	SFr. 28'000.00
etc.	AP XX			SFr. 0.00
Reserve				
Reserve Personalaufwand		10%		SFr. 19'000.00
Reserve Diverses (UV, Finanzierung, etc.)				SFr. 10'000.00
Risiko Rückstellungen (aus Risk Management)				SFr. 20'000.00
Total Projektkosten				SFr. 324'800.00

Personalkosten in einem Unternehmen

Der Gehalt ist nur die Hälfte der wirklichen Kosten, für das Unternehmen kommt nochmals der Bruttolohn dazu.

■ Lohnkosten

- Gehälter (fix und variabel)
- Sozialkosten (AHV, BVG, ALV, NBU, KV etc. ca. 25 bis 35%)
- Ausbildung



■ Arbeitsplatzkosten

- Büromiete, Energie, Wasser
- Büromöbel
- Computer und Netzwerke, Tools und Lizenzkosten



■ Verwaltungskosten

- Personalwesen, Personalführung, Marketing und Verkauf

■ Auslastung und Gewinn

- Gewinn (ca. 10%)
- Auslastung, hat einen grossen Einfluss auf den internen Stundensatz

Die Auslastung des Mitarbeiters hat einen Einfluss auf den Stundensatz gegenüber dem Kunden. Die Unternehmen haben ein Zielband von 60 bis 80 Prozent, dass bei 80 % in 160 Franken pro Stunde resultiert.

Nutzen

Nutzelemente von Projekten

Finanzen	Compliance
Return of Invest	Erfüllung der Gesetze
Weniger Ausgaben	Einhaltung Normen
Mehr Einnahmen	Einhaltung Soft-Law
Agilität	Qualität
Schnelle Prozesse	Image erhalten
Flexible Strukturen	Zuverlässigkeit
Hohe Geschwindigkeit	Datenhygiene

Der Nutzen eines Projekts muss wie die Kosten auch immer vor dem Projektstart bekannt und vom Management akzeptiert sein. Quantitativer Nutzen (materiell) zeigt vor allem einen Finanz-Nutzen. Qualitativer Nutzen (immateriell) zeigt vor allem Elemente von Compliance, Agilität und Qualität, z.B.

- Erfüllung neues Datenschutzgesetz
- Schneller am Markt als Konkurrenz
- Stärkung der eigenen Marke dank hoher Verfügbarkeit

ROI

Return of Invest → Hole ich meine investierten Kosten über die Zeit wieder zurück und wenn ja wann?. Bei der Berechnung der gibt es eine Fallunterscheidung. Folgende Fälle gilt es zu unterscheiden.

Betrachtung ob Neuanschaffung oder Ersatzinvestition

- Neuanschaffung
 - o Einmalige Kosten und jährliche Betriebskosten
 - o Zukünftiger Nutzen
- Ersatzinvestition
 - o Einmalige Kosten
 - o Unterschied Betriebskosten neu/alt
 - o Unterschied Nutzen neu/alt

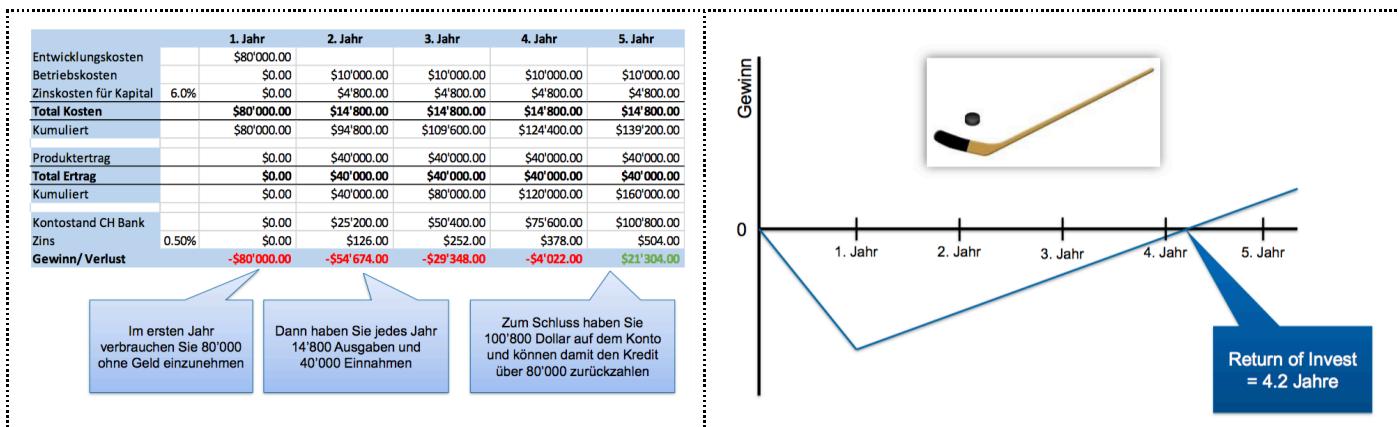
Betrachtung der Finanzierung

- Fremdfinanzierung
 - o Betrachtung der Zins- und Rückzahlungskosten
- Eigenfinanzierung
 - o Diskontierung (Zins + Risiko) einbeziehen

ROI Beispiel – Entwicklung einer neuen App

Ein amerikanischer Partner gibt Ihnen 80'000 Dollar als Darlehen, welches Sie für die Entwicklung der Fitness-App verwenden. Sie zahlen dafür ab dem zweiten Jahr jährlich 6 Prozent Zins für dieses 5-Jahres-Darlehen.

Sie brauchen genau ein Jahr für die Entwicklung, dann können Sie das App auf einem Server betreiben. Sie benötigen keine Lizenzen oder separate Hardware. Der Serverbetrieb (as a Service) kostet 10'000 Dollar pro Jahr. Pro App verdienen Sie einen Dollar. Sie rechnen nach Entwicklungsende vier Jahr lang mit je 40'000 bezahlten Downloads pro Jahr. Wenn Sie den jeweiligen Ertrag auf eine Schweizer Bank legen, kriegen Sie 0.5 Prozent Zins pro Jahr.



ROI Beispiel – WLAN Access Points ersetzen

Sie betreiben als Student nebenbei für mehrere Hotels jeweils ein WLAN Access Point Netzwerk. Die Hotels schätzen diesen Dienst sehr, weil sie wenig damit zu tun haben und die WLAN Codes an die Gäste weiterverkaufen können. Sie nehmen dadurch total jedes Jahr CHF 12'000.- ein und haben selber sehr geringe Betriebskosten von jährlich CHF 2'000.-.

Nun möchten Sie für CHF 29'000.- alle Access Points innerhalb von drei Monaten und dabei Modelle einkaufen, die nur noch CHF 1'000.- Betriebskosten pro Jahr verursachen. Das Ersetzen ist innerhalb drei Monate möglich. Während dem Vierteljahr sind die Betriebskosten gemischt.

Leider fehlt Ihnen momentan das Geld für den Ersatz. Mittels Crowdflending haben sie die CHF 29'000.- als Darlehen gesammelt, 4.3% Zins pro Jahr. Wann erreichen Sie den ROI?

	1. Jahr	2. Jahr	3. Jahr	4. Jahr	5. Jahr
Entwicklungskosten	CHF 29'000.00				
Betriebskosten	CHF 1'250.00	CHF 1'000.00	CHF 1'000.00	CHF 1'000.00	CHF 1'000.00
Zinskosten für Kapital	4.3%	CHF 1'247.00	CHF 1'247.00	CHF 1'247.00	CHF 1'247.00
Total Kosten	CHF 31'497.00	CHF 2'247.00	CHF 2'247.00	CHF 2'247.00	CHF 2'247.00
Kumuliert	CHF 31'497.00	CHF 33'744.00	CHF 35'991.00	CHF 38'238.00	CHF 40'485.00
Produktertrag	CHF 12'000.00	CHF 12'000.00	CHF 12'000.00	CHF 12'000.00	CHF 12'000.00
Total Ertrag	CHF 12'000.00	CHF 12'000.00	CHF 12'000.00	CHF 12'000.00	CHF 12'000.00
Kumuliert	CHF 12'000.00	CHF 24'000.00	CHF 36'000.00	CHF 48'000.00	CHF 60'000.00
Gewinn/Verlust	CHF -19'497.00	CHF -9'744.00	CHF 9.00	CHF 9'762.00	CHF 19'515.00

Sie können das Crowdflending Darlehen nach ziemlich genau drei Jahren wieder zurückzahlen.

Diskontierung – Was ist das?

Die Diskontierung ist für einen Ingenieur ohne betriebswirtschaftliche Vorbildung ein schwer zu verstehendes Konstrukt. In diesem Modul behandeln wir das Thema so oberflächlich wie möglich und zwar mit folgenden Grundsätzen:

- 100 Franken heute sind nicht 100 Franken morgen
- Gewinne in der Zukunft sind somit weniger wert als Gewinne heute
- Wenn ich Geld ausleihe, möchte ich Zins und eine Risikoprämie
- Jedes Unternehmen hat seinen eigenen internen Zinssatz, wenn es für «sich selber Geld ausleiht» und in ein Projekt investiert.
- **Diskontierungssatz = Interner Zinssatz + Risikozuschlag**

ROI Beispiel – Online Shop einer Handelskette

Eine bestehende Online Shop Software hat Betriebskosten in der Höhe von CHF 300'000.- jährlich. Die Entwicklung einer verbesserten Version kostet CHF 945'000.- und kann vom Unternehmen selber bezahlt werden, Diskontierungssatz beträgt 4.5%.

Die verbesserte Software hat Betriebskosten in der Höhe von CHF 200'000.-. Mit der verbesserten Software wird ihr Online Shop CHF 50'000.- pro Monat mehr Gewinn erzielen.

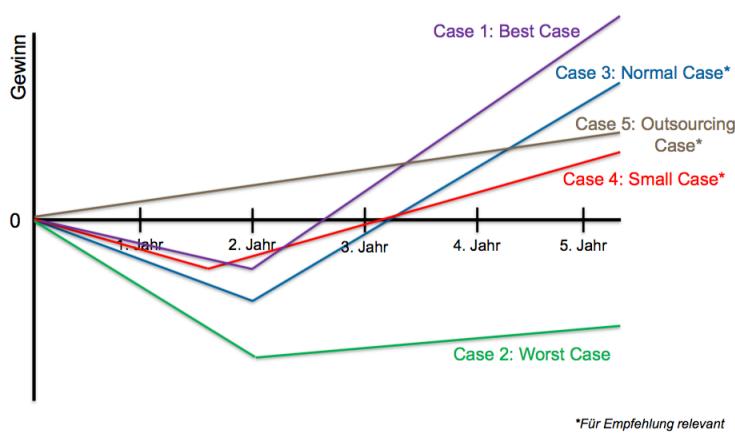
Sie benötigen für die Überarbeitung 6 Monate, in dieser Zeit fallen die Betriebskosten der neuen und alten Software an. Wann ist der ROI erreicht?

	1. Jahr	2. Jahr	3. Jahr	4. Jahr	5. Jahr
Entwicklungskosten	CHF 945'000.00				
Betriebskosten	CHF 450'000.00	CHF 200'000.00	CHF 200'000.00	CHF 200'000.00	CHF 200'000.00
Total Kosten	CHF 1'395'000.00	CHF 200'000.00	CHF 200'000.00	CHF 200'000.00	CHF 200'000.00
Gewinnsteigerung	CHF 300'000.00	CHF 600'000.00	CHF 600'000.00	CHF 600'000.00	CHF 600'000.00
Total Ertrag	CHF 300'000.00	CHF 600'000.00	CHF 600'000.00	CHF 600'000.00	CHF 600'000.00
Jahresrechnung (FCF)	CHF -1'095'000.00	CHF 400'000.00	CHF 400'000.00	CHF 400'000.00	CHF 400'000.00
Diskontierungsfaktor	4.50%	1	0.955	0.912025	0.870983875
Diskont (DCF)	CHF -1'095'000.00	CHF 382'000.00	CHF 364'810.00	CHF 348'393.55	CHF 332'715.84
Gewinn/Verlust (CCF)	CHF -1'095'000.00	CHF -713'000.00	CHF -348'190.00	CHF 203.55	CHF 332'919.39

Für Finanzkenner:
 FCF = Free Cash Flow
 DCF = Discounted Cash Flow
 CCF = Cumulative Cash Flow

Der ROI ist Ende Dezember
des 4. Jahres erreicht

Business Cases



Ein Business Case vergleicht mehrere Geschäftsfälle oder auch mehrere Varianten miteinander. Im Prince2 Standard muss vor jedem Projekt ein Business Case gemacht werden. Ein Business-Case beinhaltet

- Management Summary
- Abgrenzung
- Kosten der verschiedenen Cases
- Monetäre Nutzen der Cases
- ROI Berechnung pro Case

- Quantitative Nutzen pro Case
- Risiken und Changen sofern noch nicht einberechnet, - Empfehlung Entscheidung

Finanzierung und Liquidität

Ein Projekt muss finanziert werden, das heisst die notwendigen finanziellen Mittel müssen am richtigen Ort und im richtigen Moment bereits sein. Eine Finanzierung kann auf zwei Arten passieren:

- Interne Mittel des Unternehmens
- Externe Mittel durch Ertrag am Markt (z.B. Teilzahlung durch Kunde)

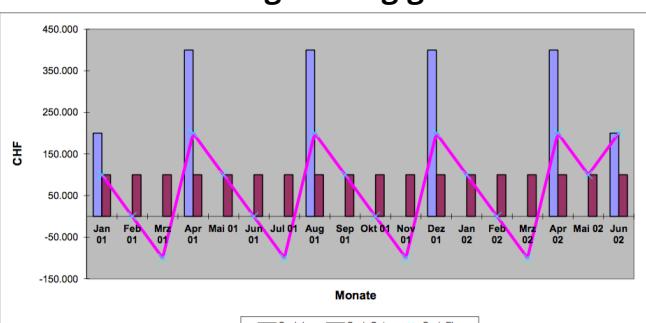
Projekte die extern finanziert werden, wie die internen Mittel nicht ausreichen sind per se Hochrisiko-Projekte (Konkursrisiko). Extern finanzierte Projekte sind tendenziell rentabler und unterliegen einer schärferen Kontrolle.

Projektliquidität

Die Liquidität beschreibt die vorhandenen flüssigen (finanziellen) Mittel. Bei intern finanzierten Projekten muss das Unternehmen selber die notwendigen Liquidität bis zum Schluss bereitstellen, meistens werden die Mittel phasenweise freigegeben. Bei extern finanzierten Projekten müssen die stetigen Einnahmen jeweils vor den Ausgaben erzielt werden. Wenn Sie für einen Kunden arbeiten, dann gibt es zwei mögliche Zahlungsmodalitäten:

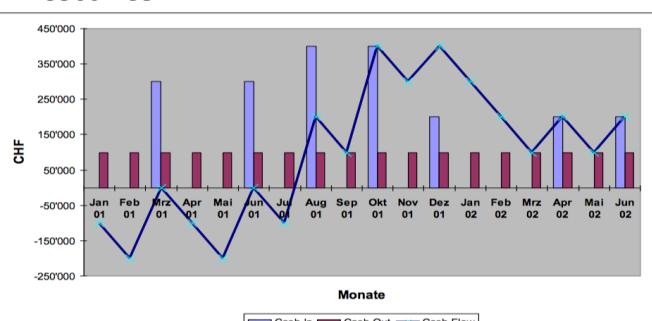
- Gestaffelte Zahlung, abhängig von Zeit und Fortschritt (z.B. Meilensteine)
- Regelmässige Zahlung nach Aufwand (z.B. monatlich)

Gestaffelte Zahlung abhängig von der Zeit



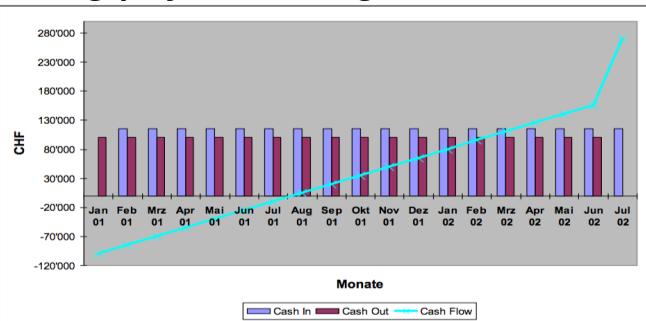
In diesem Beispiel reicht die externe Finanzierung jeweils Ende Quartal nicht mehr aus, weswegen auch eine interne Finanzierung von ca. CHF 100'000.- notwendig ist.

Gestaffelte Zahlung abhängig von den Milestones



Hier ist zu Beginn ebenfalls eine interne Finanzierung notwendig, wobei es abhängig von dem Projektfortschritt ist, wann das Projekt nur noch extern finanziert wird.

Auftragsprojekt – Zahlung nach Aufwand



Hier zeigt sich, dass nur zu Beginn eine interne Finanzierung von ca. CHF 100'000 notwendig ist, nach einem halben Jahr ist das Projekt komplett extern finanziert.

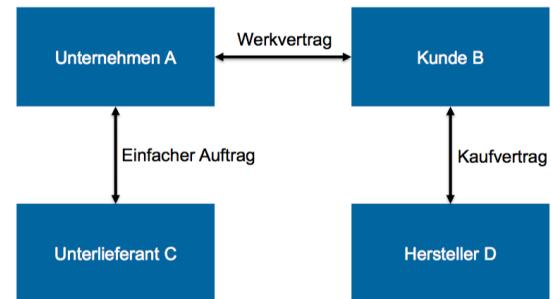
Wichtige Punkte

- Zahlungsziel wird meistens NICHT eingehalten (Zahlungsmoral)
- Rechtzeitige Verrechnung der Projektleistungen
- Vorfinanzierung
- Wie wir gesehen haben, benötigen die meisten Projekte eine interne Vorfinanzierung. Wenn diese intern nicht möglich ist, gibt es nebst Kunden folgende weitere externen Möglichkeiten:
 - o Investitionskredit bei einer Bank (4%-8%)
 - o Betriebskredit/Kontokorrent ausschöpfen (4%-8%)
 - o Kapitalerhöhung
 - o Crowdsharing
- Bauen Sie wenn immer möglich eine Liquiditätsreserve ein
- Liquiditätsplanung soll mit dem Risikomanagement verknüpft werden
- Liquidität ist wie die Luft zum Atmen für den Menschen!

Beschaffung und Verträge

Sie können selten alle Mittel für ein Projekt selber zur Verfügung stellen sondern müssen daher auch Mittel projektspezifisch beschaffen, dies können sein:

- Hardware
 - o Kaufvertrag nach Art. 184ff OR
 - o Miete (Innominatvertrag)
- Softwarelizenzen (Lizenzvertrag)
- Externe Dienstleistungen
 - o Einfacher Auftrag nach Art. 394ff OR
 - o Werkvertrag nach Art. 363ff OR



Was ist ein Vertrag?

Eine gegenseitige Verpflichtung (Obligation) von zwei oder mehreren Parteien. Es gibt drei mögliche Vertragsformen: Schriftlich, mündlich und konkludent. In der Schweiz gibt es nur für wenige Verträge eine Formvorschrift, daher können Sie können als zum Beispiel einen Arbeitsvertrag auf einem Bierdeckel oder mündlich vereinbaren. Konkludente Verträge, also ohne Schrift und Wort, sind eher für kleinere Verträge vorgesehen, z.B. Besteigen eines Postautos (Billett beim Fahrer lösen).

Unterscheidung der Verträge für Dienstleistungen

Werkvertrag	Einfacher Vertrag
<ul style="list-style-type: none"> - Lieferergebnis geschuldet - Preis ist festgelegt - Fixer Termin mit Pönale - Lieferant ist Projektleiter - Erbringung ortsunabhängig - Abnahme der Lieferergebnisse - Changes gegenseitig - Keine gesetzliche Haftung - Risiko beim Lieferanten 	<ul style="list-style-type: none"> - Arbeitsleistung geschuldet - Preis nach Aufwand - Fixer Termin ohne Pönale - Kunde ist Projektleiter - Erbringung idR beim Kunden - Keine Abnahme der Arbeit - Changes einseitig - Beratungshaftung - Risiko beim Kunden

IT Service Management nach ITIL v3

Einleitung

Motivation

Unternehmen sind zunehmend von der IT abhängig (wachsender Bedarf an IT Services). Zudem sind die Informationen meist das wertvollste Gut einer Unternehmung. Es ist ein Trend weg von der Entwicklung der IT Anwendungen hin zum Management von IT Services (SaaS). Da unterstützt das ITIL (Information Technology Infrastructure Library), ein defacto Standard welcher Best Practices behinhaltet.

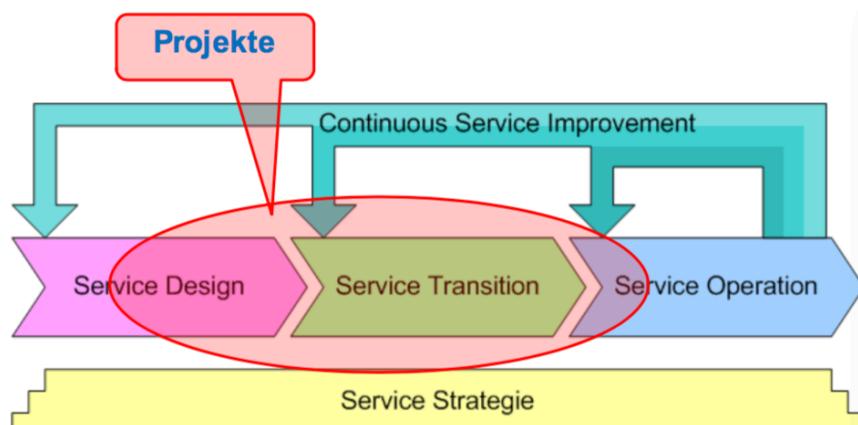
Historie

1989	ITIL V1 80486 und i860 Prozessoren Portabler Mac Nintendo Game Boy HTML wird erfunden und www wird entwickelt
2000	ITIL V2 Tablets und Mobiles Erstes Mobile mit Kamera PlayStation 2 USB Flash Drive Windows 2000
2007	ITIL V3 iPhone Amazon Kindle Dropbox Windows Vista

Ziele von IT Service Management

- Zuverlässige und konsistente IT Services
 - o Mit hoher Qualität
 - o Zu tragbaren Kosten
- Abstimmung der IT Services auf die Geschäftsanforderungen
 - o Kostenoptimierung
 - o Qualität Sicherung (steuerbares Qualitätsniveau)
 - o Betriebsoptimierung
- Trennung von Entwicklung und Betrieb → neuer Trend: Dev-Ops
- Definierte Schnittstelle zum Business (Kunde / Benutzer) → neuer Trend: Produkte-Teams

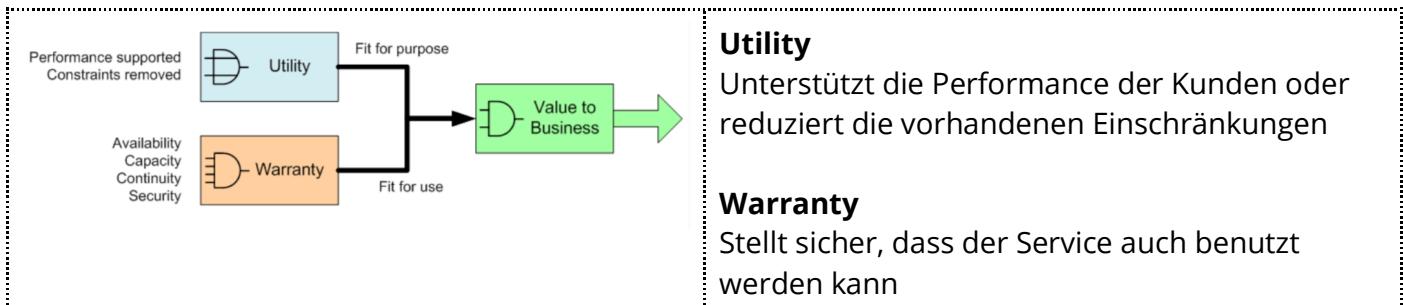
ITIL Übersicht v3



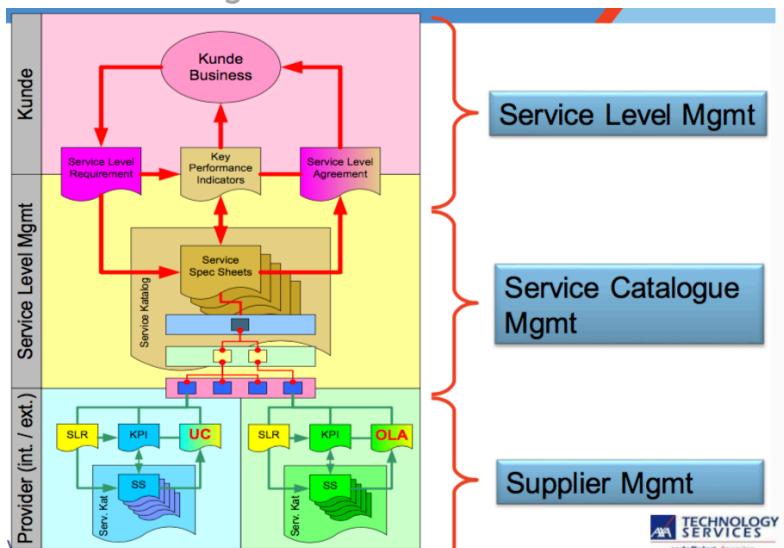
Service Design

- Service Level Management (Vereinbarungen durch SLA (Service Level Agreements))
- Unterstützt durch alle anderen Service Design Prozesse
 - o Service Catalogue Management (Service Katalog über alle Services)
 - o Capacity Management (Verwaltung und Steuerung der Ressourcen)
 - o IT Service Continuity Management (Sicherstellen der vitalen IT Services im K-Fall)
 - o Availability Management (Gewährleisten eines kosteneffektiven Verfügbarkeitsniveaus)
 - o IT Security Management (Vertraulichkeit, Integrität, Verfügbarkeit der Daten)
 - o Supplier Management (Vereinbarung mit Providern)

Business Nutzen

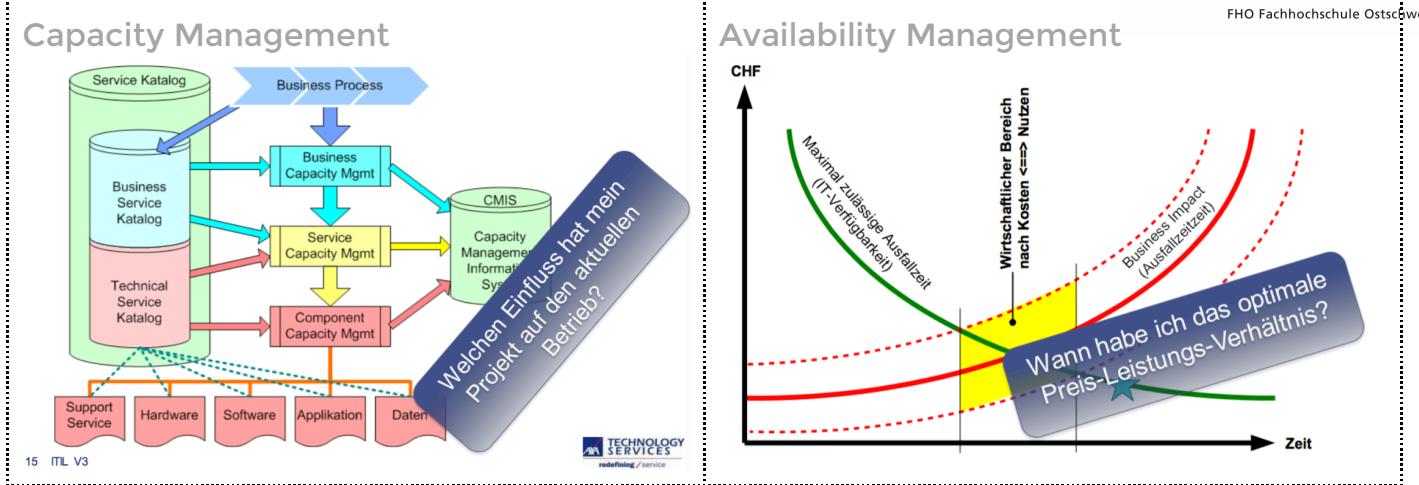


Service Management



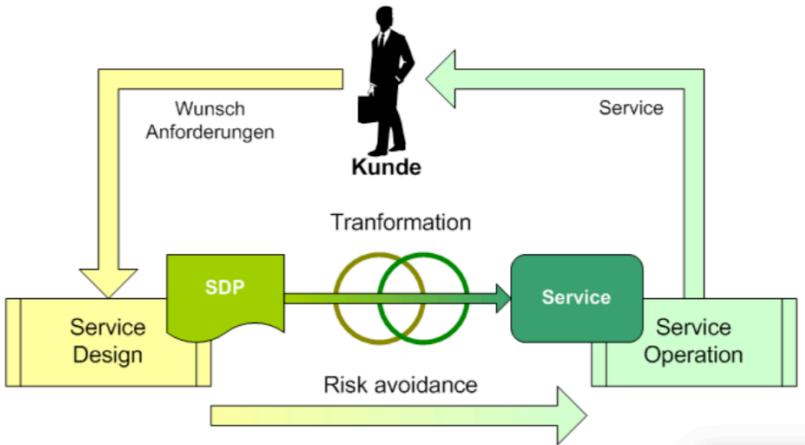
Key Performance Indikatoren

- Verfügbarkeit des Service während der Betriebszeit
 - o In Prozent
 - o Intern: Wie viel Arbeitszeit wurde durch nicht Verfügbarkeit verwendet?
 - o Extern: Wie viele Minuten stand der Service nicht zur Verfügung?
- Performance (Reaktionszeit des Systems auf Anfragen)
- Lösungszeit bei Incidents
 - o Wie viele Incidents konnten in der vereinbarten Zeit gelöst werden?
- Frist Fix Rate / Lost Rate
 - o Wie viele Anfrufe am Service Desk konnten nicht entgegen genommen werden?
 - o Wie viele Anfragen und Störungen konnten direkt am Telefon erledigt werden?
- Lieferzeiten
 - o Wie viele Bestellungen (HW, SW, Zugriffsrechte) konnten in der vereinbarten Zeit erledigt werden?



Service Transition

Service Transition - Kreislauf



Transition Planning & Support

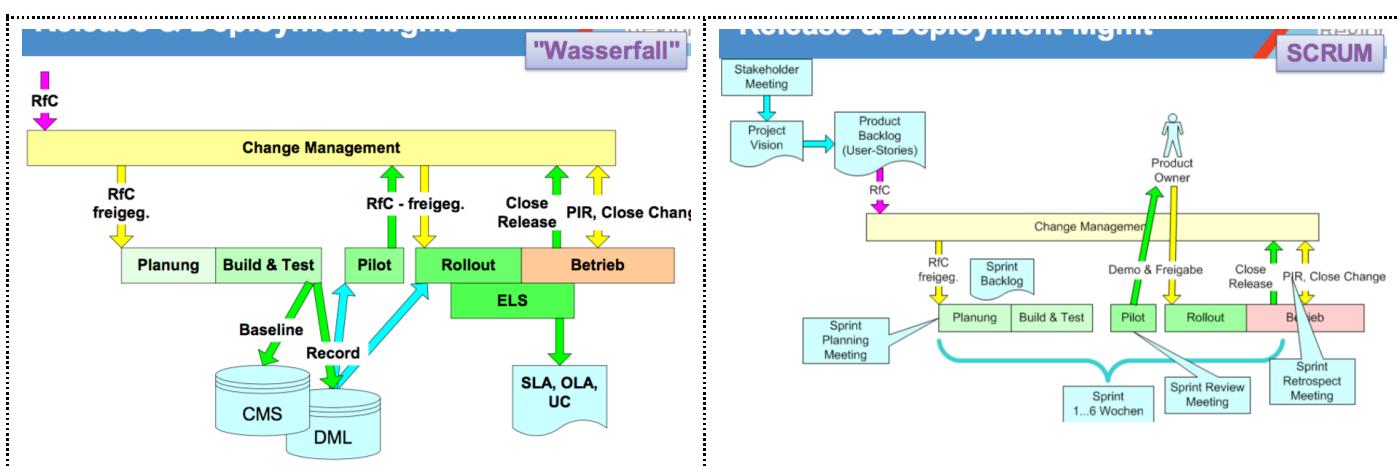
Allgemeine Planung der Überführung neuer oder geänderter Services in die Produktion.

Change Management

Änderungen kontrolliert und mit geringsten möglichen Auswirkungen auf den Geschäftsbetrieb durchführen

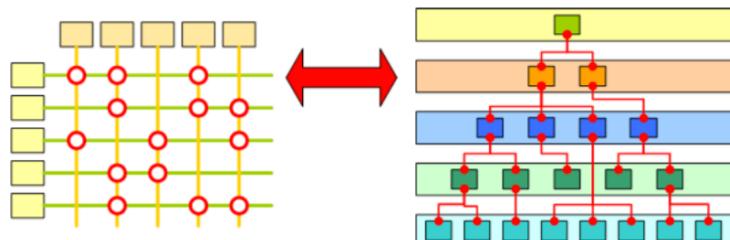
Release & Deployment Management

Grössere Veränderungspakete projekthaft entwickeln und durchführen → ausrollen, Validation & Testing, Evaluation



SCRUM – Nach jedem Sprint sind alle Dokus auf dem aktuellen Stand.

Erstellen und Pflegen eines logischen Abbildes der «IT Welt» genauer gesagt: Erstellen und Pflegen eines vollständigen Configuration Management System (CMS) mit allen Service Assets, Configuration Items und deren Beziehungen.



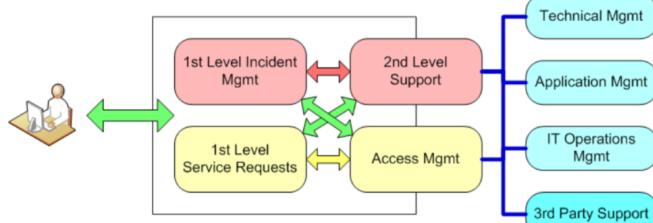
Knowledge Management

Richtige Information zur richtigen Zeit am richtigen Ort

Service Operation

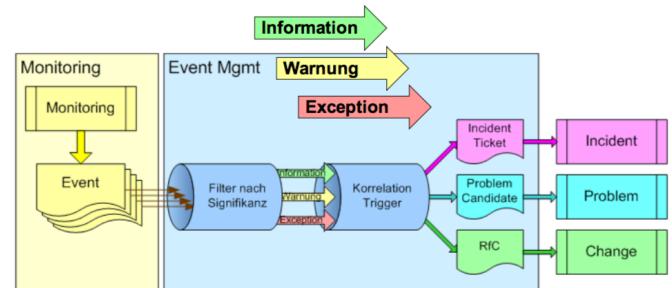
Service Desk

Erreichbarkeit der IT sicherstellen
(Entgegennehmen von Störungen und Anfragen)



Event Management

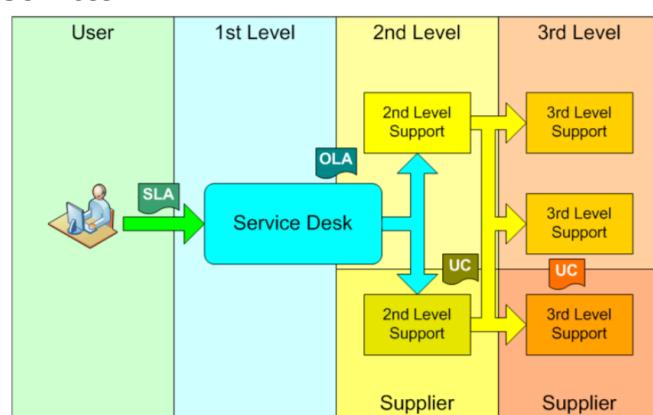
Überwachen und eskalieren der Events entsprechend ihrer Signifikanz



Normal = Information, Unusual = Warning,
Exceptional = Exception

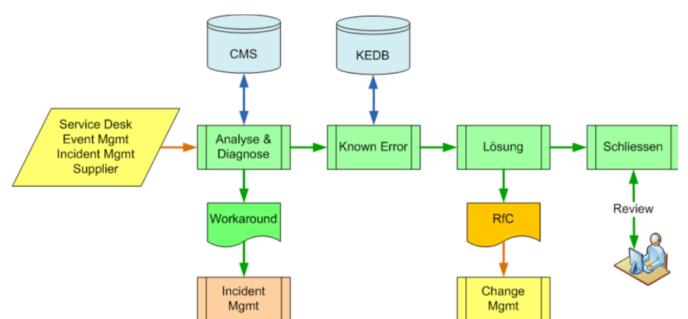
Incident Management

Schnellstmögliche Behebung von Incidents,
Incident = Beeinträchtigung des vereinbarten
Services



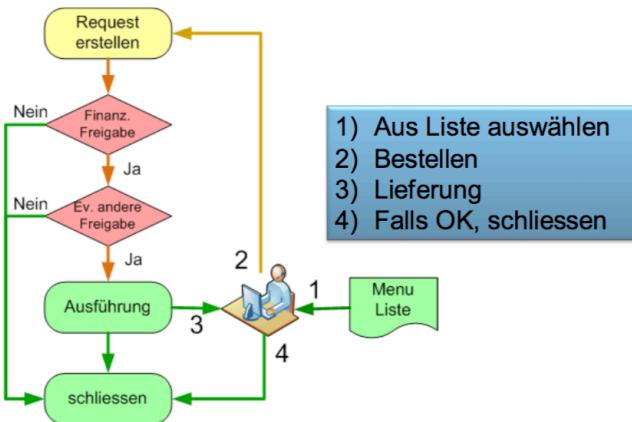
Problem Management

Nachhaltiges Vermeiden von Incidents, Problem = Störung unbekannter Ursache



Request Fulfillment

Ausführen von Bestellungen der Benutzer



Betriebsfunktionen

Technical Mgmt:

Unterstützt mit seinem Wissen das Mgmt der IT Infrastruktur. Es stellt die Spezialistengruppe innerhalb der IT dar.

IT Operations Mgmt:

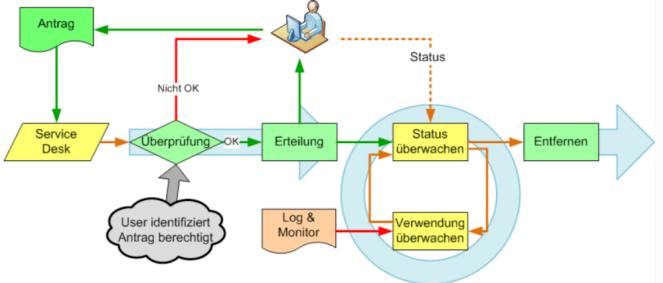
Führt die täglichen Aktivitäten aus, die für einen stabilen Betrieb der IT Infrastruktur nötig sind.

Application Mgmt:

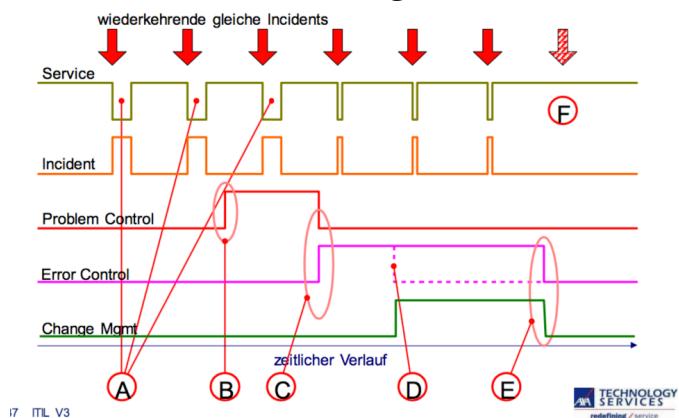
Ist verantwortlich für das Mgmt der Applikationen über den gesamten Lifecycle.

Access Management

Erteilen und Verwalten von Zugriffsrechten



Incident – Problem – Change



Tipps aus der Praxis

Der Projekterfolg zeigt sich im Betrieb des Service! Daher:

Für konventionelle Projekte

- Entscheidungsträger des Operation frühzeitig abholen → Stakeholder
 - Bereitstellung der Infrastruktur durch Leute des Betriebs im Projektteam verpflichten
 - «Nicht funktionale» Anforderungen (Capacity, Availability, Security, Continuity) von Anfang an berücksichtigen

Für agile DevOps Teams

- Verfügbare Standards verwenden (Cloud)
 - Infrastruktur frühzeitig beim Provider sicherstellen
 - Operation berücksichtigen (Service Zeiten)
 - Supportfunktionen einbeziehen (Service Desk, Monitoring, ...)
 - «Nicht funktionale» Anforderungen (Capacity, Availability, Security, Continuity) von Anfang an berücksichtigen

Risikomanagement

Ziele des Kapitels

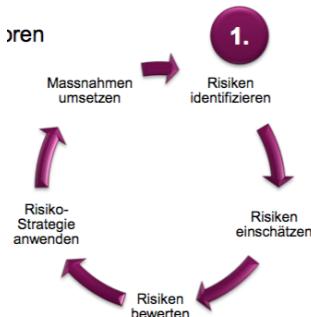
Lerninhalte

- Analyse
- Bewertung
- Bewältigung
- Risiken aus der Praxis
- Chancen

Ziele: Die Studierenden...

- identifizieren Risiken und schätzen diese ein
- lernen wie Risiken beurteilt werden
- üben die Bewältigung von Risiken
- wissen wie das Risikomanagement in der Praxis funktioniert

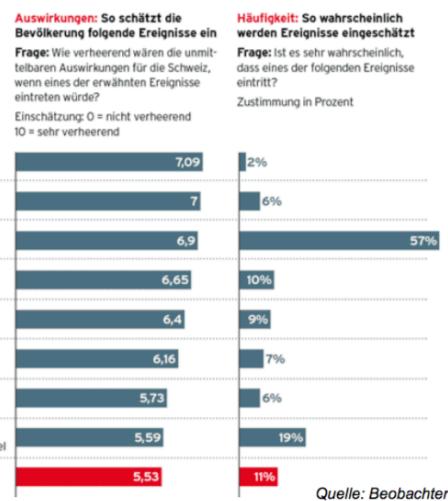
Analyse



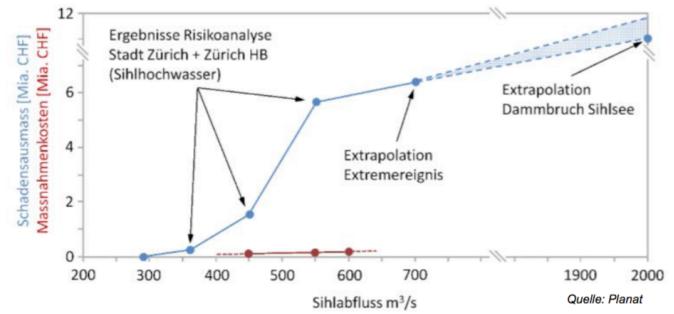
Risiken und Chancen sind Unsicherheitsfaktoren, welche das gewünschte Ergebnis negativ oder positiv beeinflussen können. Das Management der Risiken muss Bestandteil im Projektmanagement sein, das Management der Chancen sieht man eher stelten, ist jedoch gerne gesehen.

Ein Risiko besteht im Wesentlichen aus zwei Faktoren. Der Eintretens Wahrscheinlichkeit (oder Häufigkeit) und dem Schadensausmass (oder Auswirkung). Risiken müssen zuerst identifiziert werden.

Beispiele aus der Schweiz



Hochwasser Sihl



Zum Zeitpunkt der Studie entsprach der Abfluss von 450 m³/s dem 300-jährlichen Ereignis (0.0033%) und 550 m³/s dem 500-jährlichen (0.002%).

Identifizieren

Die in der Fachwelt vier häufigsten Analyse für das Identifizieren sind:

Impact Analyse

Dies ist eine geläufige Alternative zu einer vollständigen Risikoanalyse, mit der ich die wichtigsten Auswirkungen eines Projekts untersuche und deren Störung als Risiko strukturiere

Bedrohungsanalyse

Damit untersuche ich das Projekt gegen alle möglichen Bedrohungen aus einem Bedrohungskatalog und strukturiere die relevanten als Risiko

Schwachstellenanalyse

Aus der Erfahrung ähnlicher Projekte identifiziere ich die grössten Schwachstellen eines Projekts und strukturiere diese als Risiko

Beliebige Kombinationen

Jegliche Kombinationen sind möglich, wobei eine Analyse von Impact, Bedrohung und Schwachstellen einer vollständigen Analyse entspricht

Einschätzen

Sie müssen jedes Risiko qualitativ (hoch, mittel, tief) oder quantitativ (Zeit und Kosten) einschätzen, dies können Sie auf verschiedene Arten tun:

- Schätzung der maximalen Werte
- Schätzung der mittleren Werte
- Schätzung der Werte mittels statistischen Verteilungsfunktionen

Schätzen Sie zuerst die Eintretens Wahrscheinlichkeit und dann das Schadensausmass, entweder Bottom-Up oder Top-Down

- Bottom-Up bedeutet eine Analyse mit Detail die sie dann zu einer grossen Analyse aggregieren
- Top-Down bedeutet eine übergreifende grobe Analyse, die sie dann Schritt für Schritt detaillieren

Einschätzungsbeispiel «Serverausfall»

Bottom-Up quantitative Bedrohungsanalyse mittels Ereignisbaum

Eintretenswahrscheinlichkeit

- R1: Ausfall Harddisk im Server → alle 36 Monate
- R2: Ausfall Stromversorgung Server → alle 36 Monate
- R3: Unbeabsichtigtes Herunterfahren → alle 36 Monate
- R1-R3: Alle **12 Monate** fällt der Server aus



Der Ergebnisbaum fasst also von den einzelnen Ereignissen kommend ähnliche Risiken zusammen und gruppiert diese so, wobei an Schärfe verloren geht.

Schadensausmass

- R1: Ausfall Harddisk im Server → CHF 5'000
- R2: Ausfall Stromversorgung → CHF 2'000
- R3: Unbeabsichtigtes Herunterfahren → CHF 500
- R1-R3: Ein Ausfall kostet durchschnittlich **CHF 2'500**



Bewertung

Durch die Bewertung versuchen Sie die erhaltenen Risikowerte in einen Kontext zu bringen und diese auch wenn möglich terminlich zu fixieren oder priorisieren. Beispiele

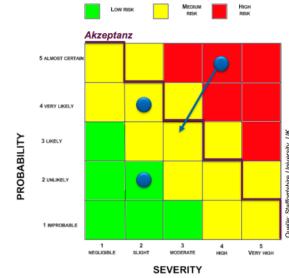
Beispiel 1 In einem grossen Entwicklungsprojekt mit 30 Mitarbeitenden wird der Personalausfall aufgrund der Fehltage im Unternehmen auf qualitativ «hoch» angesetzt. Eine Betrachtung im Kontext zeigt aber auf, dass nur ein Teilprojekt mit 5 Personen auf dem kritischen Pfad einen leichten Verzug hat. Mit gesundem Menschenverstand drängt sich daher eine Reduktion auf «mittel» auf.

Beispiel 2 In einem Infrastrukturprojekt werden Mitarbeitende eines Modehauses mit einem speziell gehärteten Tablet versorgt, auf welchem das zu entwickelnde App Beratungsvorschläge erteilt. Die Tablets müssen spätestens 3 Monate vor Projektende geliefert werden. Das Risiko einer Verspätung wurde qualitativ auf «mittel» angesetzt. Hier drängt sich eine terminliche Fixierung des Risikos auf.

Risikoliste und Risikograph

Führen Sie in einem Projekt immer eine Risikoliste, diese kann ihre «Lebensversicherung» sein, wenn etwas schief gehen sollte. Zum Schluss der Bewertung tragen Sie alle Risiken auf einem Graphen ein. Die Risikostrategie ihres Unternehmens definiert die Risiko Akzeptanzlinie. Alle Risiken über der Linie reduzieren sie priorität und auf jeden Fall.

Risk ID	Risiko	Maßnahme	Kosten der vorbeugende n Maßnahme in CHF	Max. Schaden In CHF	Wahrscheinlichkeit des Eintretens	Gewichteter Schaden in CHF	Priorität	Vorweg genommen Entscheide wenn Risiko eingetreten
R01	Personalausfall	Stellvertreterungen sicherstellen	10'000	50'000	50%	25'000	normal	
R02								
R03								
R04								
R05								
R06								
R07								
R08								
R09								
R10								
R11								
R12								
Total Kosten in Arbeitspaketen enthalten			10'000					
Total Rückstellungen						25'000		



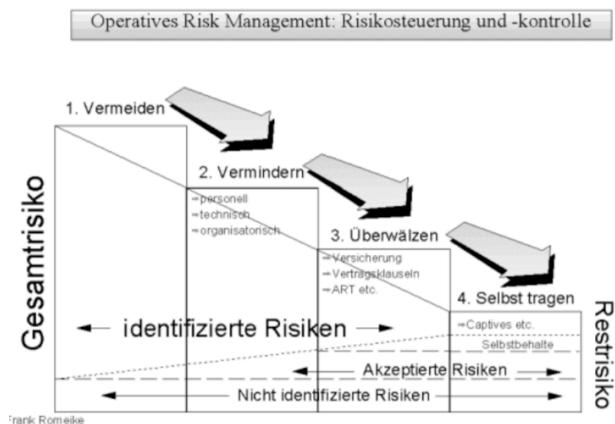
20 Fehler nach Lindecker (Software Projects, 1989)

- Erster Fehler: Die „prometheische“ Selbstüberschätzung
- Zweiter Fehler: Man weiss nicht, was man eigentlich will
- Dritter Fehler: Unrealistische Leistungszusage in der Vorprojektphase
- Vierter Fehler: Unterschätzen der Komplexität der Aufgabenstellung
- Fünfter Fehler: Fehlende Normen und Standards
- Sechster Fehler: Mangelnde Dokumentation des abzulösenden Systems
- Siebter Fehler: Ungenügend strukturierte Projektorganisation
- Achter Fehler: Mangelnde Qualifikation der Projektmitarbeiter
- Neunter Fehler: Die Illusion des Maurerdreisatzes (Chinesenprinzip)
- Zehnter Fehler: Fehlende Motivation des Projektteams

- Elfter Fehler: Zu wenig Standfestigkeit gegenüber den Anwender
- Zwölfter Fehler: Diskontinuität in der Projektdetaillierung
- Dreizehnter Fehler: Nichteinfrieren der Projektspezifikation
- Vierzehnter Fehler: Nichteinhalten des festgelegten Phasenplans
- Fünfzehnter Fehler: Methoden diskutieren statt anwenden
- Sechzehnter Fehler: Fehlende Software-Werkzeuge
- Siebzehnter Fehler: Mangelhaftes Datendesign
- Achtzehnter Fehler: Späte Fehlererkennung
- Neunzehnter Fehler: Nicht verbindlich festgelegte Abnahmeprozedere
- Zwanzigster Fehler: Fehlender Durchgriff auf die Organisationsentwicklung

Bewältigung

4. Maßnahmen zum pro-aktiven Risk Management



Vorbeugende Massnahmen zur Risikoreduktion dürfen im Wesentlichen nicht teurer sein, als das Risiko selber. Nehmen Sie sich wenige aber griffige Massnahmen aufs Mal vor und kommunizieren Sie dies im Projekt. Akzeptieren Sie die Restrisiken, wenn keine Massnahmen mehr von Nutzen sind. Wenn das Risiko trotzdem eintreffen sollte, können Sie bereits Entscheidungen vorweg vorbereiten.

Beispiel von Massnahmen

Gegen Personalausfall Jeder Projektmitarbeiter hat einen Stellvertreter und informiert diesen wöchentlich über den Stand der Dinge.

Gegen schlechte Datenqualität Zusätzliche Tests überprüfen systematisch die Anzahl Records und stichprobenweise die Inhalte der Records.

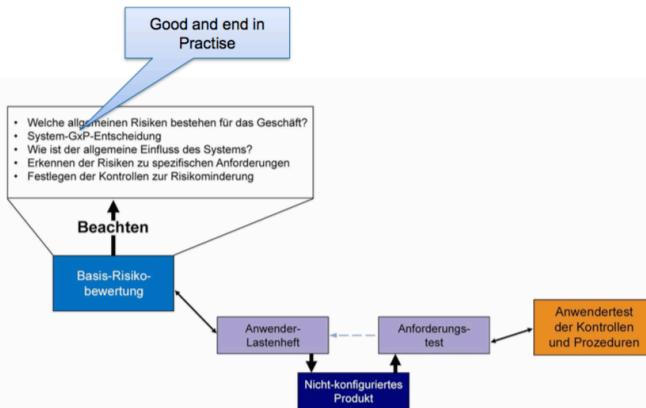
Gegen unklare Anforderungen Zusätzliche QS Stelle im Projekt

Gegen Währungsrisiken Das Projektvolumen im Euro wird mit einem Hedge abgesichert.

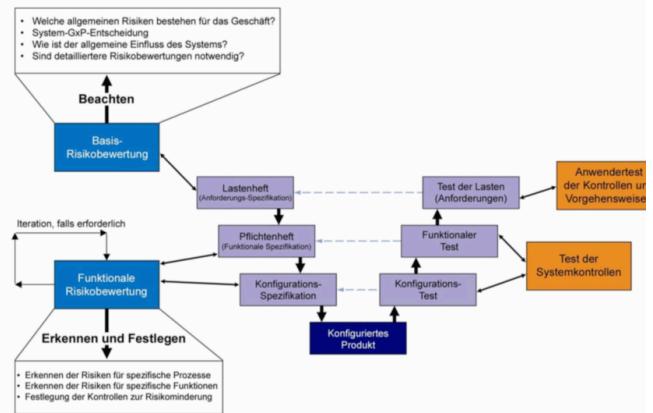
Top-Ten Risiken einer Software-Entwicklung

Risikoelement	Risikomanagement-Techniken	Risikoelement	Risikomanagement-Techniken
1 Personelle Defizite	<ul style="list-style-type: none"> Hochtalentierte Mitarbeiter einstellen Teams zusammenstellen 	6 Kontinuierliche Anforderungsänderungen	<ul style="list-style-type: none"> Hohe Änderungsschwelle Inkrementelle Entwicklung (Änderungen auf spätere Erweiterungen verschieben)
2 Unrealistische Termin- und Kostenvorgaben	<ul style="list-style-type: none"> Detaillierte Kosten- und Zeitschätzung mit mehreren Methoden Produkt an Kostenvorgaben orientieren Inkrementelle Entwicklung Wiederverwendung von Software Anforderungen streichen 	7 Defizite bei extern gelieferten Komponenten	<ul style="list-style-type: none"> Leistungstest Inspektionen Kompatibilitätsanalyse
3 Entwicklung von falschen Funktionen und Eigenschaften	<ul style="list-style-type: none"> Benutzerbeteiligung Prototypen Frühzeitiges Benutzerhandbuch 	8 Defizite bei extern erledigten Aufgaben	<ul style="list-style-type: none"> Prototypen Frühzeitige Überprüfung Verträge auf Erfolgsbasis
4 Entwicklung der falschen Benutzungsschnittstellen	<ul style="list-style-type: none"> Prototypen Aufgabenanalyse Benutzerbeteiligung 	9 Defizite in der Echtzeitleistung	<ul style="list-style-type: none"> Simulation Leistungstest Modellierung Prototypen Instrumentierung Tuning
5 Vergolden (über das Ziel hin-ausschiessen)	<ul style="list-style-type: none"> Anforderungen streichen Prototypen Kosten/Nutzen-Analyse Entwicklung an den Kosten orientieren 	10 Überfordern der Softwaretechnik	<ul style="list-style-type: none"> Technische Analyse Kosten/Nutzen-Analyse Prototypen

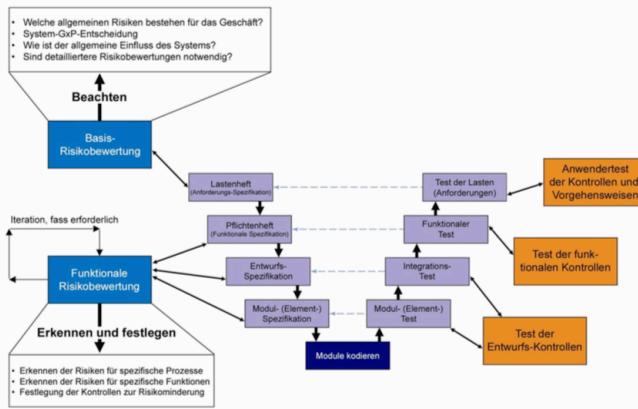
Risikobasiertes Vorgehen für ein nicht konfiguriertes Produkt



Risikobasiertes Vorgehen für ein konfiguriertes Produkt



Risikobewertung für eine kundenspezifische Applikation



Chancen

Chancen sind das Gegenteil von Risiken, überlegen Sie sich auch hier ob es welche gibt und welche **Eintretens Wahrscheinlichkeit** und **Nutzen** es gibt. Typische Chance sind:

- Neue Hardware-Generation machen Tuning überflüssig
 - Entsprechende Klassen werden auch von einem anderen Kunden verlangt (Kostenteilung)

Mit einer SWOT Analyse macht man in der Betriebswirtschaft meist einen Gegenüberstellung.

Lerninhalte

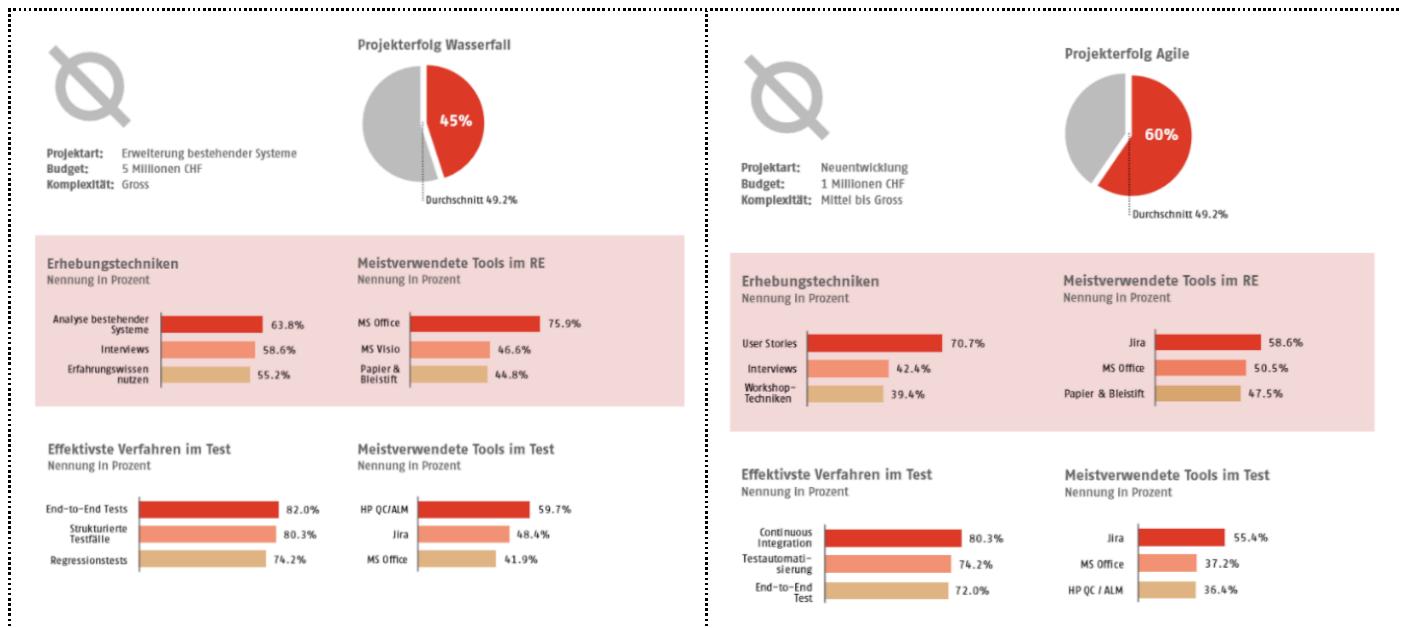
- Motivation für Agilität
- Agiles Manifest
- Verwendete Methoden

Ziele: Die Studierenden...

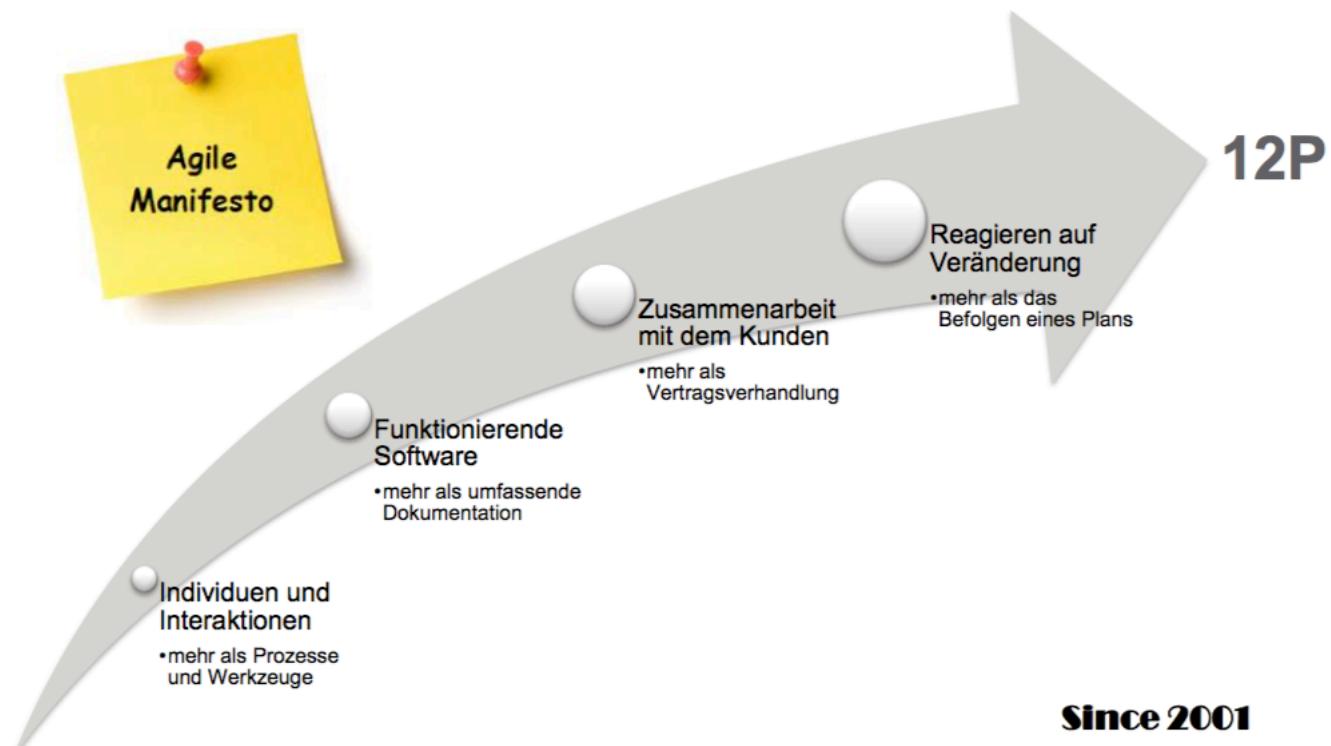
- lernen das Agile Manifest kennen
- verstehen die Motivation hinter dem agilen Vorgehen
- verschaffen sich einen Überblick über agile Methoden

Motivation für Agilität

Meist ist der Plan eine gerade Strasse, in der Realität trifft man dann aber eine Hügellandschaft mit weiteren Herausforderungen an. Die Agilität kann hier helfen. Die beiden folgenden Statistiken zeigen den Unterschied zwischen Wasserfall und Agil in der Schweiz.



Je grösser das Projekt, desto höher ist die Wahrscheinlichkeit zu scheitern. Dies aufgrund von vielen Abhängigkeiten, Ursache – Wirkung nicht erkannt, Fachwissen zu tief oder Anforderungen unklar. Daher Iterationen anstatt unpräzise Analyse zu beginn. Die Wahrscheinlichkeit lässt sich so von 3 Prozent auf 18 Prozent steigern bei grösseren Projekten.



Zwölf Prinzipien

Unsere höchste Priorität ist es, den Kunden durch frühe und kontinuierliche Auslieferung wertvoller Software zufrieden zu stellen.	Heisse Anforderungsänderungen selbst spät in der Entwicklung willkommen. Agile Prozesse nutzen Veränderungen zum Wettbewerbsvorteil des Kunden.
Liefere funktionierende Software regelmäßig innerhalb weniger Wochen oder Monate und bevorzuge dabei die kürzere Zeitspanne.	Fachexperten und Entwickler müssen während des Projektes täglich zusammenarbeiten.
Errichte Projekte rund um motivierte Individuen. Gib ihnen das Umfeld und die Unterstützung, die sie benötigen und vertraue darauf, dass sie die Aufgabe erledigen.	Die effizienteste und effektivste Methode, Informationen an und innerhalb eines Entwicklungsteams zu übermitteln, ist im Gespräch von Angesicht zu Angesicht.
Funktionierende Software ist das wichtigste Fortschrittsmaß.	Agile Prozesse fördern nachhaltige Entwicklung. Die Auftraggeber, Entwickler und Benutzer sollten ein gleichmäßiges Tempo auf unbegrenzte Zeit halten können.
Ständiges Augenmerk auf technische Exzellenz und gutes Design fördert Agilität.	Einfachheit - die Kunst, die Menge nicht getaner Arbeit zu maximieren - ist essenziell.
Die besten Architekturen, Anforderungen und Entwürfe entstehen durch selbstorganisierte Teams.	In regelmäßigen Abständen reflektiert das Team, wie es effektiver werden kann und passt sein Verhalten entsprechend an.

Damit Prinzip 10, Einfachheit erreicht werden kann muss man immer alle Tests laufen lassen, es darf kein redundanter Code vorhanden sein, und die Methoden und Klassen müssen immer minimalistisch sein.

Inspect & Adapt	Transparenz
Selbstorganisation	Timeboxing

Inspect & Adapt

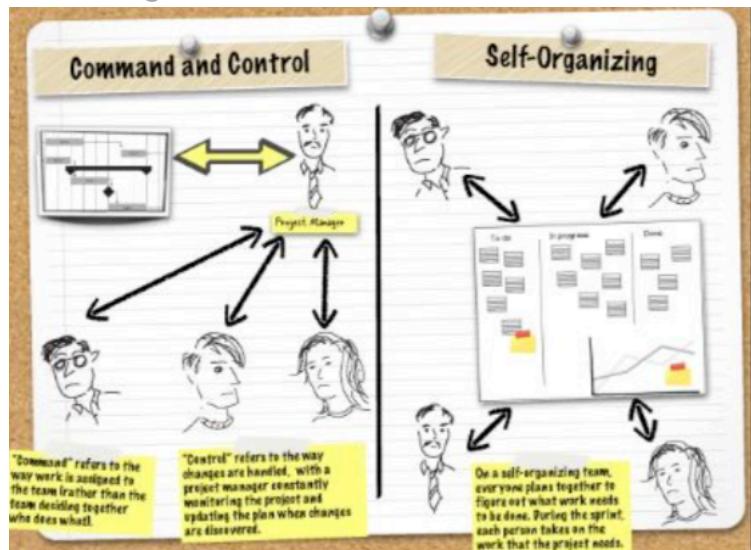


Sämtliche Bestandteile im Projekt werden regelmässig überprüft und wenn notwendig angepasst, daher auch wichtige Anforderungen dürfen jederzeit einfließen.

Transparenz

Informationen stehen jedem zur Verfügung. Es gibt keine versteckte Agenda oder alternative Fortschrittskontrollen. Dies verlangt nach einer Visualisierung der Daten, zum Beispiel mit Post-Ist auf einer Wand.

Selbstorganisation



Ein Unternehmen beauftragte Ken Schwaber (Erfinder von Scrum) agiles Projektmanagement einzuführen. Für ein Grossprojekt waren 140 Personen vorgesehen. Nun sollte man von Ken Schwaber wissen, wie die Projektorganisation aussehen sollte und wie viele Teams in welcher Grösse zu bilden sind. Seine Philosophie lautete : «Optimal Development Team size is small enough to remain nimble and large enough to complete significant work within a Sprint. Fewer than three Development Team members decrease interaction and results

in smaller productivity gains. Smaller Development Teams may encounter skill constraints during the Sprint, causing the Development Team to be unable to deliver a potentially releasable Increment. Having more than nine members requires too much coordination.”

Timeboxing

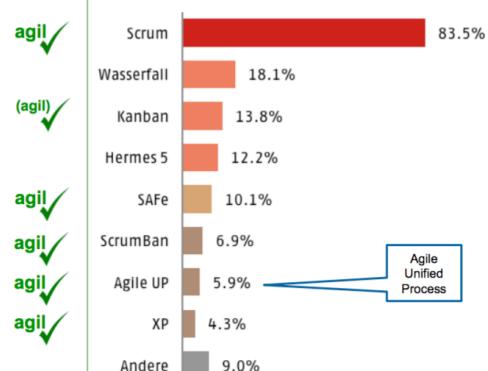
Zeit vor Inhalt	Inhalt vor Zeit
<p>Zeit vor Inhalt</p> <ul style="list-style-type: none"> • Fixe Zeitspanne, nicht länger und nicht kürzer • Nicht erledigte Arbeiten werden in die nächste Box eingeplant • Mitarbeitende gegen Einfluss von aussen abgeschottet 	<p>Inhalt vor Zeit</p> <ul style="list-style-type: none"> • Fixer Scope, darf nicht gekürzt oder erweitert werden • Arbeiten werden immer erledigt • Mitarbeitende gegen Einfluss von aussen abgeschottet

User Stories

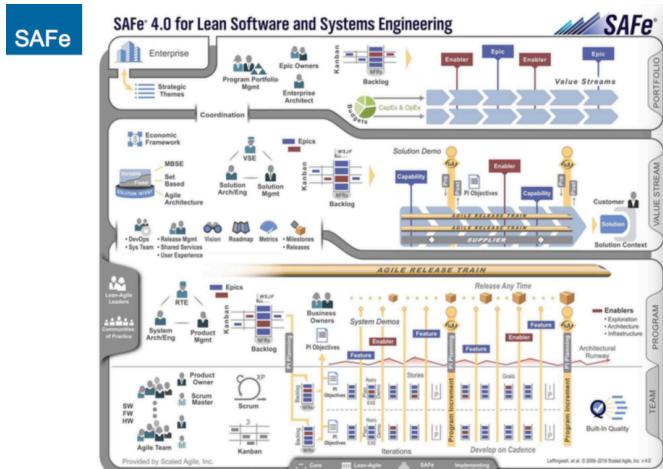
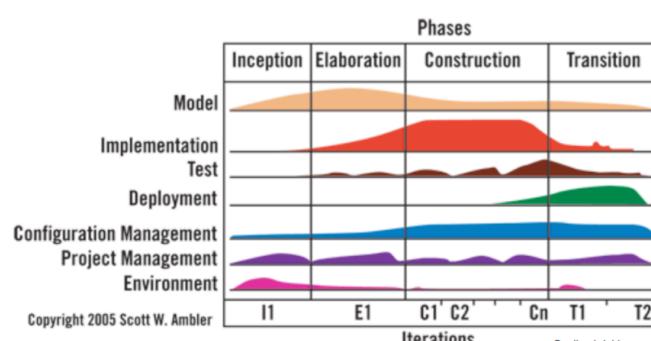
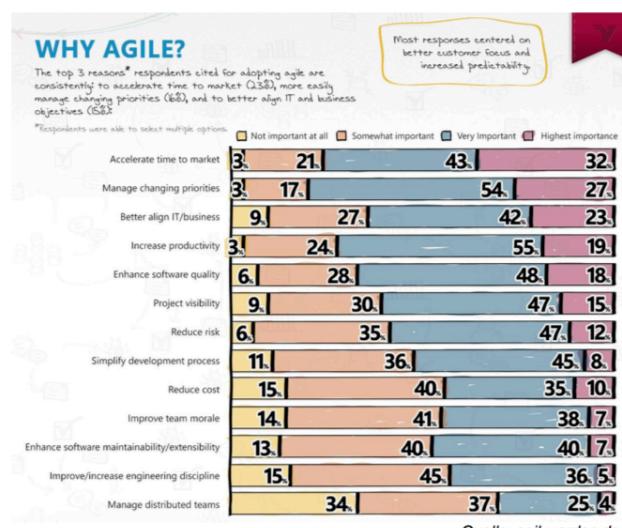
Eine User Story wird im agilen Projektmanagement anstelle von Anforderungen erhoben und in einem Backlog abgelegt. Schema: «Wer» möchte «was» (und «warum»).

Beispiel für Zeiterfassungssoftware

Ich als Projektleiter möchte die Buchungen meiner Mitarbeitenden freigeben können, damit diese Gültigkeit erlangen.

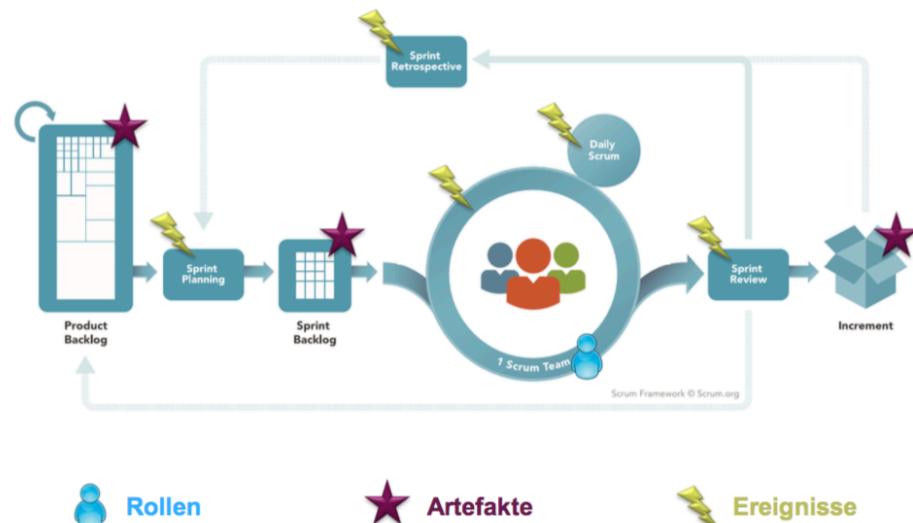
Heute verwendete agile Methoden**Welche agilen Methoden werden in der Schweiz verwendet**

Quelle: swissq.it (2016)

**Agile Unified Process****Warum werden agile Methoden verwendet**

Ziele des Kapitels

Lerninhalte	Ziele: Die Studierenden...
<ul style="list-style-type: none"> - Scrum - Kanban - Weitere agile Methoden 	<ul style="list-style-type: none"> - lernen Scrum noch besser kennen oder repetieren die Methode - verstehen das Vorgehen von Kanban - wissen um weitere Ausbaumöglichkeiten der agilen Methoden

Scrum Framework in der Tiefe**Die Elemente**

Rollen	Artefakte	Ereignisse	Dauer
Product Owner	Product Backlog	Sprint	2- 4 Wochen
Development Team	Sprint Backlog	Sprint Planning	<4 – 8 Stunden*
Scrum Master	Increment	Sprint Review	2 – 4 Stunden*
		Sprint Retrospektive	1.5 – 3 Stunden*
		Daily Scrum	<15 Minuten*

* Streng ausgelegt kein Timeboxing

Fundament von Scrum

Transparenz	Überprüfung	Anpassung
<ul style="list-style-type: none"> - Wesentliche Aspekte sind sichtbar - Gemeinsamer Standard 	<ul style="list-style-type: none"> - Fortschritt ständig überprüfen - Zu gegebenen Zeiten überprüfen 	<ul style="list-style-type: none"> - Anpassung so schnell wie möglich - Anpassung ist stetig möglich

Ereignis	Überprüfung	Anpassung
Sprint Planning	Backlog, Increment	Sprint Ziel
Daily Scrum	Fortschritt	Sprint Backlog, Tagesplanung
Sprint Review	Increment, Sprint, Product Backlog	Product Backlog
Sprint Retrospektive	Verhalten, DoD	Regeln

Unterscheidungen

Sprint Review und Sprint Retrospektive

Das Increment wird überprüft Das Product Backlog darf angepasst werden Stakeholder erhalten Einblick in das Produkt	Sprint Review
Das Team überprüft DoD Das Team ändert DoD Interaktion im Team besprechen	Sprint Retrospektive
Lenkungsausschuss Statusreport zu Handen Auftraggeber (nicht in Scrum vorgesehen) Der Scrum Master zeigt die Performance (Burn Down) Eine Earned Value Analyse wird gemacht	Keines von beiden

Product und Sprint Backlog

Das Product Owner priorisiert Kann so viele Stories wie möglich beinhalten Einträge daraus werden alle 2-4 Wochen besprochen	Product Backlog
Das Development Team priorisiert/nimmt entgegen Enthält nur eine bestimmte Anzahl Stories Das Development Team passt es während Sprint an Einträge daraus werden täglich besprochen Maximal 10 % der Zeit für die Vereinfachung verwenden	Sprint Backlog
Besteht ausschliesslich aus Testfällen Der Scrum Master priorisiert	Keines von beiden

Formuliert die Product Backlog Einträge Nimmt nie am Daily Scrum teil Optimiert das Increment hinsichtlich Geschäftsnutzen	Product Owner
Modifiziert das Sprint Backlog während des Sprints Bestimmt was im Sprint Backlog Platz hat Ist Eigentümer der Elemente im Sprint Backlog Legt Definition of Done (DoD) fest	Development Team
Nimmt am Daily Scrum teil oder stellt Durchführung sicher Beseitigt Hindernisse die Sprints aufhalten Schafft Verständnis für die Produktplanung	Scrum Master

«Verunreinigung» von Scrum

- Scrum kennt keinen Zero-Sprint
- Sprints werden nie verlängert oder verkürzt (Timebox)
- Nur der Product Owner darf Sprint abbrechen
- Burn Down Charts verschiedener Teams nie vergleichen
- Teams werden nicht bestimmt
- Daily Standup ist unbekannt, es heisst Daily Scrum
- Der Product Owner gehört zum Team und ist nicht (nur) Stakeholder
- Arbeit wird nicht verteilt (Push) sondern genommen (Pull)

Semi-Agil mit Kanban

Was ist Kanban (in der IT)?

«Kanban ist ein Vorgehensmodell zur Softwareentwicklung, bei dem die Anzahl paralleler Arbeiten, der Work in Progress (WiP), reduziert und somit schnellere Durchlaufzeiten erreicht und Probleme – insbesondere Engpässe – schnell sichtbar gemacht werden sollen.»

Nutzen



- Arbeit visualisieren
- Ist-Zustand abbilden
- Workflow beobachten
- Beschränkungen einbauen (Fokus auf aktive durch WiP-Limit)
- Gesamtarbeit wird in gut überschaubare Aufgaben unterteilt
(Zu grosse Aufgaben fliessen schlecht durch das System)
- Prognosen ableiten (und reagieren)

Woher kommt Kanban?

Es kommt aus dem japanischen und stellt dort ein System zur Herstellung dar. Bis 1947 hat man nach dem Push Mechanismus produziert. Als einfach mal darauf los. Dies brauchte immer mehr Lagerhallen. 1947 sagte einer bei Toyota, dies sei nicht effizient und ruft den Pull Mechanismus ins Leben. Es wurde nur noch auf Anfrage produziert. Das Lager meldet einen Aufbau bei der Fabrik für Nachfüllungen und das Schiff wiederum beim Lager, wenn es etwas brauchte.

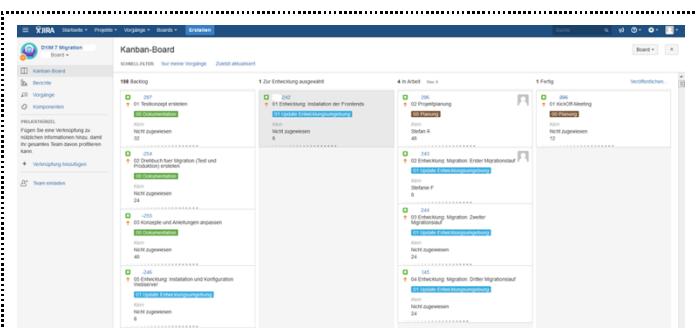
Die genaue Idee lautete wie folgt : «Es müsste doch möglich sein, den Materialfluss in der Produktion nach dem Supermarkt-Prinzip zu organisieren, das heißt, ein Verbraucher entnimmt aus dem Regal eine Ware bestimmter Spezifikation und Menge; die Lücke wird bemerkt und wieder aufgefüllt.»

Folgen des Einsatzes von Kanban bei Toyota

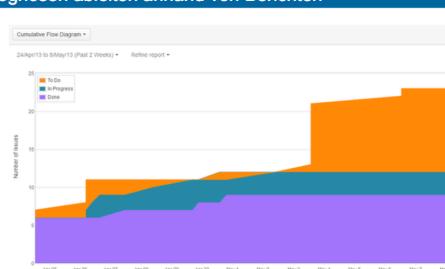
- Innerhalb der Logistikkette erhält jede Arbeitseinheit nur die Materialmenge, die sie effektiv zum Produzieren benötigt.
- So wird sehr flexibel immer genau die richtige Menge in der richtigen Qualität produziert.
- Effizienzsteigerung in der Produktion
- Verschwendungen durch Lagerhaltung wird weitgehend vermieden
- Kürzere Reaktionszeiten auf sich ändernde Rahmenbedingungen
 - o Marktsegmente, Gesetze
- Senkung der Kosten -> Steigerung des Gewinnes

Später (um 2010) wurde dieser Mechanismus auf die Software-Entwicklung angewendet. Folgende Bilder zeigen eine solche Implementierung.

Backlog	Ready for development (2/6)	Work in progress (4/4)	Done
Item 10 Item 11 Item 12 Item 13 Item 14 Item 15 Item 16 Item 17 Item 18 Item 19	Item 9 Item 8	Item 7 Item 6	Item 1 Item 2 Item 3 Item 4 Item 5

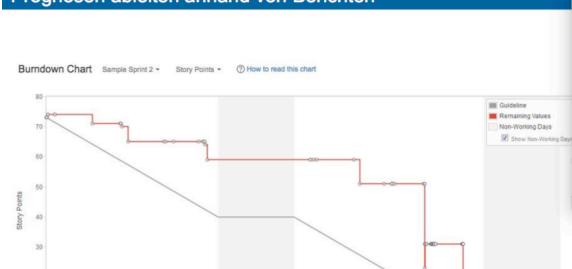


Prognosen ableiten anhand von Berichten



Overview
Drag and drop cursor across chart or chart overview by selected date range (double click overview for reset).

Prognosen ableiten anhand von Berichten



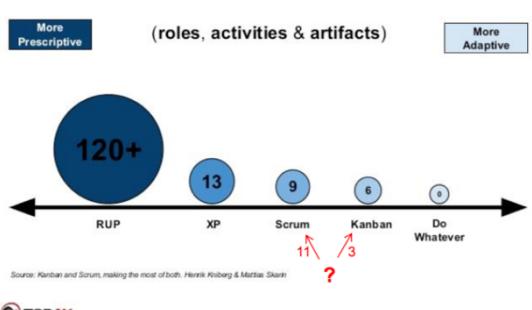
Burndown Chart
Sample Sprint 2 • Story Points • How to read this chart
Guideline Remaining Values Non-Working Days Show Non-Working Day

Kommentar

Kanban gilt offiziell zu Recht nicht als agile Methode. Kanban enthält aber so gute Elemente, dass diese immer wieder in der Praxis sichtbar sind, auch im Zusammenhang mit Scrum.

Weitere agile Methoden

Prescriptive vs. Adaptive



Wenn wir Kanban gegen Agilität prüfen, dann wird Inspect & Adapt sowie Timeboxing nicht erfüllt. Die anderen beiden Aspekte schon.

Die Lösung dafür ist **Scrumban**, zum Beispiel mit Sprints (pro Sprint ein Board) und Sprints Reviews.

Somit ergänzen wir also Kanban um agile Elemente aus Scrum.

Qualitätsmanagement

Ziele des Kapitels

Lerninhalte

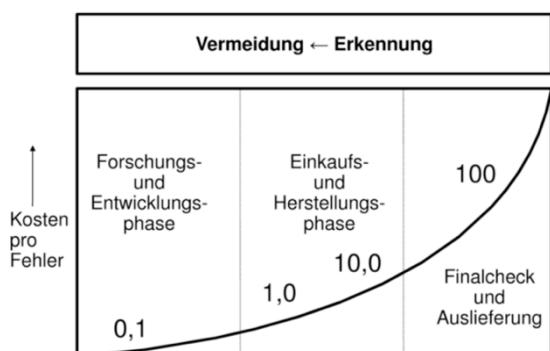
- QS im Projekt
- Change-Management
- QS im Unternehmen mit TQM und SixSigma

Ziele: Die Studierenden...

- kennen Verfahren, um Fehler möglichst gut und früh zu erkennen
- verstehen warum Änderungen im Projekt sauber abgehandelt werden
- erweitern ihr Wissen um zwei QS Standards im Unternehmen

QS im Projekt

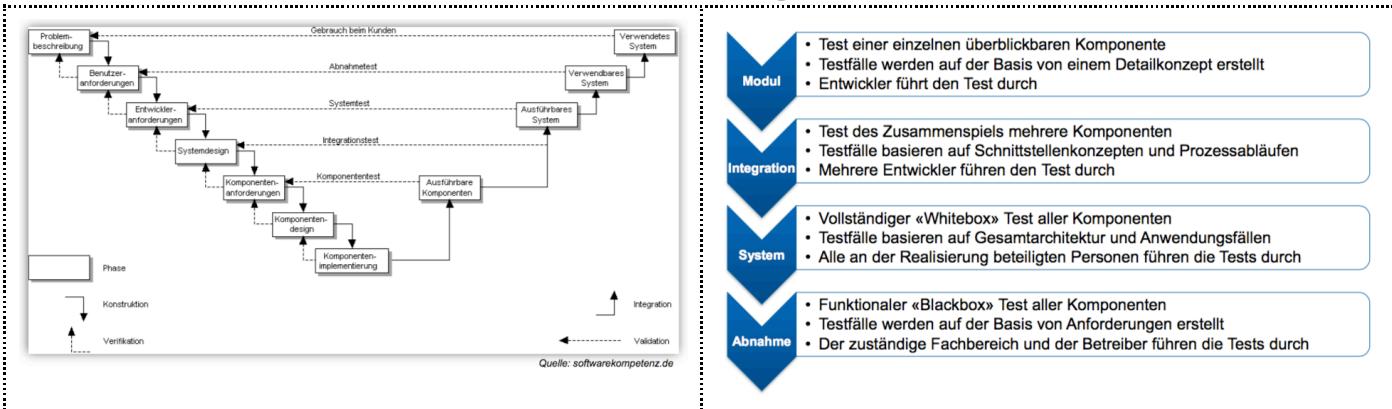
Unklare Anforderungen lassen keine exakte Messung zu, womit auch kein genaues Ergebnis erzielt werden kann.



Qualitätssicherung heißt strukturierte Qualitätsprüfung. In den Projekten können frühzeitige Qualitätsinstrumente sicherstellen, dass die Lieferobjekte den Anforderungen entsprechen. Je früher eine Fehlerbehebung ist, desto günstiger – späte Korrekturen sind um einiges teurer.

Negative Beispiel wäre Toyota, welche über 6.5 Millionen Autos in die Werkstatt zurückrufen musste.

Das V-Modell im Bestreben einer Früherkennung



Verfahren zur Qualitätssicherung in Softwareprojekten

Review Die Lieferergebnisse werden von einer weiteren Person einem Review unterzogen. Dies betrifft Spezifikation sowie auch Code. Nebst der Qualität wird die Wartbarkeit sehr stark verbessert. Dies kann bis zur Extreme praktiziert werden (z.B. Paarprogrammierung).

Continuous Delivery Regelmässige Auslieferung von Software in die Acceptance oder Produktion. Es wird im Zuge von Externe Programming verwendet. Entwickler arbeiten nur in einer Versionsverwaltung, eingecheckter Code wird automatisch kompiliert, getestet und paketiert sowie integriert.

Ad absurdum / Negativtests Die involvierten Personen müssen die Anforderungen und das Ergebnis aus Prinzip bezweifeln und damit ernsthaft hinterfragen. Gegenteilige Anforderungen werden diskutiert bis diese im Widerspruch enden. Es werden Testfälle kreiert, die bewusst einen Fehler auslösen.

Beispiel Ad Absursum

Der Kunde sagt: Im Papeterie Webshop für Geschäftskunden sollen die Produkte einzeln ausgewählt und in den Warenkorb gelegt werden. Nachdem der Kunde auf «Einkauf» drückt, darf nichts mehr am Warenkorb geändert werden. Die Währung wird vor der Bezahlung ausgewählt. Danach wird die Mehrwertsteuer mit einer Multiplikation von 1.08 erhoben. Die Zahlung wird vom Vorgesetzten und dem Kostenstellenverantwortlichen des Kunden nacheinander freigegeben.

Ich behaupte das Gegenteil: Im Papeterie Webshop sollen die Produkte einzeln/**mehrfach** ausgewählt und in den Warenkorb gelegt werden. Nachdem/**bevor** der Kunden auf «Einkauf» drückt, darf nichts/**noch** mehr am Warenkorb geändert werden. Die Währung wird vor/**nach** der Bezahlung ausgewählt. Zum Schluss wird die Mehrwertsteuer mit einer Multiplikation/**Division** von 1.08 erhoben. Die Zahlung wird dem Vorgesetzten und den Kostenstellenverantwortlichen des Kunden nacheinander/**gleichzeitig** freigegeben/**abgelehnt**.

Der Kunde muss Ihnen die gegenteilige Anforderung soweit widerlegen, dass diese wirklich keinen Sinn macht und daher angenommen werden kann, dass die ursprüngliche Anforderung korrekt und gewollt ist. Beim Code-Review muss der Entwickler Ihnen beweisen, dass die gegenteilige Anwendung nicht möglich ist. Beim Schreiben der Testfälle wählen Sie die gegenteiligen Anforderungen, z.B. das Wechseln der Währung von Euro in tschechische Kronen nach Festlegung der Betragshöhe, bewusst aus und definieren eine Fehlermeldung als erwartetes Resultat.

Change-Management im Projekt



Ein Change ist eine Änderung im Projekt, ein Request also dessen Antrag. Beispiel: Die Vorabklärung hat zu viel Geld benötigt, weshalb für Konzeption und Entwicklung nicht mehr der benötigte Betrag zur Verfügung steht. Entweder Scope reduzieren (Weniger Inhalt, damit das Geld reicht) oder Budget beantragen (Zusätzliche Mittel um Projektziel zu erreichen). Folgende Kapitel zeigen die verschiedenen Teile eines Change Request.

Kategorien von Changes Request (I)

Streng genommen sind es genau drei Kategorien (Mehrfachauswahl möglich). **Scope**, **Budget** und **Time**. Weitere Kategorien sind der Praxis anzutreffen:

Legal Terms

Änderung des Vertrages zwischen Lieferant und Kunde, zum Beispiel aufgrund der Verlegung des Projektortes

Ressourcen

Änderung von namentlich genannten Personen, zum Beispiel der Projektleiter als Hauptansprechperson wird ausgetauscht

Projekt- und Qualitätsmanagement PmQm

Voraussetzungen eines Changes (II)

Der Change Prozess muss definiert sein, das heisst Sie müssen wissen wie und an welches Board (z.B. Lenkungsausschuss) Sie den Change richten müssen. Die Änderung muss fassbar und die beteiligten Personen müssen sich dessen bewusst sein. Die technischen und organisatorischen Voraussetzungen müssen erfüllt oder als Voraussetzung definiert sein.

- Change ist machbar
- Ressourcen sind vorhanden
- Termine sind realistisch
- Alle sonstigen «Hausaufgaben», damit der Change durchführbar ist.

Beschreibung eines Changes (III)

Begründen Sie den Change kurz und prägnant zu Handen einer Zielgruppe, die sich im Projekt nur oberflächlich auskennt. Erpressen Sie keinen Entscheiden sondern suggerieren Sie Wahlfreiheit. Erwähnen Sie unbedingt, wenn der Change auf Basis eines früher erkannten Risikos notwendig wird (macht guten Eindruck, weil PL es früh erkannt hat). **Verwenden** sie Wörter wie: *Ermöglichen, ausserordentlich, neue Rahmenbedingungen, mehr Gewissheit, mehr Sicherheit, Nutzen, Vorteile, etc.* und **vermeiden** Sie Wörter wie *muss, darf nicht sein, keine Ahnung, aus heiterem Himmel, keine Chance, verlangen, nur diese Lösung, etc.*

Auswirkungen und Risiken (IV)

Beschreiben Sie die Auswirkung des Change mit all seinen Konsequenzen separat und nicht in der Change Beschreibung. Beschreiben Sie auch die Auswirkungen, wenn der Change abgelehnt wird z.B. *Eine Ablehnung des Changes würde bedeuten, dass der Funktionsumfang begrenzt bleibt und damit bessere Akzeptanz ermöglicht wird.*

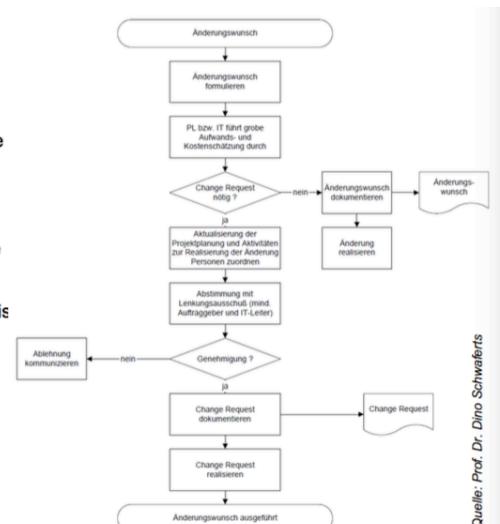
Erwähnen Sie, welche neuen Projektrisiken ein Change mit sich bringt z.B. Durch das neue Modulk des Lieferanten XY aus den USA, erhöhen sich die Währungsrisiken, da wird in USD einkaufen.

Change quantifizieren (V)

In einem Change Request müssen Sie immer Budget und Zeit quantifizieren. Oftmals steht die Ampel des Projekts vor dem Change auf rot, da man z.B. erkannt hat, dass das Budget aufgrund von Zusatzanforderungen nicht mehr ausreicht. Mit dem Change haben Sie die Möglichkeit, mehr Budget und mehr Zeit zu «legalisieren» und damit ihr Projekt wieder auf Status grün zu setzen. Quantifizieren Sie die Changes so hoch wie möglich und so tief wie nötig. Wenn Sie neue Anforderungen aufs Auge gedrückt bekommen, ändern Sie nicht nur das Budget sondern verlangen Sie auch mehr Zeit für die Ausführung

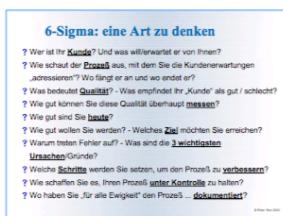
Vorgehen

Gehen Sie nur mit dokumentierten und quantifizierten Anträgen zum Entscheidungsgremium. Nummerieren und dokumentieren Sie die Changes in gleicher Masse, wie einen Projektauftrag oder einen Projektvertrag. Dokumentieren Sie entgegen der nebenstehenden Grafik auch abgelehnte Changes (cover my ass). Im Kunden/Lieferanten Fixpreis-Verhältnis sind Changes eine Änderung des Werkvertrages. Beim Kunden und Lieferanten müssen gemäss Unterschriftenregelung kompetente Personen unterschreiben



Quelle: Prof. Dr. Dino Schwaiger

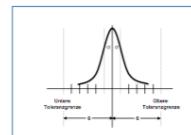
➤ „Denkzeug“



➤ Werkzeug



➤ Kennzahl



Das Ziel ist Six Sigma. Sigma ist ein statistischer Ausdruck, der die Abweichung von der Norm misst; ein Prozess der auf einem Six-Sigma-Niveau arbeitet, wird in 99,99966 % der Fälle fehlerfrei ablaufen»

Vergleich klassisches QS mit SixSigma

Klassisch «99% gut», also (3,8)

- **20000** verlorene Briefe pro Stunde in Europa
- **5000** falsche chirurgische Eingriffe pro Woche in Europa
- **2** Landungen ausserhalb der Rollbahn auf den grössten europäischen Flughäfen täglich

SixSigma «99,99966% gut» (6)

- **7** verlorene Briefe pro Stunde in Europa
- **1,7** falsche chirurgische Eingriffe pro Woche in Europa
- **0,0007** Landungen ausserhalb der Rollbahn auf den grössten europäischen Flughäfen täglich

Was bedeutet Sigma genau

Sigma Zahl	Fehler pro Million	fehlerhaft %	fehlerfrei %
1	691.462	69 %	31 %
2	308.538	31 %	69 %
3	66.807	6,7 %	93,3 %
4	6.210	0,62 %	99,38 %
5	233	0,023 %	99,977 %
6	3,4	0,00034 %	99,99966 %

SixSigma – 5 stetige Verbesserungsschritte



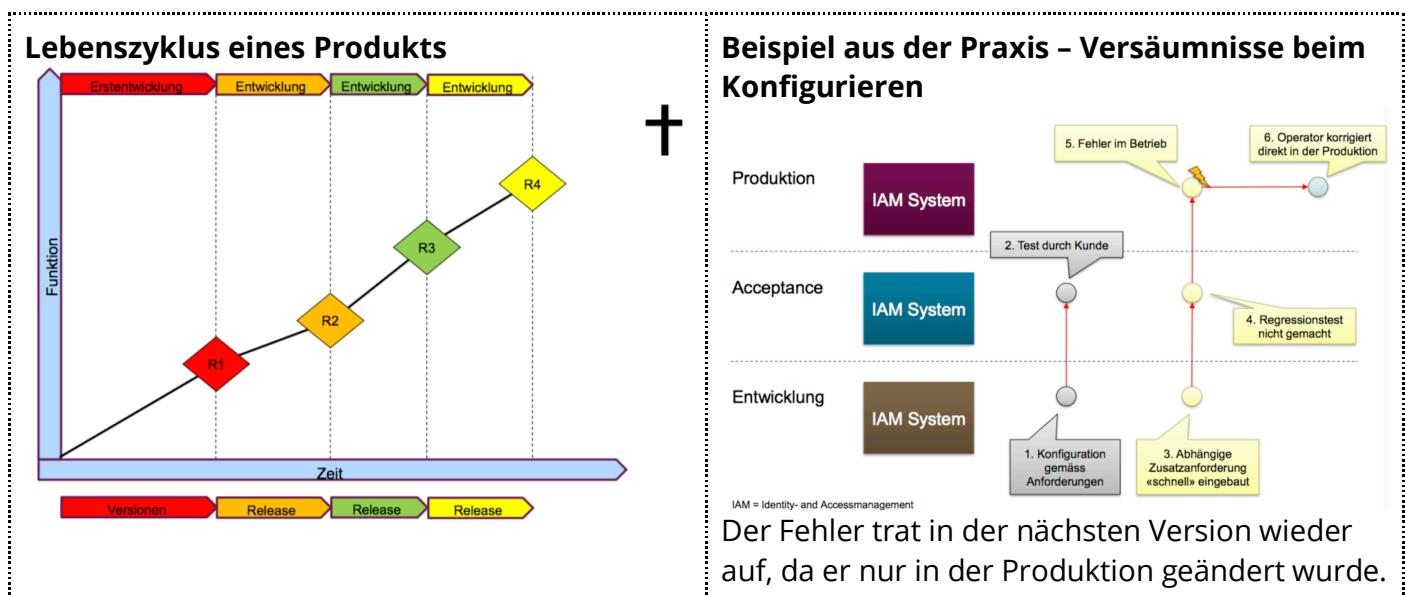
Wozu braucht es Konfigurationsmanagement?

- Disziplin übergreifende Überwachung der Arbeitsschritte in der Entwicklung und in der Produktion
- Produktpflege Sicherstellung der qualitativ richtigen Lieferobjekte
- Qualitativ richtige Auslieferung der Lieferobjekte
- Aufzeichnung der Entwicklungsgeschichte
- Strukturierung des Änderungswesens
- Überwachung der Änderungen und der Änderungsprozesse
- Fehlerkorrekturen und Festlegung der Auswirkungen auf einzelnen Versionen
- Konfigurationsmanagement ist ein wichtiger Pfeiler des Qualitätsmanagement

Typische Fehler in der Software-Entwicklung, welche mit einem sauberen

Konfigurationsmanagement verhindert werden können:

- Software lässt sich sehr leicht ändern, dies verleitet zu nicht dokumentierten und schnellen Reparaturen, teilweise sogar direkt auf dem Produktionssystem
- Artefakte werden stetig geändert und weiterentwickelt, aus Unachtsamkeit könnten alte Artefakte verwendet werden
- Mehrere Personen arbeiten am gleichen Code und überschreiben sich gegenseitig

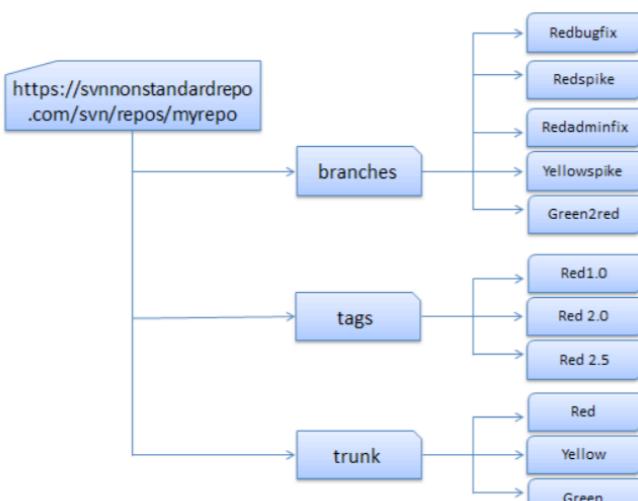


Releases und ihre Versionsbezeichnung

Es lohnt sich, zu Beginn des Projekts festzulegen, wie die Releases und Versionen bezeichnet werden. Es gibt keinen gängigen Standard dazu. Bei www.semver.org gibt es eine Empfehlung für die Softwareentwicklung.

MAJOR.MINOR.PATCH (z.B. 1.2.3)

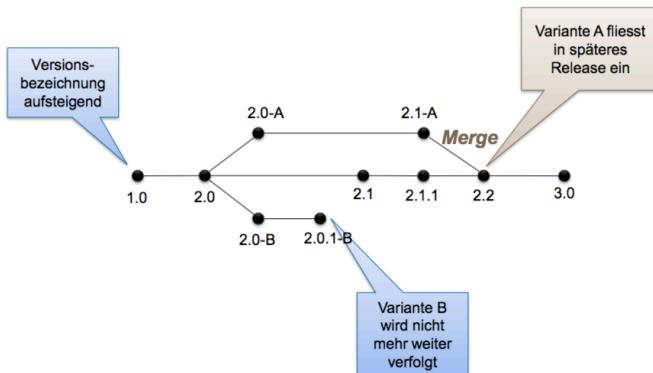
1. MAJOR version when you make incompatible API changes
2. MINOR version when you add functionality in a backwards-compatible manner
3. PATCH version when you make backwards-compatible bug fixes



Der Trunk (Stamm) enthält die aktuelle Entwicklungslinie, hier wird in der Regel ausgecheckt und weitergearbeitet.

Der Branch (Ast) enthält alternative Entwicklungszweige oder Varianten.

Ein Tag enthält ein unveränderlicher (eingefrorener) Stand des Produkts.



Merge Strategien

Speeren «Check-Out»

Die Artefakte werden solange gesperrt, wie sie in Bearbeitung sind. Paralleles Arbeit nicht wirklich möglich. Sicher und immer möglich. Keine Vermischung innerhalb eines Artefakts, nur ausserhalb.

Mischen

Automatischer Merge des Inhaltes. Dezentrale unabhängige Entwicklung möglich. Steckt voller Unsicherheiten, nur möglich wenn Source vorhanden. Vermischung benötigt Absprache zwischen Entwickler.

Manuelles Mischen

Zuständige Stelle mischt manuell und verwendet die «guten» Teile. Bei kompetenter Stelle nur Vorteile ausser Zeitfaktor.

Vorteile eines guten Tools

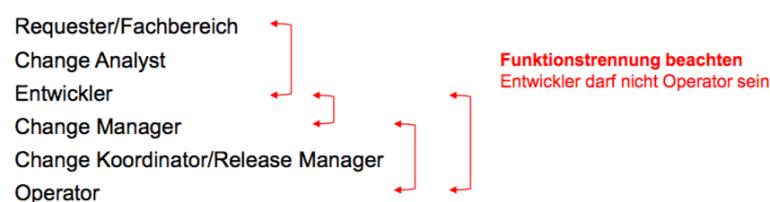
- Verwaltet alle Komponenten an einem Ort, z.B. mit einem virtuellen Dateisystem über mehrere Standorte hinweg
- Änderungen können identifiziert und beurteilt werden
 - o Annehmen
 - o Verwerfen
- Macht Unterschiede zwischen Versionen sichtbar, mittels Compare-Funktion oder automatischem Merging
- Wiederherstellung voriger Zustände
- Mit Tags können besondere Versionen, die einen speziellen Zwischenstand darstellen, definiert werden
- Mittels Workflow können Change- und Release-Prozesse elektronisch abgebildet werden

ITIL

ITIL gibt als Best Practise Framework einen Prozess vor, welcher von Konfigurationsmanagement flankiert wird. Das Service Level Management gibt die Anforderungen, diese werden umgesetzt und durch das Change Management freigegeben, so dass diese vom Release Management eingeplant werden. Das Change Management unter ITIL hat mit dem Change Management aus der Projektwelt nur am Rande zu tun. ITIL beschreibt vor allem Änderungen (auch noch so kleine) im Betrieb und im Projektmanagement sind Changes grössere Abweichungen von Budget, Scope, etc.

Berechtigungen in SCM

Wenn Sie Konfigurationen und Prozesse in einem Tool verwalten, ist es notwendig die Berechtigungen korrekt zu setzen. Aus Sicht Funktionsbreite modellieren Sie pro Prozessteilnehmer am besten eine Rolle, zum Beispiel:



Aus Sicht Datenbreite unterteilen Sie wenn notwendig (Need-to-Know) noch die verschiedenen Produkte oder Projekte in unterschiedliche Datenräume und hinterlegen Sie jeweils einen Owner. So könnte eine Berechtigungsmatrix aussehen:

	Change stellen	Change beurteilen	Code bearbeiten	Code kompilieren	Change bewilligen	Daten lesen
Requester	X					X
Change Analyst		X				X
Entwickler			X	X		X
Change Manager					X	X
Change Koordinator	X	X				X
Operator	X					X

Wenn Sie ein Multi-Role System haben, also wenn Personen mehrere Rollen zugewiesen werden dürfen, müssen Sie die Funktionstrennung zusätzlich beachten und dies möglichst ausschliessen.

Kommunikation und Dokumentation

Ziele des Kapitels

Lerninhalte	Ziele: Die Studierenden...
<ul style="list-style-type: none"> - Stakeholder - Führung / Ressourcenmanagement - Kommunikation - Projektmarketing - Anschlussdokumentation und laufende Dokumentation 	<ul style="list-style-type: none"> - wissen um die Bedeutung der Stakeholder und lernen deren systematische Bearbeitung - kennen die Führungsansprüche in einem Projekt - erkennen die Wichtigkeit von Kommunikation und Projektmarketing - erweitern ihr Wissen hinsichtlich Projektdokumentation und kennen mehrere Beispiele

Stakeholder

Als Stakeholder gelten alle interessierten Parteien am Projekt. Die Stakeholder bilden verschiedene Anspruchsgruppe und beeinflussen das Projekt. Projektleiter tun gut daran, die verschiedenen Stakeholder von Beginn weg zu identifizieren und die Kontakte zu pflegen. Eine stetige Aktualisierung der Stakeholder Informationen ist unerlässlich. Typische Stakeholder sind die Mitglieder des Lenkungsausschusses, die Kunden, die Lieferanten, die verschiedenen Endbenutzer, die Leistungserbringer aus IT oder Facilitymanagement und generell alle Personen die durch irgendeine Art vom Projekt betroffen sind. Beim sogenannten Stakeholder Management gibt es unterschiedliche Möglichkeiten diese zu visualisieren. Diese werden in diesem Unterkapitel noch genau erläutert.

Sechs Fragen um Stakeholder zu finden

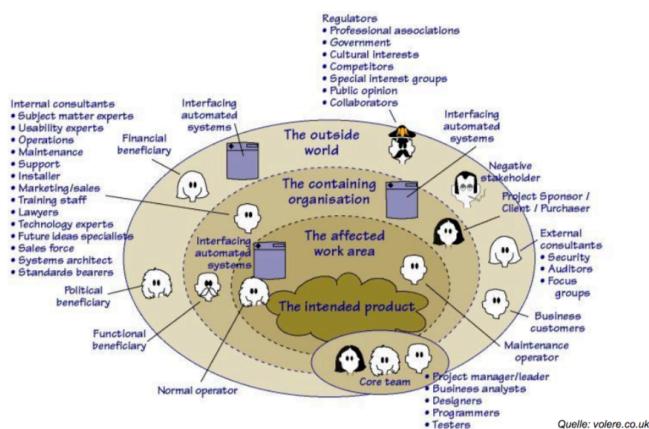
- Wer arbeitet am Projekt mit oder stellt Material zur Verfügung?
- Wer kann Einfluss nehmen auf das Projekt?
- Wer ist vom Projekt besonders betroffen?
- Wer kann den Projekterfolg beeinflussen?
- Wer ist Nutzniesser des Projekts?
- Wer muss zwingend involviert werden?

Gehört das Projektteam zu den Stakeholdern?

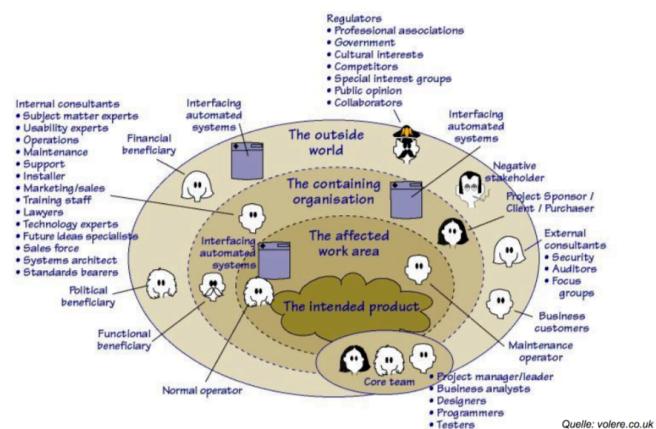
In der Fachliteratur existiert keine einheitliche Meinung, ob das Projektteam selber zu den Stakeholdern gehört. Je nach Betrachtungsweise gehört das Team daher zu den Stakeholdern oder auch nicht.

- **PRO:** Die sechs Fragen passen auf die Mitglieder des Projektteams
- **CONTRA:** Stakeholder sind Anspruchsgruppen die es zu identifizieren und analysieren gibt, für das Projektteam ist dies nicht notwendig.

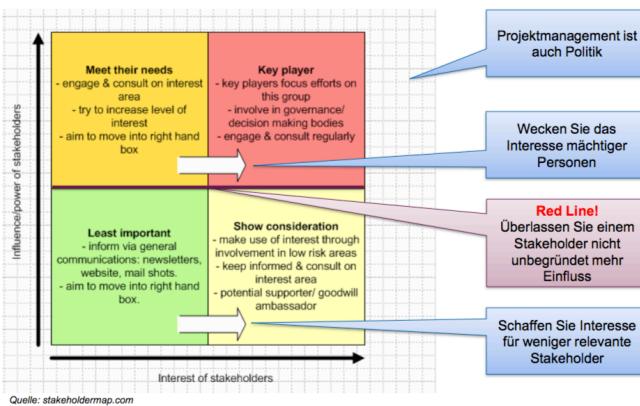
Wer ist alles Stakeholder eines Projekts?



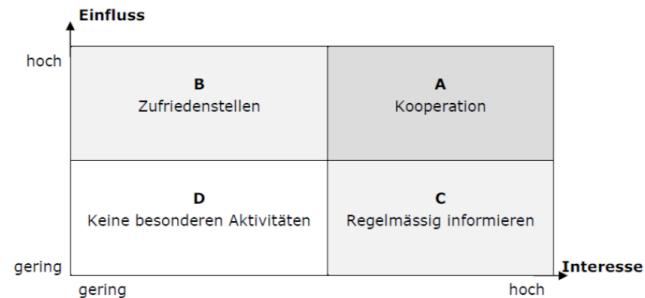
Nach der Identifikation kommt die Analyse



Abwägen von Einfluss und Interessen



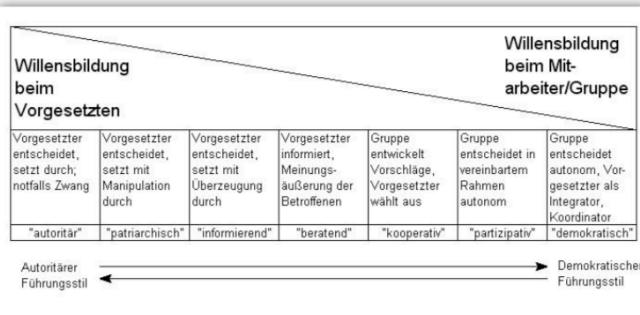
Klassische Einordnung nach IPMA



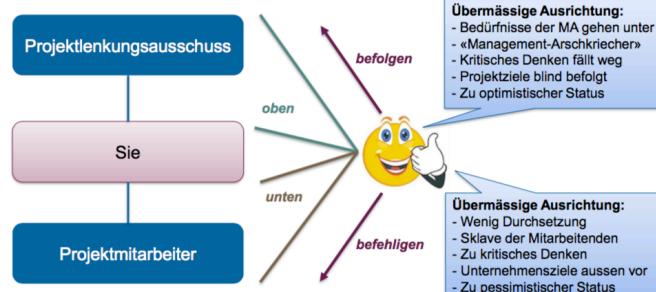
Führung / Ressourcenmanagement

«Unter Führung (Leadership) versteht man die Anleitung und Motivation anderer bei der Erfüllung ihrer Funktionen oder Aufgaben im Dienst der Projektziele. Es handelt sich um eine für Projektmanager ausschlaggebende Kompetenz» NCB 4.1. Sobald mehr als eine Person am gleichen Auftrag oder gleichen Produkt arbeiten, braucht es Führung oder mindestens Koordination. Das «Ein-Mann-Vorhaben» benötigt Arbeitstechnik und Disziplin. Projektführung ist die Anleitung der Arbeitstechnik für eine Gruppe. Heutige IT-Projekte sind keine «Ein-Mann-Vorhaben» mehr und benötigen daher zwingend Führung. Wer selber eine gute Arbeitstechnik hat, beweist Methodenkompetenz und besitzt eine wichtige Eigenschaft eines guten Projektleiters.

Führungsstile nach Tannenbaum



Führungs-(Nahrungs-)kette – Gleichgewicht der Interessen



Der Führungsrythmus beschreibt die Häufigkeit mit welcher eine Führungskraft Informationen aufnimmt, beurteilt, entscheidet und Resultate kontrolliert.



Gestalten Sie den Rhythmus so selten wie möglich, aber so häufig wie nötig. Ein hoher Rhythmus benötigt viel Zeit, gibt aber viel Einfluss. Organisieren Sie strukturiert und strategisch, also z.B. wöchentliche Projektsitzungen, monatliche Einzelgespräche, etc. Erhöhen Sie den Rhythmus bei anstehenden Herausforderungen.

Ressourcenmanagement

Das Ressourcenmanagement im Projekt bezeichnet die Kapazitäts- und teilweise auch die Einsatzplanung von Personen und Systemen. In den Stammesorganisationen müssen Sie frühzeitig die Ressourcenanfragen tätigen. Berücksichtigen Sie das Wissen der entsprechenden Mitarbeitenden, lassen Sie sich nicht mit dem «B-Kader» abspeisen. Planen Sie so grob wie möglich, aber so detailliert wie nötig. Führen Sie Änderungen stetig nach.

Kommunikation

Kommunikation haben einen sehr grossen Zusammenhang mit dem Stakeholder Management und der Führung. Daher haben gute Projekte ein Kommunikationskonzept. In einem Kommunikationskonzept legen Sie fest, wer im Projekt was, wie, wann und warum kommuniziert.

Projekt- und Organisationsanalyse

Es gilt, die Projektorganisation zu analysieren und deren Aufbau darzustellen. Hierzu ist das Organigramm des Projekts notwendig, um die folgenden Fragen zu beantworten:

- Wer ist für die Gesamtkommunikation innerhalb des Projektes verantwortlich?
- Welchen Einfluss habe ich als Projektleiter auf die Kommunikation ausserhalb des Projektes?
- Wie stellen sich die Kommunikationswegen zwischen den einzelnen Einheiten dar?
 - o Wer sind die Entscheider für ein Projekt?
 - o Welches sind die Ansprechpersonen innerhalb der Projektorganisation?

Zudem sollen die wichtigen Termine und Meilensteine aufgelistet werden, die für die Kommunikation im Projekt relevant sind.

Beispiel einer Komm-Matrix aus der Praxis							Sitzungslandkarte					
Stakeholder (Kategorie)/ Adressat	Instrument(e)	Ziel / Zweck	Inhalt	Zeitpunkt / Periodizität	Verantwortlich	Bemerkungen	Stakeholder (Kategorie) / Anlass	Ziel / Zweck	Zeitpunkt / Periodizität	Teilnehmer	Verantwortlich	Bemerkungen
Stakeholder	Welches Instrument wird eingesetzt?	Welche Ziele/Absichten werden mit der Maßnahme verfolgt?	Welche Informationen sind für den jeweiligen Stakeholder relevant?	Zeitpunkt und Periodizität der Kommunikationsmaßnahme	Wer ist verantwortlich für die Kommunikation?	Allfällige Bemerkungen (z.B. Eigenschaften des Stakeholders, besondere Bedingungen etc.)	Sitzung	Ziele und Zweck des Meetings	Wann und wie oft findet das Meeting statt?	Wer nimmt am Meeting teil?	Wer ist verantwortlich?	Kurzbeschreibung des Meeting (z.B. Traktanden, die regelmäßig behandelt werden müssen).
Bsp: Leiter HR	Bsp: E-Mail, Persönliches Gespräch, Projektnewsletter	Bsp: Mit regelmässigen Informationen versorgen, um Stakeholder als Befürworter zu halten.	Bsp: Informationen zum Projektfortschritt, Nutzen und Konsequenzen für HR.	Bsp: Projektnewsleiter alle zwei Wochen, Persönliches Gespräch je nach Bedürfnis.	Bsp: PL	Bsp: Stakeholder bevorzugt E-Mail-Kommunikation da schwierige Erreichbarkeit.	Bsp: Teamsitzung Projektteam	Bsp: Update Projektstatus, Austausch allfälliger Schwierigkeiten, Abgleich Informationsstand	Bsp: Die Teamsitzung findet jeweils jede zweite Woche am Montag statt.	Bsp: PL, MA, MA Kunde	Bsp: PL	Bsp: Allgemeine Informationen zum Projektstatus, Schwierigkeiten, Personelles, Meilensteine etc.

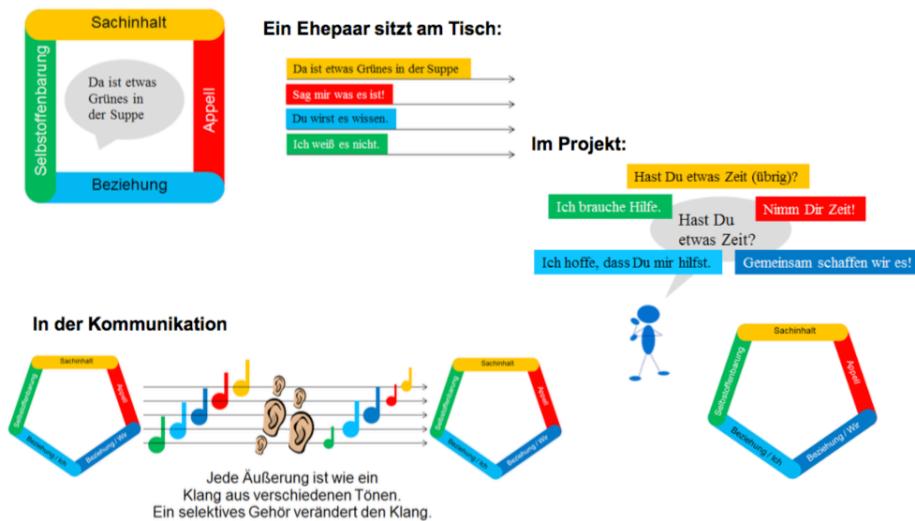
Gute Kommunikation

- Jeder weiss fundiert was er benötigt und ein wenig darüber hinaus
- Sitzungen sind eine Nebensache aber notwendig
- Die Stakeholder fühlen sich «abgeholt»

Schlechte Kommunikation

- Alle wissen nichts oder jeder weiss alles
- Viele Sitzungen behindern das Arbeiten
- Relevante Stakeholder sind nicht informiert

Kommunikation nach Schulz von Thun



Projektmarketing

Marketing bedeutet zum Markt tragen, es ist weit mehr als reine «Werbung», wobei es im Zusammenhang mit Projekten auf die «Werbemassnahmen» reduziert wird.



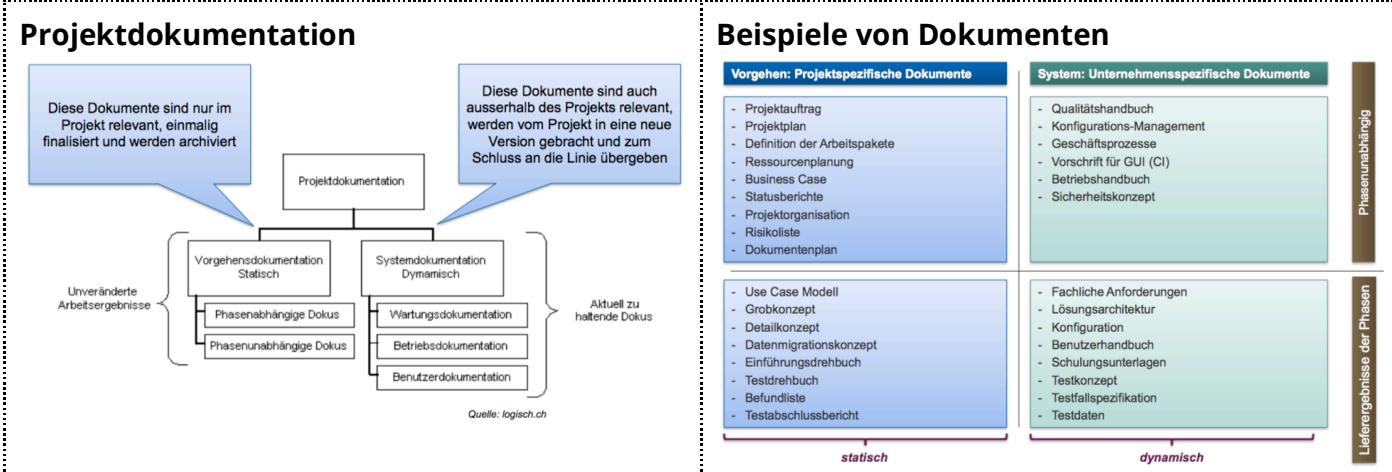
Beispiele von Marketing-Aktionen

- Roadshows bei den Endbenutzer
- Film über das Projekt
- Infos an der Sitzung der Führungskräfte
- Tischset in der Kantine des Kunden
- Meldungen im Intranet des Kunden
- Bericht in der Kundenzeitung
- Erstellung einer Projektbroschüre
- Master-Slides in PowerPoint über das Projekt
- Events

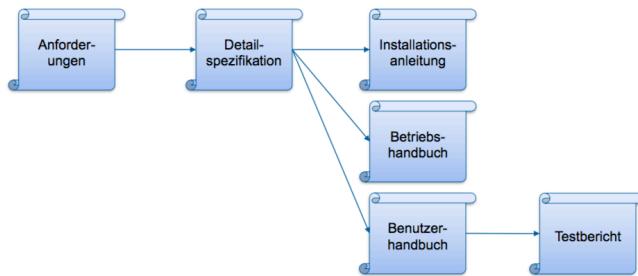
Der Projektleiter als Botschafter

Die Präsentationsstärke eines Projektleiters ist einer der massgeblichsten Faktoren im Projekt. Zwischenergebnisse müssen souverän präsentiert und Fragen kompetent beantwortet werden. Die Präsentation ist die wichtigste Visitenkarte des Projektleiters, viele Führungskräfte achten darauf und sich «Bauchentscheider». Mit Präsentationen bauen Sie als Projektleiter Beziehungen zu den Stakeholdern auf. Der Projektleiter vertritt hauptsächlich das Projekt gegen aussen und somit gegenüber den Kunden, Gelbgebern sowie anderen Stakeholdern. Bei wichtigen Ereignissen wird erwartet, dass der Projektleiter persönlich teilnimmt.

Projekte wie Insieme haben gezeigt, dass Dokumentation etwa arg wichtiges ist.



Am besten erstellt man zu Beginn des Projekts einen Dokumentenplan, welcher zeigt, welche Dokumente sie wem schuldig sind.



Lösungsfindung und Teamarbeit

Ziele des Kapitels

Lerninhalte	Ziele: Die Studierenden...
<ul style="list-style-type: none"> - Verschiedene Methoden zur Lösungsfindung - Reviewtechniken - Sitzungstechniken - Verhandlungstechnik 	<ul style="list-style-type: none"> - kennen mehrere Methoden zur Lösungsfindung - verstehen den Ablauf eines Reviews - wenden die Sitzungsplanung an - erweitern ihr Wissen hinsichtlich Verhandlungstechnik

Verschiedene Methoden zur Lösungsfindung

Das methodische Vorgehen zur Findung von Lösungen mag auf den ersten Blick als übertrieben scheinen (wie mit Kanonen auf Spatzen schiessen). Es gibt aber durchaus Situationen, wo eine geordnete Findung der einzige Ausweg ist oder die Bewertung nachvollziehbar sein muss. Hier ein Auszug einer Methoden, welche nun noch genauer betrachtet werden.

Finden	Bewerten
<ul style="list-style-type: none"> - Brainstorming - Methode 635 - Morphologischer Kasten 	<ul style="list-style-type: none"> - Nutzwertanalyse - Entscheidungsbaum

Finden | Brainstorming

Die **Zusammensetzung** der Gruppe sollte eine möglichst breite Streuung des Fachwissens haben. Innerhalb der Gruppe dürfen keine zu grossen hierarchischen Unterschiede vorhanden sein (sonst Angst etwas zu sagen). **Teilnahme** daran **freiwillig** oder wenigstens nicht **widerwillig**. Das Problem sollte bereits **vorher bekanntgegeben** werden. Besonders bei komplexen Problemen etwa zwei Tage vor der Sitzung. Der Sitzungsraum **soll vorbereitet** werden, dass eine **Isolierung** von **äusseren Einflüssen** vorhanden ist. Hilfsmittel wie Handtafel und Flipchart gelten vorbereitet. Eventuell Verpflegung zur Schaffung einer zwanglosen Atmosphäre. Protokollführer für schriftliche Fixierung der geäussernten Ideen.

Regeln

Für den Diskussionsleiter

- Er formuliert das Problem nochmals möglichst exakt (paraphrasieren)
- Er überwacht den Ablauf der Sitzung
- Er greift ein, wenn die Regeln nicht beachtet werden oder der Ideenfluss zu versiegen droht

Für den Protokollführer

- Er schreibt alle Ideen auf
- Er strukturiert die Ideen und fertigt ein Protokoll für die Entscheidungsinstanz an

Für alle anderen Teilnehmer

- Kritik ist verboten
- Der Phantasie soll freien Lauf gelassen werden, um möglichst kühne Ideen zu erhalten
- Je mehr Ideen, desto besser: Quantität geht vor Qualität
- Die Ideen anderer sollen aufgenommen, kombiniert und weiterentwickelt werden
- Gelegenheit geben innert 24h nochmals darüber nachzudenken

Vorgehen

Jedes Mitglied der aus genau **6 Teilnehmern** bestehenden Gruppe schreibt **3 Ideen** auf ein Blatt Papier, das **5 mal weitergereicht** wird. Jeder Teilnehmer ergänzt das Blatt um 3 weitere Ideen und lehnt sich möglichst an das vorhandene an. Folgende Zeitvorgaben pro Runde haben sich als geeignet erwiesen:

- 1. Runde | 5 min
- 2. Runde | 6 min
- 3. Runde | 7 min
- 4. Runde | 8 min
- 5. Runde | 9 min
- 6. Runde | 10 min

Finden | Morphologischer Kasten

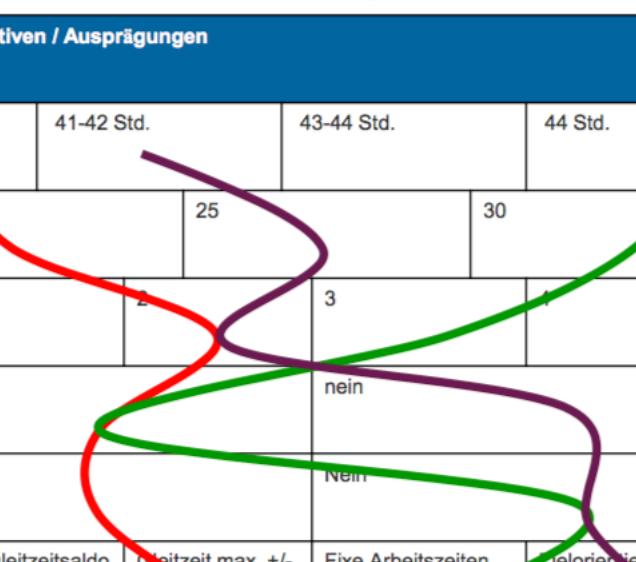
Die Morphologie wurde vom Schweizer Forscher F. Zwicky entwickelt (er spricht dabei von «diskursiver Problemlösung»). Mit dieser Methode soll durch das systematische Zusammenstellen aller logischen denkbaren Möglichkeiten über einen bestimmten Bereich in Form einer Tabelle das unvollständige, eingefahrene, fixierte Denken vermieden werden.

Hauptinstrument der Morphologie ist ein «morphologischer Kasten». Dabei wird in vier Schritten vorgegangen.

1. Problem als Überschrift des Kastens notieren
2. Zerlegung in Teilprobleme oder Handlungsfelder
3. Lösungsvarianten pro Teilproblem oder Handlungsfeld ermitteln
4. Kombination herausfinden, zum Beispiel jede Person kriegt einen andersfarbigen Stift.

Beispiel

		Problem: Festsetzung der Arbeitszeit			
Teilprobleme / Parameter		Alternativen / Ausprägungen			
Gesamtarbeitszeit pro Woche		40 Std.	41-42 Std.	43-44 Std.	44 Std.
Anzahl Ferientage pro Jahr		20	25	30	
Zahl der Schichten		1	2	3	
Bezahlte Pausen		ja		nein	
Kaffee & Gipfeli & Früchte zur Pause		ja		Nein	
Gleitzeitmodell		Jahresgleitzeitsaldo	Gleitzeit max. +/- 20h	Fixe Arbeitszeiten	Zielorientierung ohne Gleitzeit



Bewerten | Nutzwertanalyse

Die Nutzwertanalyse weist den gewichteten Nutzen pro Variante aus. Als Bewertung findet man in der Schweiz oft das Schulnotensystem (1 = sehr schlecht, 6 = sehr gut). Folgende vier Schritte werden benötigt:

1. Kriterien (oder Anforderungen) auflisten
2. Kriterien gewichten, wobei Gesamtgewicht = 1 (oder 100 %) sein muss
3. Jedes Kriterium für jede Variante bewerten
4. Punktzahl zusammenzählen

Bei knappen Unterscheidungen empfiehlt es sich die Sensitivität der Gewichtung mittels kleinen Verschiebungen zu überprüfen.

Kriterien auflisten

Noten setzen

Gesamtgewicht beachten

Exceldatei

Nutzwertanalyse für Zeiteinsparungssoftware

Kriterien	Anbieter 1		Anbieter 2		Anbieter 3		
	Gewicht	Note	Punkte	Note	Punkte	Note	Punkte
Bewertung Hersteller	10%	4	0.4	6	0.6	5	0.5
Funktionalität	25%	3	0.75	6	1.5	5	1.25
Look & Feel	15%	6	0.9	5	0.75	5	0.75
Wartbarkeit	20%	2	0.4	6	1.2	5	1
Initialkosten	10%	6	0.6	2	0.2	5	0.5
Betriebskosten	20%	6	1.2	2	0.4	5	1
	100%		4.25		4.65		5

Gesamtbewertung ablesen

Nr.	Eigenschaft	Gewichtungsfaktor	Ziel(Punkte)	(Unter-)Punkte	Srum	Nutzwert	Ziel(Punkte)	Kanban	Nutzwert
1	Plumung und Kontrolle	15%	10	8	1,2	4	0,6		
1.1	Artefakte			4	0,6	2	0,3		
1.2	Prozess			4	0,6	2	0,3		
2	Organisationsstruktur	5%	10	8	0,4	5	0,23		
3	Projektoptimierung	20%	10	7	1,4	8	1,6		
3.1	Organisation			5	1	4	0,8		
3.2	Geschwindigkeit			4	0,4	4	0,8		
4	Umgang mit komplexen Projekten	15%	10	9	1,35	6	0,9		
4.1	Projektkennung			4	0,6	1	0,15		
4.2	Projektsonderheiten			2	0,15	2	0,3		
4.3	Teamgröße und Teamorganisation			4	0,6	3	0,45		
5	Dauer der Einführung	5%	10	4	0,2	9	0,45		
6	Vorständlichkeit/Einfachheit der Entwicklungsprozessmethode	5%	10	2	0,1	10	0,5		
7	Produktivität/Effizienz der Entwicklungsprozesse	35%	10	9	0,45	7	0,35		
	Summe	100%	70	47	5,1	49	4,65		

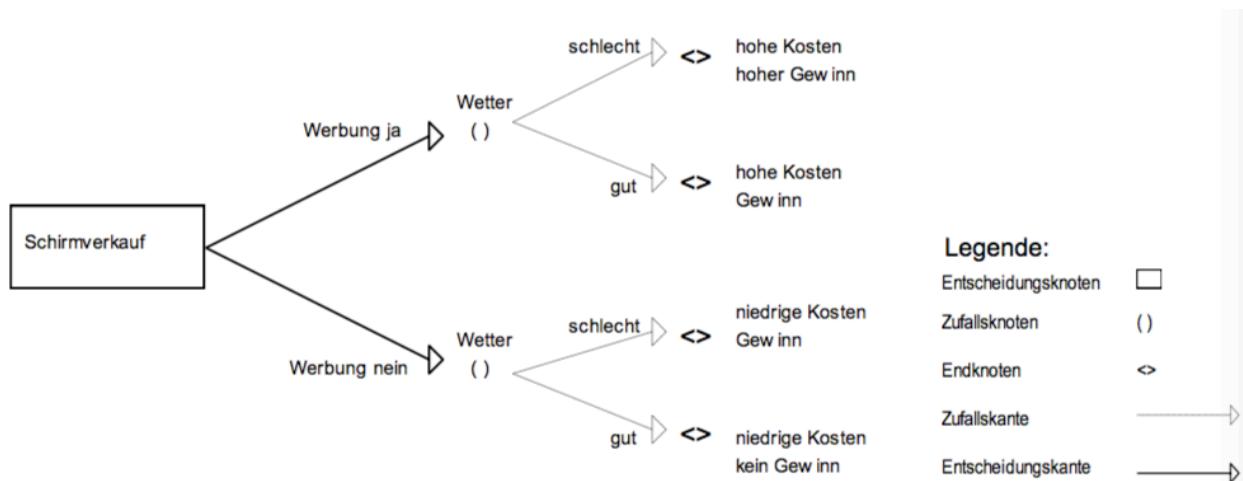
Quelle: wi-form.de

Bewerten | Entscheidungsbaum

Im Entscheidungsbaum wird ein Problem systematisch dargestellt. Die möglichen Entscheidungen werden mit dem zufälligen Ereignis verknüpft und die möglichen Endzustände mit den entsprechenden Konsequenzen dargestellt. Nimmt man weiter an, dass die zufälligen Ereignisse mit bestimmten Wahrscheinlichkeiten eintreten und ordnet man einzelnen Aktionen und Knoten gewisse Vergleichsgrößen (Nutzen, Aufwand) zu, kann man Entscheidungsbäume für technische Problemlösungen nutzen.

Beispiel eines Entscheidungsbaums

Ein Schrimgeschäft überlegt sich, ob es eine Werbeaktion starten soll. Dabei wird das Wetter (unbeeinflussbares Ereignis) mit in die Überlegung einbezogen. Die Entscheidung lässt sich wie folgt darstellen (ohne Angaben von Wahrscheinlichkeiten).



Reviewtechnik

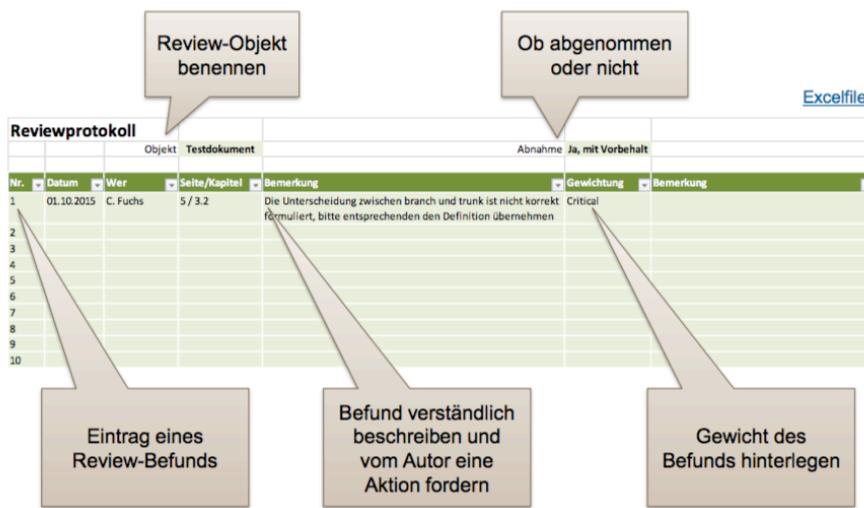
Ein Review ist ein systematischer Check durch einen Stakeholder oder durch unabhängige Personen.

Bespiele von Review-Objekten

- Dokumente
- Code
- Konzepte
- System Architektur
- Software Architektur

Ein Review kann als Sitzung oder schriftlich als Protokoll durchgeführt werden. Man darf auch loben und muss nicht nur bemängeln. Das Object wird nach dem Review abgenommen, unter Vorbehalt abgenommen oder zurückgewiesen.

Beispiel eines Reviewprotokolls



The screenshot shows an Excel spreadsheet titled "Reviewprotokoll". The columns are labeled "Objekt", "Testdokument", "Abnahme", and "Ja, mit Vorbehalt". The "Bemerkung" column contains a note: "Die Unterscheidung zwischen branch und trunk ist nicht korrekt formuliert, bitte entsprechenden den Definition übernehmen". The "Gewichtung" column is labeled "Critical". The table has 10 rows, with row 1 being the header.

Nr.	Datum	Wer	Seite/Kapitel	Bemerkung	Gewichtung	Bemerkung
1	01.10.2015	C. Fuchs	5 / 3.2	Die Unterscheidung zwischen branch und trunk ist nicht korrekt formuliert, bitte entsprechenden den Definition übernehmen	Critical	
2						
3						
4						
5						
6						
7						
8						
9						
10						

Annotations in the diagram:

- "Review-Objekt benennen" points to the first column of the table.
- "Ob abgenommen oder nicht" points to the "Abnahme" column.
- "Excelfile" points to the file extension ".xlsx" at the top right of the table.
- "Eintrag eines Review-Befunds" points to the first row of the table.
- "Befund verständlich beschreiben und vom Autor eine Aktion fordern" points to the "Bemerkung" column.
- "Gewicht des Befunds hinterlegen" points to the "Gewichtung" column.

Sitzungstechnik

Der Sitzungsleiter bereitet die Sitzung vor und eröffnet diese, in Projekten ist dies in der Regel der Projektleiter. Folgende Fragen können Sie sich als Vorbereitung stellen:

- Ist die Notwendigkeit der Sitzung gegeben?
- Sind die Ziele bekannt?
- Welche Teilnehmer sind notwendig?
- Haben die Teilnehmer auch Zeit für die Sitzung?
- Was muss am Ende für jeden heraus kommen?
- Wie sind die Abhängigkeiten zum Umfeld?
- Welche Traktanden und in welcher Reihenfolge werden besprochen?

Frühzeitige Festlegung des Termins und eine persönliche Einladung verstehen mit Traktandenliste und Zielen empfiehlt sich sehr.

So agieren Sie als Sitzungsleiter

- Protokollscreiber bestimmen
- Wo notwendig, Protokoll der letzten Sitzung verabschieden
- Einleiten und Traktanden mit den Zielen nochmals präsentieren
- Zeitrahmen einhalten
- Sitzung moderieren und nicht bestimmen
- Bei Abschweifung zurück zum Thema finden
- Aufmerksame Beobachtung der Teilnehmer, jeder Teilnehmer muss zu Wort kommen
- Störungen unterbinden
- Festhalten der Entscheide, Termine und Tätigkeiten
- Nie persönlich werden

Die Teilnehmer einer Sitzung

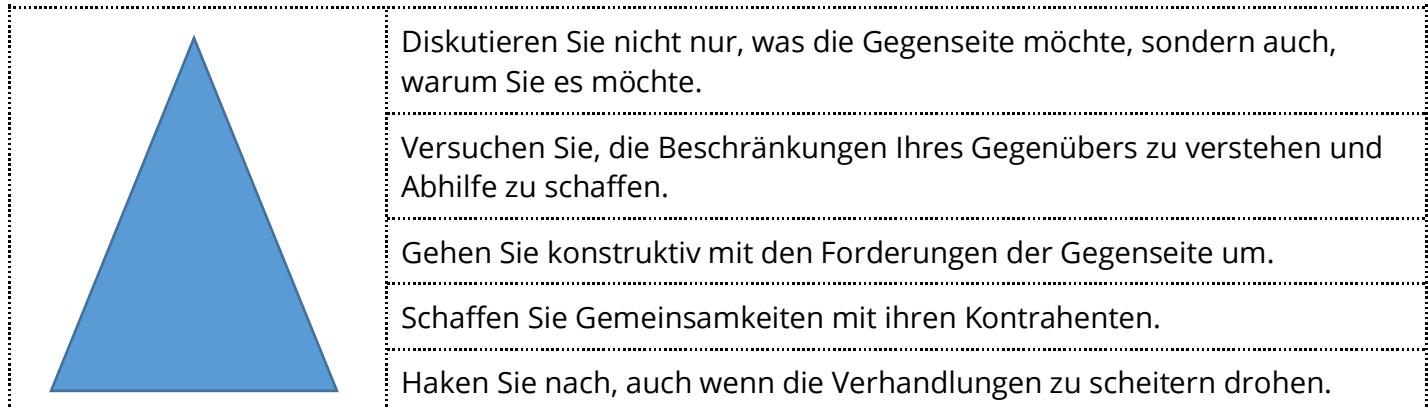
 <p>Bulldoge</p> <ul style="list-style-type: none"> • Widerspricht aggressiv • Kritisiert destruktiv 	 <p>Reh</p> <ul style="list-style-type: none"> • Hält sich zurück • Schweigt am liebsten
 <p>Pferd</p> <ul style="list-style-type: none"> • Sanftmütig und selbstsicher • Zielorientiert 	 <p>Igel</p> <ul style="list-style-type: none"> • Opponiert möglichst immer • Will sich nicht integrieren
 <p>Affe</p> <ul style="list-style-type: none"> • Weiss alles • Unterbricht mit Einwänden 	 <p>Flusspferd</p> <ul style="list-style-type: none"> • Uninteressiert, wortkarg • Unterbricht mit Einwänden
 <p>Frosch</p> <ul style="list-style-type: none"> • Redet und redet und redet • Redet um des Reden willens 	 <p>Giraffe</p> <ul style="list-style-type: none"> • Eingebildet • Nicht kritikfähig

Führungsrythmus

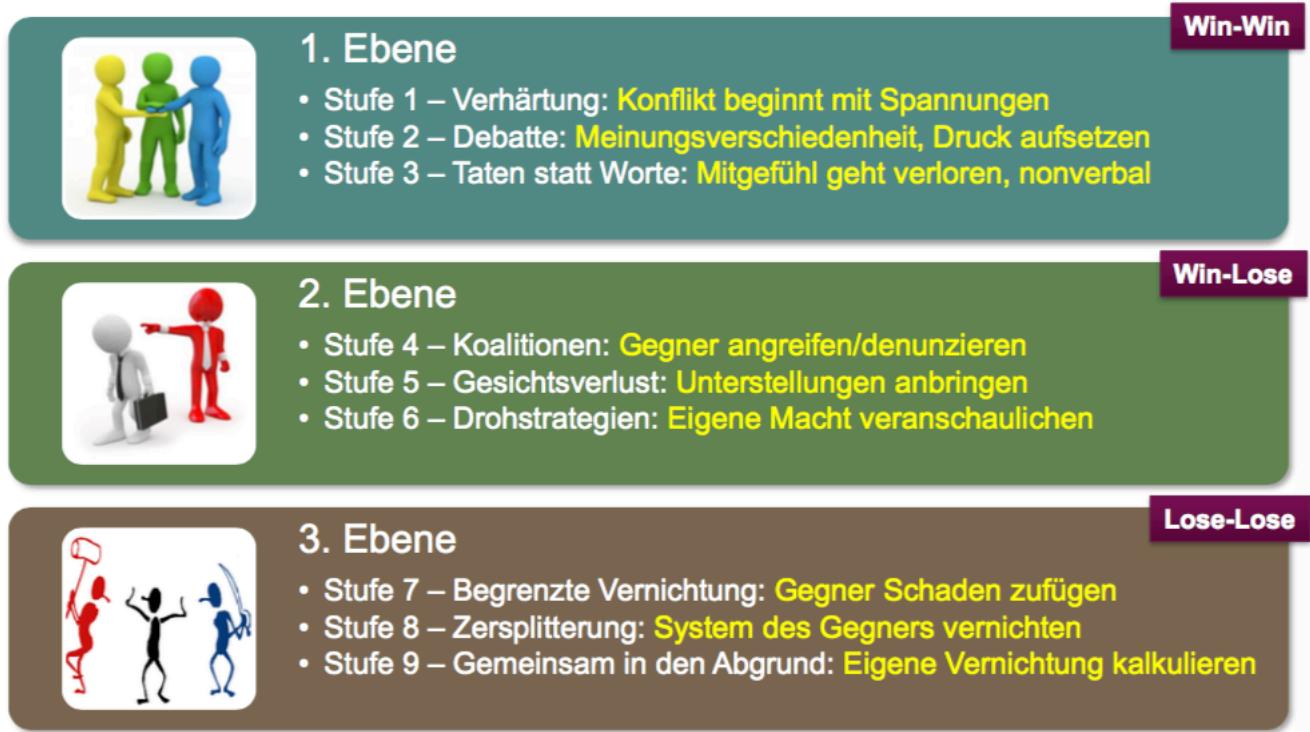
Projektteams werden in der Regeln mit Sitzungen geführt. Zum Beispiel ist es in Scrum das Daily Meeting. Sie als Projektleiter definieren die «**Sitzungslandkarte**» (siehe vorherige Kapitel) im Projekt und legen damit einen passenden Rhythmus fest. Ändern Sie diesen bei Bedarf wie ein Dirigent in einem Orchester.

Verhandlungskommunikation

- Sorgen Sie schon im Vorfeld um eine gute Beziehungsebene
- Verhandeln Sie möglichst auf der Sachebene
- Vermeiden Sie Selbstoffenbarungen
 - o Definieren Sie die gewünschten Ziele vorgängig und legen Sie die Mindestziele fest, welche Sie aber verdeckt halten.
- Verzichten Sie auf Appelle

Die Harvard-Methode | 5 Grundsätze**Kompromiss vs. Konsens**

Suchen Sie den Konsens. Je stärker die Win-Win Situation, desto stabiler wird die Abmachung sein.

Konflikt-Eskalation nach Friedrich Glasl

Projekte in der Praxis

Ziele des Kapitels

Lerninhalte

- Projektziele/-abgrenzung
- Projektantrag/-auftrag
- Projektstruktur (Objekt-,Aktivität, Phasenorientierung)
- Gantt-/Netzplan
- Phasen / Sprints
- Meilensteine
- Ressourcenplanung / AKV
- Kickoff Veranstaltung
- Typische Lieferergebnisse
- Einführung, Big Bang

Ziele: Die Studierenden...

- vertiefen mit Beispielen aus der Praxis ihre Kenntnisse

Projektziele/-abgrenzung

Beispiel Privatwirtschaft

Ausgangslage

Das heute bestehende Identity & Access Management (IAM) erfolgt bei rund 40 Applikationen weitgehend manuell, es ist fehleranfällig und risikobehaftet. Kontrollprozesse für das IAM können nur schwer bewirtschaftet werden. Es soll der Projektsetup für die Umsetzung der Ziellösung spezifiziert werden.

Projektziele

- Employee-Self-Service (ESS) anstelle papierbasiertem Benutzerberechtigungsformular
- Genehmigungsworkflow (Verkürzung der Prozesslaufzeiten)
- Unterstützung von Compliance- und Auditprozessen
- Role Based Access Control (RBAC)
- Automatische Provisionierung
- Single Sign-On (SSO)

Beispiel Bundesauschreibung

Mit dem Projekt Identitätsverbund Schweiz (IDV), welches durch das SECO geführt wird, soll eine Vermittlungsinfrastruktur als umfassenden Identitätsdienst für die Schweiz aufgebaut werden.

Für das Projekt (Konzept, Realisierung, Einführung) sowie gegebenenfalls für den Support, Wartung Weiterentwicklung des funktionsfähigen Systems benötigt das SECO externe Unterstützung. Gleichermaßen gilt für die projektspezifische Begleitung des Projektes.

Vor diesem Hintergrund sollen mit der vorliegenden Ausschreibung zwei fachlich versierte, bestens ausgewiesene externe Leistungserbringer gefunden werden, welche Lieferergebnisse als „Los 1 Entwicklungspartner“ sowie „Los 2 Projektleitung“ für das Projekt IDV-Schweiz anforderungsgemäß liefern können. Über die optionale Einlösung von Support, Wartung bzw. Pflege, Weiterentwicklung und Betrieb wird während des Projektverlaufs entschieden.

5.1 Abgrenzung

Nicht Bestandteil dieser Ausschreibung ist die Realisierungseinheit 3, der Betrieb des Systems. Der Partner hierzu wird im Projektverlauf durch separate Evaluationen ermittelt (vgl. vorstehende Erläuterungen).

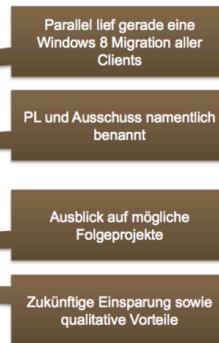
[Ausschreibung](#)

Quelle: simap.ch

Projektantrag/-auftrag

Projektantrag einer Unternehmung mit Prince2

- Ausgangslage
- Abhängige Projekte
- Projektinhalt
- Projektorganisation
- Projektmeilensteine
- Projektbudget
- Roadmap
- Nutzen
- Risikoanalyse
- Laufweg



Woran merkt man, dass es ein Prince2 Projekt ist?

Projektantrag mit Stakeholder besprechen



Beispiel 1 | Service Management

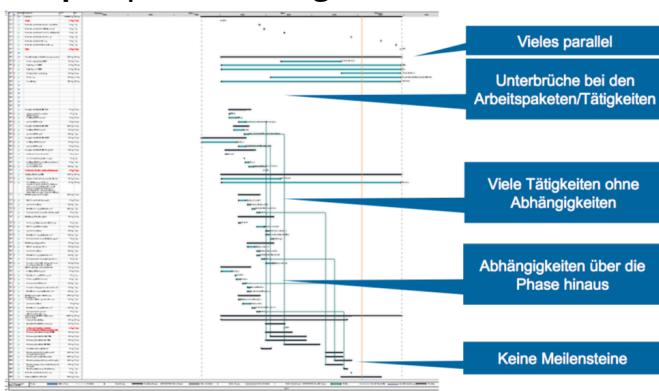
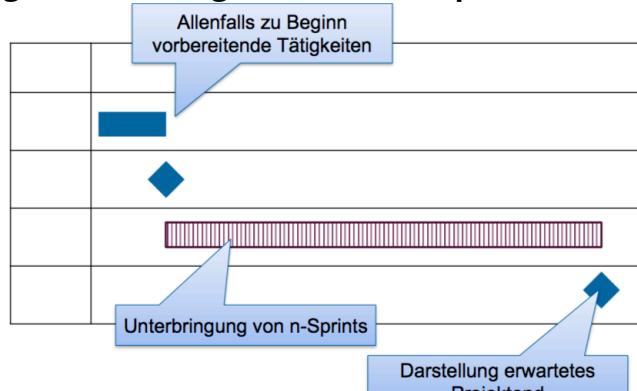
1	Definition.....	2
1.1	Arbeitspaket 201 – Detailkonzept.....	2
1.2	Arbeitspaket 301 – Testumgebung aufsetzen.....	2
1.3	Arbeitspaket 302 – Installation OTRS.....	3
1.4	Arbeitspaket 303 – Konfiguration OTRS.....	3
1.5	Arbeitspaket 304 – Testing OTRS.....	4
1.6	Arbeitspaket 401 – Projektsetup	5
1.7	Arbeitspaket 402 – UI Grundansicht erstellen.....	5
1.8	Arbeitspaket 403 – Benutzerverwaltung umsetzen.....	6
1.9	Arbeitspaket 404 – Webservice Anbindung implementieren.....	6
1.10	Arbeitspaket 405 – Use Case Implementation	7
1.11	Arbeitspaket 406 – Testing Webapplikation.....	7
1.12	Arbeitspaket 501 – Deployment OTRS in Produktion.....	8
1.13	Arbeitspaket 502 – Integrationstest Gesamtsystem	9
1.14	Arbeitspaket 601 – Einführung	9
1.15	Arbeitspaket 602 – Abnahme Gesamtprojekt	10

Beispiel 3 | Erweiterung IAM System

Nr.	Start-Datum	End-Datum	Beschreibung	Verantwortlich IPG	Aufwand IPG
Access-Prozesse					
001	01.09.2014	26.09.2014	Access-Prozesse finalisieren und abnehmen	Pevestorf / Lamparth	80h / 10 PT
002	23.09.2014	02.10.2014	Abgleich der Prozesse	Lamparth	32h / 4 PT
002a	13.10.2014	31.10.2014	Anpassung der Access-Prozesse im Annov IDM	Annov	94h / 12PT
003	23.09.2104	17.10.2014	Erstellung Testfälle und Testszenarien	offen	80h / 10 PT
004	03.11.2014	17.11.2014	Testdurchführung	Annov	80h / 10 PT
005	10.11.2014	28.11.2014	Coaching und Schulung	-	-
Rollenmodell und Rollenprozesse					
006	29.09.2014	10.10.2014	Definition und Beschreibung Rollenmodell	Pevestorf / Lamparth	40h / 5PT
007	06.10.2014	24.10.2014	Rollenprozesse und Rollenmanagement	Pevestorf / Lamparth	90h / 11PT
008	2015	2015	Implementierung Rollenprozesse und	-	-

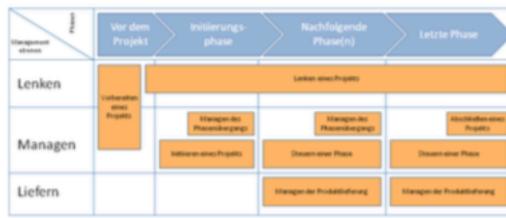
Beispiel 2 | Einführung IAM System

1.	Detailkonzeption	3
2.	Einrichtung des Systems	3
3.	Konfiguration des Systems	4
4.	Customizing	4
5.	Anbindung KISIM	4
6.	Anbindung SAP HR als Quellsystem	5
7.	Anbindung AD	5
8.	Anbindung Exchange	6
9.	Anbindung KABA exos	6
10.	Anbindung e-Archiv	6
11.	Test und Abnahme	7
12.	Vorbereitung Go-Live	7
13.	Deployment in Produktion	7
14.	Go-Live, Produktivsetzung	8
15.	Abnahme der Produktion	8
16.	Detailkonzeption 2. Welle	8
17.	Anbindung SAP	9
18.	Anbindung IBI-Care	9
19.	Anbindung Medora	9
20.	Anbindung weitere Zielsysteme	10
21.	Schulung inkl. Unterlagen in Deutsch	10
22.	Technische Prozesse IAM	11
23.	Erweiterung IKS Kontrollziele	11

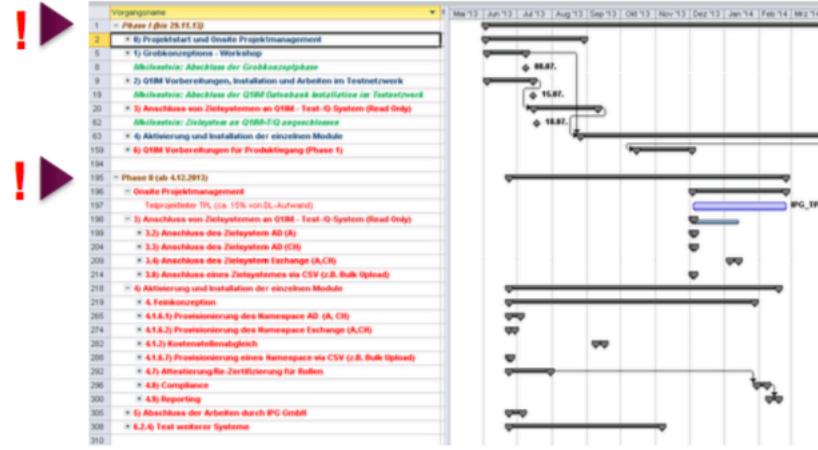
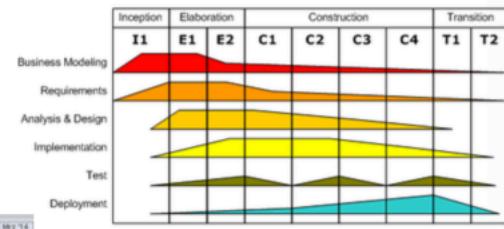
Gantt-/Netzplan**Beispiel | Betrachtung «von Weitem»****Agile Umsetzung in einem Ganttplan**

Phasen / Sprints

Phasen – ein oft missbrauchter Begriff

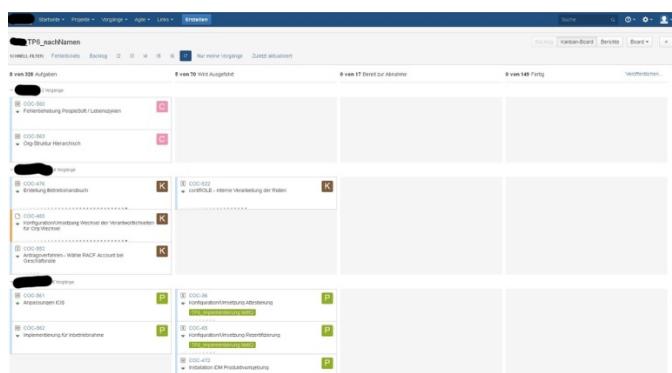


Iterative Development
Business value is delivered incrementally in time-boxed cross-discipline iterations.



Wasserfall | Häufige Fehler bei der Planung der Phasen

Zu vielen parallelen Phasen unterwandern den Grundgedanken einer seriellen Abarbeitung und machen die Vorteile hinsichtlich Transparenz zu Nichte (z.B. RUP nicht sehr vorteilhaft, HERMES viel strikter). Die Phasen werden oftmals mit Ausbaustufen verwechselt. Phasen weisen keine Meilensteine auf. Arbeitspakete und Tätigkeiten werden phasenübergreifend definiert, wodurch eine Phase nicht wirklich abgeschlossen werden kann.



Die Bilder zeigen eine nicht ideale Situation. In einer Phase, welche vorüber ist, dürfen keine offenen Arbeitspakete oder Tasks mehr vorhanden sein.

Meilensteine

Meilensteine können zum Start, beim Ende oder mitten in einer Phase sein, wobei beim Abschluss einer Phase zwingend ein Meilenstein gesetzt werden sollte. Im agilen Projektmanagement ist der Einsatz von Meilensteinen weniger verbreitet. Meilen werden in der Regel eindeutig nummeriert und mit einem Partizip 2 beschrieben (z.B. «MS 1 : Schnittstellenspezifikation abgeschlossen» während dessen Tätigkeiten aktiv mit einem Verb in der Grundform beschrieben werden. Meilensteine können in Werkverträgen als Zahlungszeitpunkte verwendet werden. Ähnlich wie beim Haubau.

Beispiel | Zahlungsplan nach Meilensteinen

Bau eines Hauses im Werte von CHF 400'000

- Zahlungsplan

Der Zahlungsplan geht aus der nachfolgenden Abbildung hervor. Wir stellen fest: Bei der Übergabe des Werks an den Bauherrn ist mit der Zahlung Nummer 5 bereits der ganze Werkpreis fällig (zahlbar innerst 10 Tagen). Nach diesem Stichtag gibt es keinen Rückbehalt mehr auf dem Werkpreis: Der Besteller hat den ganzen Werklohn bereits bezahlt.

Zahlung		Betrag	Kommentar
1	10%	40 000	bei Vertragsunterzeichnung
2	10%	40 000	Baubewilligung liegt vor
3	45%	180 000	Rohbau 1 fertig
4	15%	60 000	Ausbau 1 fertig
5	20%	80 000	bei Übergabe
total	100%	400 000	

Quelle: roeplanner.ch

Meilensteine und deren Abnahme

Meilensteine müssen unter Umständen noch genauer definiert werden (Definition of Done) und in einem Werkvertrag muss insbesondere die Abnahme als Meilenstein ausgewiesen werden, z.B.:



Was passiert wenn der Besteller nicht abnehmen möchte?

- Im Gesetz steht (Art. 370 OR): «Wird das abgelieferte Werk vom Besteller ausdrücklich oder stillschweigend genehmigt, so ist der Unternehmer von seiner Haftpflicht befreit, soweit es sich nicht um Mängel handelt, die bei der Abnahme und ordnungsmässigen Prüfung nicht erkennbar waren oder vom Unternehmer absichtlich verschwiegen wurden.»
- Als Lieferant ist es daher wichtig, eine Frist bei der Abnahme zu setzen, z.B. 10 Tage

Ressourcenplanung / AKV

Reservieren Sie die notwendigen Personen bei deren Stammorganisationen. Legen Sie Ziele pro Zeiteinheit oder gerade die Einsatztage fest. Geben Sie sich etwas Spielraum, falls es zu Verzögerungen kommt. Streben Sie danach, die Mitarbeitenden im Projekt möglichst zu 100% einzusetzen und vor Ort zu haben.



Beispiel | IPG AKV (muss ausgeglichen sein)

3 Projektleiter (IPG PL)

Aufgabe	Kompetenz	Verantwortung
<ul style="list-style-type: none"> Leitet die Projekte aus Sicht IPG Plant die anstehenden Arbeiten zusammen mit den Kunden optimal ein Kontrolliert proaktiv den Fortschritt Meldet die gewünschten Parameter an IPG PD Escaliert ans IPG PD wenn das Projekt außer Kontrolle gerät Speist Changes ins IPG CAB ein Beherrscht die IPG Projektmethodik 	<ul style="list-style-type: none"> Darf mit dem Kunden Details innerhalb Qualität, Budget und Zeitplan aushandeln und verabschieden Hat die IPG Ressourcen ohne Lücken und ohne Überlast einzuplanen Ist für das korrekte Controlling verantwortlich Ist für die Budget-gerechten Reisen und Unterkünfte verantwortlich Kennt die vertraglichen Aspekte seiner Projekte im Detail 	<p>Hohe Auslastung angestrebt</p> <p>Muss die Zahlen kennen</p> <p>Setzt den Vertrag um und muss diesen daher kennen</p> <p>Braucht eine gewisse Weisungsbefugnis</p>

Beispiel | Ausschreibung des Bundes

ZK1.02 Kompetenz in Projektmanagement

Das angebotene Profil des Projektleiters kann mit einem Zertifikat seine Kompetenz im Projektmanagement nachweisen.

Legen Sie eine Kopie des Zertifikats in Kapitel 3 des Angebots ab.

Die Punktevergabe zu diesem Zuschlagskriterium wird wie folgt vorgenommen:
- 100% der Punkte werden vergeben, wenn das Zertifikat IPMA Level B oder PMI/PMP nachgewiesen wird.
- 50% der Punkte werden vergeben, wenn das Zertifikat IPMA Level C oder ein gleichwertiges Zertifikat nachgewiesen wird.
- 0% der Punkte werden vergeben, wenn keine dieser

Bund hat hohe Ansprüche an einen Projektleiter für das IDV Projekt und versucht mit messbaren Kriterien die Person auszuwählen.

Beispiele

Kickoff mit Auftraggeber

- 1 Ausgangssituation / Zielsetzung
- 2 Umsetzungsprojekt
- 3 Organisatorisches

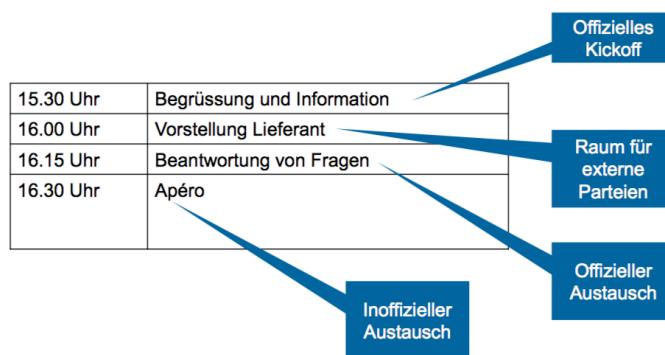
Kickoff Grossprojekt mit Lieferant

- Zielsetzung des Projekts
- Projektorganisation
- Status heute
- Rückblick
- IPG AG – The experts in IAM
- Lösungskonzept
- Kommunikation
- Planung und weiteres Vorgehen

Kickoff mit Teilprojekt

- Zielsetzung des Projekts
- Projektplan
- Angestrebtes Rollenkonzept
- Objekte
- Anwendungsfälle
- Attribute
- Systemarchitektur
- Lösungskonzept
- Planung und weiteres Vorgehen

Zeitplanung typisches Kickoff



Wortschatz Kickoff

Do

- ...ermöglichen...
- ...gemeinsam erreichen...
- Wir stehen vor Herausforderungen
- Es ist nicht einfach, aber wir schaffen das
- Basierend auf euren Schätzungen
- ...gönnen wir uns einen Abschluss

Don't

- ...leisten...
- ...ich führe...
- Wir stehen vor Problemen
- Es ist keine grosse Sache
- Ich habe festgelegt
- Wenn wir es nicht erreichen, wird sanktioniert...

Typische Lieferergebnisse

Es ist absolut wesentlich, in was für einem Verhältnis Sie zum Auftraggeber stehen.

- a) Interner Auftrag in einem Unternehmen → Flexible Vereinbarung was an Software, Papier, etc. geliefert werden muss
- b) Einfacher Auftrag → Offiziell schulden Sie gar nichts und dürfen auch nichts versprechen, um nicht den Charakter eines Werkes zu haben
- c) Werkvertrag → Definieren Sie die geschuldeten Lieferergebnisse und achten Sie darauf, dass diese auch zu erfüllen sind
- d) Kaufvertrag → Wenn Sie ein Projekt ab Stange verkaufen (z.B. eine App), definieren Sie die genauen Kosten und den Umfang der Lieferung.

Beschreiben Sie die Ergebnisse so genau wie nötig und wann diese als erfolgreich gelten.

Was Kunden besonders mögen

- Regelmässiger Statusreport in immer gleicher Form
- Installationshandbuch sofern On-Premise
- Betriebshandbuch für die entwickelte Software
- Attraktive Benutzeranleitung, z.B. als Wiki
- Schulungskonzept zu einem früheren Zeitpunkt
- Migrationskonzept
- Drehbuch für Einführung

Was Sie unbedingt für sich als Projektleiter benötigen

- Change-Log oder Sprint-Vereinbarungen
- Abnahmeprotokoll bei Werkvertrag
- Befundliste für Tests

Beispiel | Bundesprojekt zum Thema «Abnahme»

6.2 Bedingungen der Abnahme

Die abnahmerelevanten Ergebnisse (wie namentlich HERMES-Dokumente und der entwickelte Prototyp) mit Ergebnisverantwortung, welche gemäss Kapitel 5, unterliegen einem Prüf- und Abnahmeprozess.

Ergänzend wird im Besonderen auf die hierfür von den Anbieter zu akzeptierenden Eignungskriterien bezüglich der vertraglichen Regelung (AGB und Vertragsentwürfe) sowie die dazugehörigen Entwürfe für den Rahmenvertrag verwiesen.

→ Bund baut stark auf HERMES, dass heisst Sie als Lieferant müssen die nach HERMES geforderten Dokumente unter Umständen weitgehend alleine schreiben und zusammen mit dem Prototypen abnehmen lassen.

Einführung, Big Bang

Beispiel | Betrachtung «von Weitem»



Beispiel | Drehbuch

Jede notwendige Aktion beschreiben und nummerieren
Wenn möglich genau eine verantwortliche Person definieren
Start und Ende eintragen
Erledigung hart tracken
Umgebung führen
Allfällige weitere Notizen wie Verweis auf Anleitung oder Software-Paket

Unterteilung in Teileinführungen

