



Requirements Management / Change Management

bei agilen Projekten

Martin Landvoigt

[Agenda]

- Requirements Basics: Wer ist der Feind?
- 8 Thesen: Verträge und Verfahren
- ROI Awareness: IFM Crash Course

Requirements Basics

Wer ist der Feind?

„Software-Entwicklung ist **zu teuer**“ ?

Sind Anforderungen bestimmt von FUD

– Fear, Uncertainty & Doubt?

Was ist mit den vielen gescheiterten Projekten ?

Anwendungsstau → Was hindert die Umsetzung?

Beobachtung:

Das größte **Kostenloch** sind nicht die **Tagessätze der Experten**, sondern der **ineffiziente Einsatz** durch **mangelndes Anforderungsmanagement** ...

Gefragt sind Wege zu mehr **Effizienz** und **Business Value**.

Warum überhaupt IT-Projekte?

Ziel:

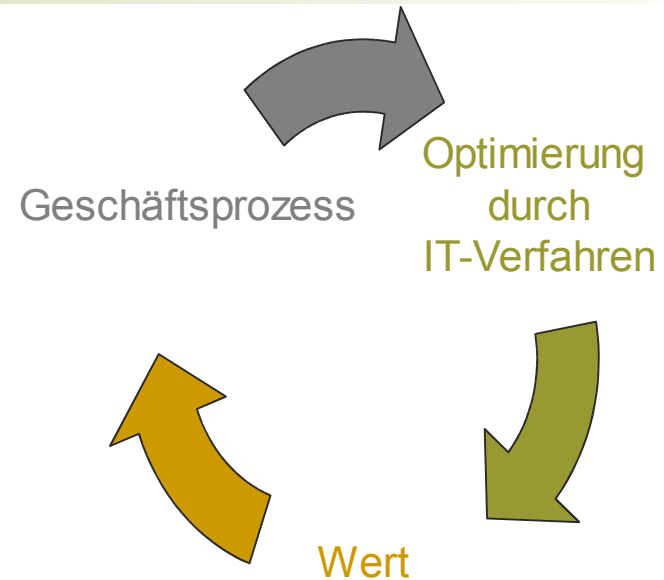
IT-Unterstützung = Investment in SW

→ **Business Value.**

Requirements Management

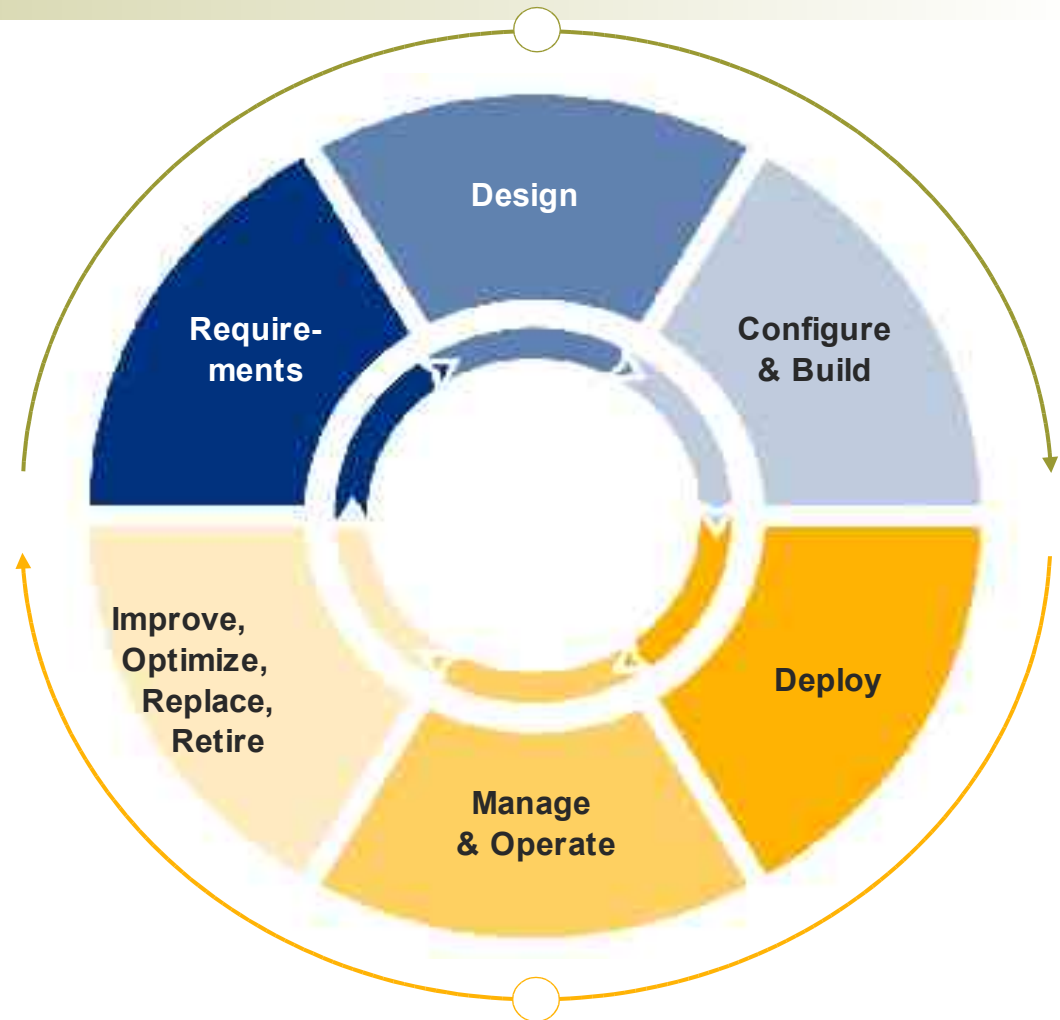
→ Schlüsselrolle :

- **Spezifikation** muss **eindeutig** sein.
- **Embrace the Change**: Kurze Iterationszyklen / wirtschaftliche Bedingungen → synchronisieren.
- Interessen der Projektbeteiligten ausgleichen

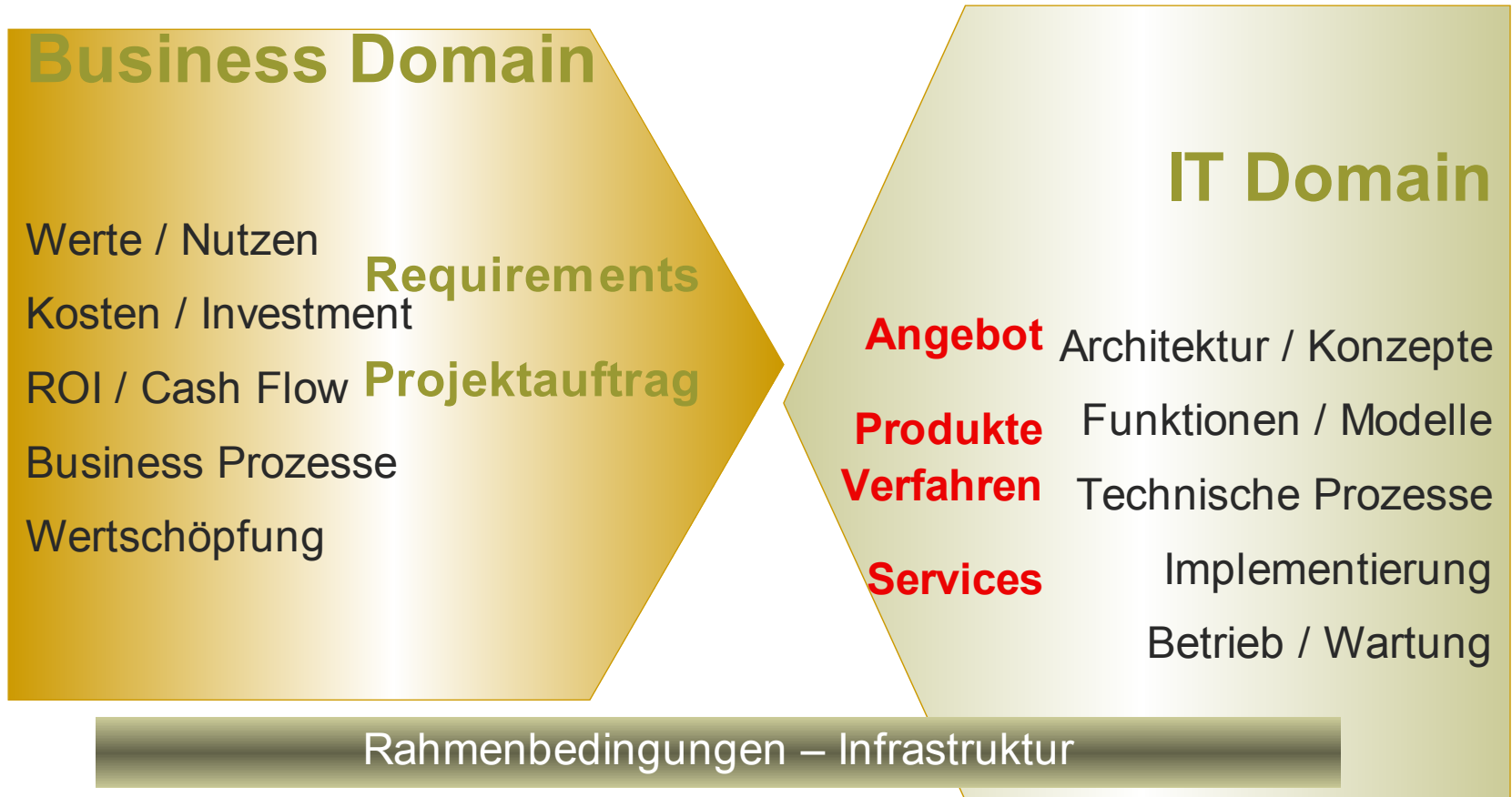


Einbettung ins Application Management

Requirements Management begleitet den Gesamtzyklus!



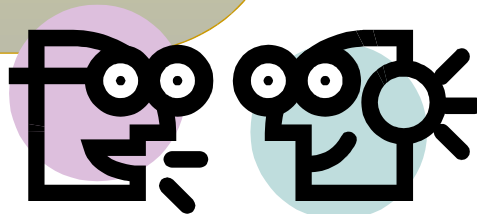
Zwei Welten ...



Ein Dialog in der Praxis ...

Lieber Herr Kunde,
wir stehen unverändert zu unserem **Festpreisprojekt**. Wir haben mehrere Anpassungen auf Ihren Wunsch ohne Preisanpassung durchgeführt.
Ihren neuen Änderungswunsch können wir allerdings nur als **kostenpflichtiger Change Request** abwickeln.

PL Auftragnehmer



Wir schätzen Ihre Arbeit und unsere konstruktive Zusammenarbeit sehr.

Allerdings gehen wir davon aus, dass so kleine Änderungen durch unseren Vertrag abgedeckt sind. Leider kann ich dafür **kein weiteres Budget** frei bekommen.

PL Auftraggeber

Customer collaboration over contract negotiation
Responding to change over following a plan

These 1: Klarer Vertrag notwendig

Jedes Projekt braucht einen **geschützten Raum**,
in dem es seine Dynamik entfalten kann.

Dieser wird durch einen **Vertrag**
mit konkreten Eckpunkten aufgespannt.

Innerhalb dieses Raumes kann eine
kooperative Kundenbeziehung gepflegt werden.

Vertragstypen: ... nach Aufwand?

Agile Projekte werden in Reinform als Aufwandsprojekte durchgeführt. Ein Auftrag wird jeweils für eine Zeiteinheit / Aufwandsmenge ohne vorher festzulegende Ergebnisse vergeben. Zum Ende der beauftragten Periode wird das Ergebnis beurteilt und ggf. weitere beauftragt. Requirements ergeben sich aus der Projekt-Interaktion zwischen Auftraggeber und Entwickler

Siehe ‚*Optional Scope Contracts*‘ - Kent Beck & Dave Cleal

Vorteile:

- ✓ Keine Risiken für Auftragnehmer
- ✓ Geringer Overhead für Analyse
- ✓ Optimale Bedarfsdeckung
- ✓ Geringer Gesamtaufwand

Nachteile:

- ✓ Ausschreibungen arbeiten anders
- Vertrauen nur bei bestehender Beziehung zwischen AN / AG
- Keine Planungssicherheit bei AG
- ✓ Komplexe Entscheidungs- und Stakeholder-Strukturen

Simplicity possible?

Ein bekannter Slogan für agile Projekte ist:

'Do the Simplest Thing That Could Possibly Work'

Was nun, wenn ‚Optional Scope Contracts‘ nicht funktionieren, weil die Kunden ein Budget beantragen müssen?

Klare Investitionsrechnungen, die Returns on Investment – ROI – ausweisen?

Keine Spezifikation → Kein Finanzierungsplan → Kein Projekt

Offenbar wird dann mehr Planung erforderlich ...

Vertragstypen: Klassischer Festpreis

Aus Auftraggebersicht sind IT-Projekte **Investments**. Diese werden hinsichtlich ihres Inhaltes, Nutzen und Aufwandes **geplant**.

Ein **Budget** muss bereitgestellt werden.

Ein **Pflichtenheft** wird intern oder extern **ausgeschrieben**.

Durchführung: Festpreisprojekt mit Lieferung zu einem Termin.
Qualitätsanforderungen sind meist implizit enthalten.

Die **Projekt-Methodik** ist meist **Wasserfall, V-Modell** oder ähnliche.

Vorteile:

- ✓ Keine (?) Risiken für Auftraggeber
- ✓ Überschaubare Planung
- ✓ Transparentes Modell
- ✓ Hohe Profit-Chancen beim AN

Nachteile:

- ✓ Hohe Analyse - Vorlaufkosten
- ✓ Spezifikation i.d.R. unzureichend.
- ✓ Hohe Risiken beim AN
- ✓ Praxis: funktioniert so nicht

Vertragstypen: Iterativer Festpreis

Gegenüber klassischen Festpreisen wird in der Planung ein gröberes Spezifikationsraster unterlegt.

Ggf. wird ein kurzes separates Spezifikationsprojekt vorgeschaltet.

Ein Budget wird inkrementell bereitgestellt → jeweils definierte Ergebnisse. Initial werden die Kernbestandteile ausgeschrieben. Über das Change Management werden Änderungen beauftragt – ggf. kostenpflichtig. *Vgl. Incremental Funding Methodology (IFM)*

Vorteile:

- ✓ Geringe Risiken für AG
- ✓ Überschaubare Risiken beim AN
- ✓ Transparente Ausschreibung möglich
- ✓ Fehlertolerante Budgetierung

Nachteile:

- ✓ Planungs-Overhead unvermeidbar
- ✓ Eingeschränkte Agilität
- ✓ **Methodenkenntnisse** erforderlich beim AG und AN

These 2: Spezifikationslücke

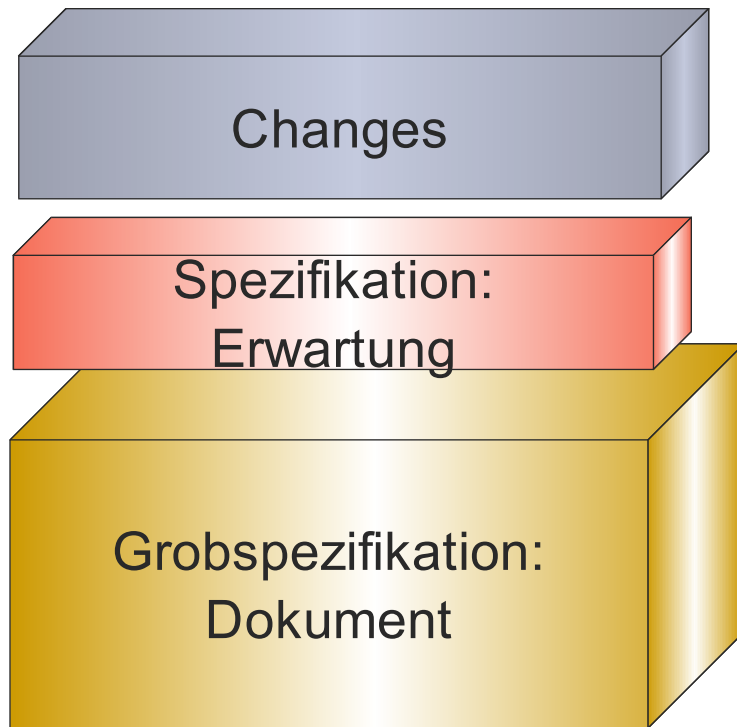
Die Spezifikation ist **nie vollständig** und präzise.

Zwischen **formaler Spezifikation** und **Erwartungen** des Auftraggebers ist eine Lücke unvermeidlich.

Diese gilt es im praktikablen Rahmen zu minimieren.

Zusätzlich ist mit **Änderungen** während des Projektablaufes zu rechnen.

Spezifikationslücke ...



Change Management →
Budgetierung beachten!

Wer trägt die Kosten?

Abwicklung über
Ausschreibung, ggf.
Festpreis

→ Wer erstellt Dokument?

These 3: Spezifikationsphase

Die **Spezifikation** eines komplexen IT Projektes ist nicht trivial und muss professionell erarbeitet werden.

In einem separaten **Spezifikations-Projekt** eines potentiellen Implementierers sollten die wesentlichen initialen Merkmale knapp definiert werden.

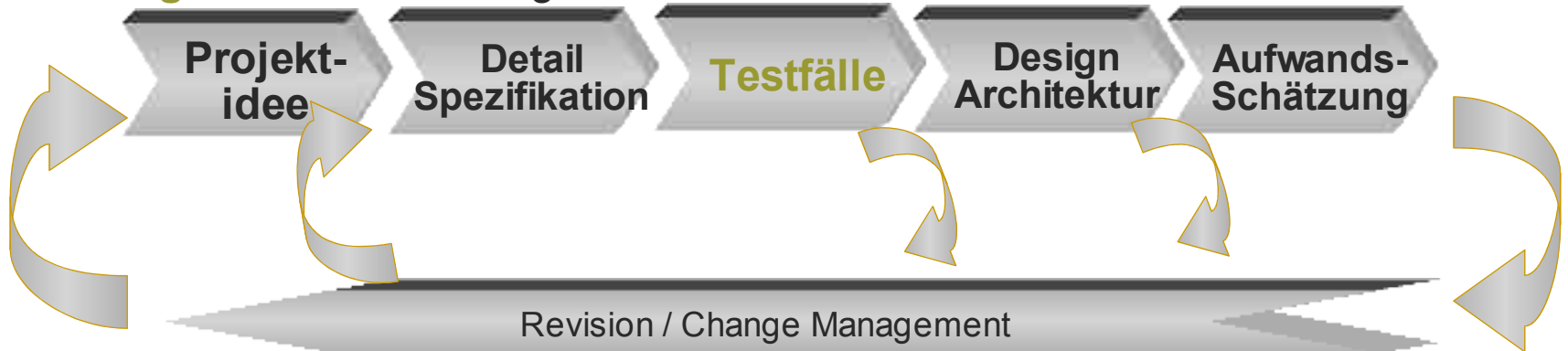
Zur Orientierung sind hier 2 – 6 Wochen anzusetzen.

Bewusst wird auf die Ausformulierung der Details verzichtet, der **Rahmen** jedoch verständlich abgesteckt.

Von der Projektidee zur Spezifikation

Die Projektidee beschreibt den Rahmen **grob** →

- **Merkmal-Spezifikationen**: fachliche Beschreibung – User Stories
- **Testfälle**: Umsetzung in Überprüfbarkeit der Anforderung
- **Design / Architektur** legt den technischen Rahmen fest incl. **NFA**



Zu jedem Zeitpunkt kann die Spezifikation angepasst werden. Diese Revisionen werden über das **Change Management** verfolgt. Nach den Detail-Aufwandsschätzungen können die **Implementierungsaufträge** erteilt werden. **Variante hierzu** → IFM

Bestandteile einer Spezifikation

- Funktionale Anforderungen
 - User Stories / Use Cases
- Nicht-Funktionale Anforderungen
-NFA
 - Mengengerüst
 - Security
 - Technische Umgebung - Produktion
 - Entwicklungs- und Test-Umgebung
 - ...

Requirements Manager

- Requirements müssen **vor Projektbeginn** bearbeitet werden ... und auch nach Projektende verfolgt werden.
- Ein Verantwortlicher – in der Regel der Auftraggeber oder Vertreter – muss die Rolle **Requirements Manager** wahrnehmen
- Methodischen Grundlagen fehlen häufig
- Ein **Requirements Coach** unterstützt den Requirements Manager



These 4: Angebot im Festpreis

Zu einer Spezifikation wird ein **Festpreis-Angebot** im beschriebenen Umfang erstellt.

Dieses beinhaltet einen kleinen **Auftragnehmer-Puffer** für inkludierte, **nicht spezifizierte Anpassungen** in begrenztem Rahmen. → Spezifikationslücke

Change Management Verfahren läuft vom Start an.

These 5: AG Budget Puffer

Der Auftraggeber sieht ein **Budget** in Höhe des Festpreis-Angebotes **zzgl. 30 - 50 %** vor. Nur so können notwendige Änderungswünsche auch implementiert werden.

Dieser AG-Puffer für anfangs noch unbekannte, **genehmigungspflichtige Anpassungen** ist **defensiv** einzusetzen.

These 6: Agile Methoden → impact

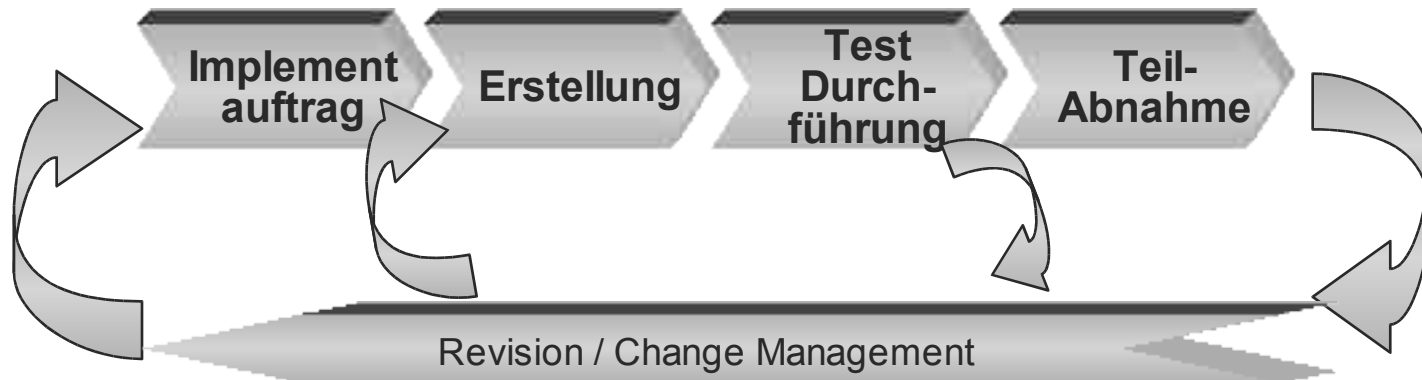
Agile Methoden sind mit weniger Verwaltungs-Overhead belastet und flexibler als konventionelle Methoden.

Wenn dem Auftraggeber als **Methoden-Laie** kein Maß und Verfahren zur Verfügung steht, wird er die Vorteile nicht verstehen und nutzen.

Das vorgeschlagene Requirements Management setzt einen **Rahmen** und ermöglicht den Einsatz agiler Methoden im Kontext kommerzieller Projekte.

Implementierungsaufträge ...

... sind auf Detailebene heruntergebrochene, überschaubare Pakete, deren Erfüllung mit mehreren Testfällen sichergestellt werden kann.
Granularität: Minimum Marketable Features - **MMF**



Durch intensive Tests und Teilabnahmen bleibt die Bedarfsdeckung überprüfbar und der Projektplan hinsichtlich Zeit und Aufwand unter Kontrolle.

These 7: Change Management

Im **Change Management Verfahren**

werden alle signifikanten Anforderungsänderungen dokumentiert.

Defect Tracking ist als Teil des Change Managements zu verstehen, da häufig Beanstandungen auf mangelnde Spezifikationen zurückzuführen sind.

Auch inkludierte Anpassungen ohne Preis-Änderungen sind **genehmigungspflichtig**.

These 8: Tool Support

Durch geeignete **Werkzeuge** wird die beste Methode des Requirements Management durchführbar:

Requirements (MMF + AE) und deren

Abhängigkeiten

Aufwand- und Nutzenreihen (ROI → NPV) Sequencing

Verknüpfte Testfälle

Change Management **Workflow**

Defect Tracking **Workflow**

[... zum Tool-Einsatz:]

Tools sind maximal so hilfreich wie das **Verfahren**, das unterstützt werden soll.



Welche Tools sind zu empfehlen?

- Standard-Textverarbeitung – unhandlich, da viele Links
- Wiki – besser, jedoch zu wenige Struktur-Elemente
- Package, z.B. RequisitePro: teuer, Lernkurve, unvollständig, nicht hinreichend anpassbar.
- DB-App. – keine fertige bekannt, eine Spec ist in Arbeit. Zur Zusammenarbeit werden noch Leute gesucht.

ROI-Awareness: IFM Crash Course

Software-Entwicklungsprojekte sind Investitionsvorhaben, die erfahrungsgemäß schwer wirtschaftlich zu managen sind:

- **Hohe Kosten: von wenigen 10 000 € bis Multimillionen**
- **Oft nur qualitativer erwarteter Nutzen**
- **Viele gescheiterte Projekte bekannt: Hohes Risiko**

Mark Denne und Jane Cleland-Huang haben eine Methode entwickelt: die den Business Value sichert und das Risiko minimiert:

Incremental Funding Methodology – IFM

IFM ist eine On-Top-Methodologie, die kompatibel zu RUP, XP, FDD und SCRUM ist. Sie hat vor allem die finanzielle Perspektive und steuert eine Makro-Ebene von größeren Projekten.

siehe www.softwarebynumbers.org

IFM – Grundlagen: MMF

Die Kernidee ist, Software-Projekte in kleinstmögliche wirtschaftlich verwertbare Komponenten herunterzubrechen.

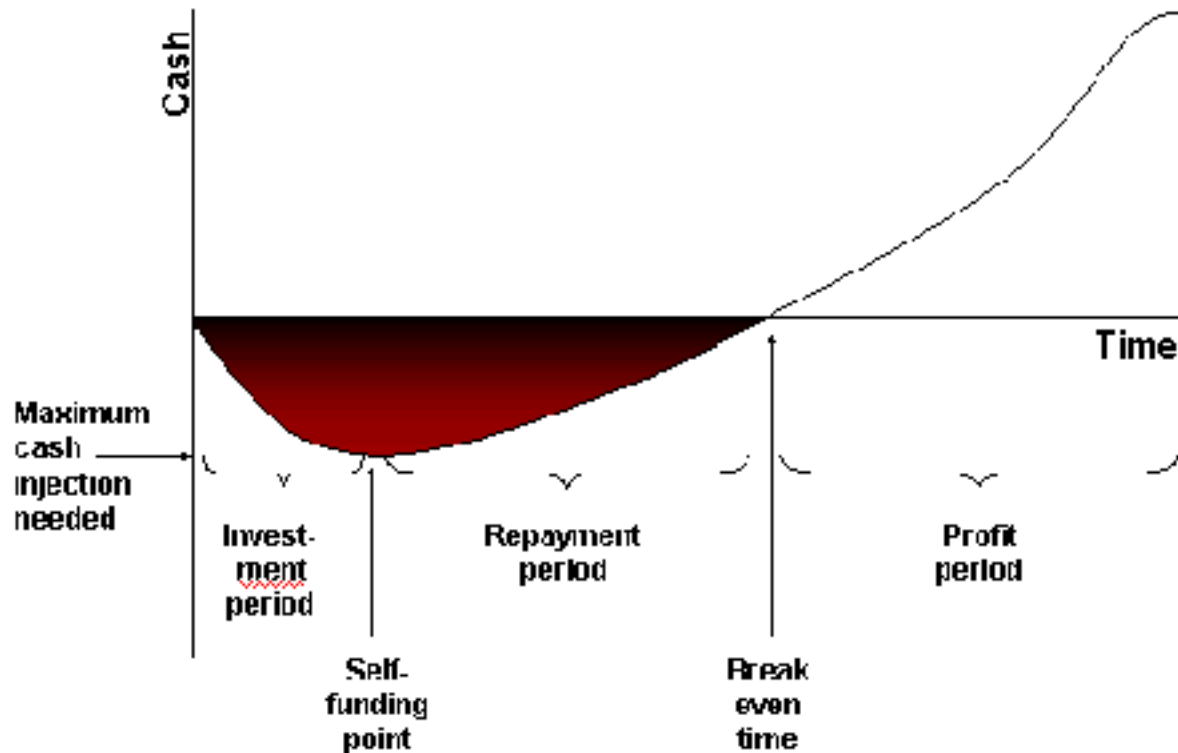
Minimum Marketable Features - MMF

Wenn nun ein Modul selbstständigen Business-Wert enthält, dann ist dieses so früh wie möglich auf Produktions-Level auszuliefern – es generiert dann bereits Returns → Einkommen. So kann sich das Projekt (weitere MMFs) selbst finanzieren.

NPV – Net Present Value:

Wenn ein Investment zunächst Kosten verursacht, dann aber periodisches Einkommen abwirft, so sind diese für einen Betrachtungszeitraum – z.B. 3 Jahre – diskontiert zu einem Gegenwartswert darstellbar. Dies ist auf Ebene MMF möglich.

IFM – Grundlagen: Ziele



Konkurrierende Ziele:

- hoher NPV
- Minimaler ‚Maximum Cash Injection‘
- Früher ‚Self Funding Point‘
- Kurze ‚Break Even Time‘

IFM – Grundlagen: Sequencing

Mehrere **MMF** können voneinander abhängig oder auch nicht sein. Daraus ergeben sich alternative Ketten und Kombinationen von **MMFs** auch **Strand** genannt. Für einen **Strand** kann ebenfalls ein **NPV** errechnet werden.

Da Kapazitäten begrenzt sind gilt es, die optimale Reihenfolge der **MMFs** zu ermitteln. Dazu dient die **IFM Heuristic**: einem Verfahren zur Optimierung der Sequences.

So wird ermittelt, welche **MMFs** in welcher Reihenfolge erstellt werden. Als Planungsergebnis liegt dann der **SANPV** vor.

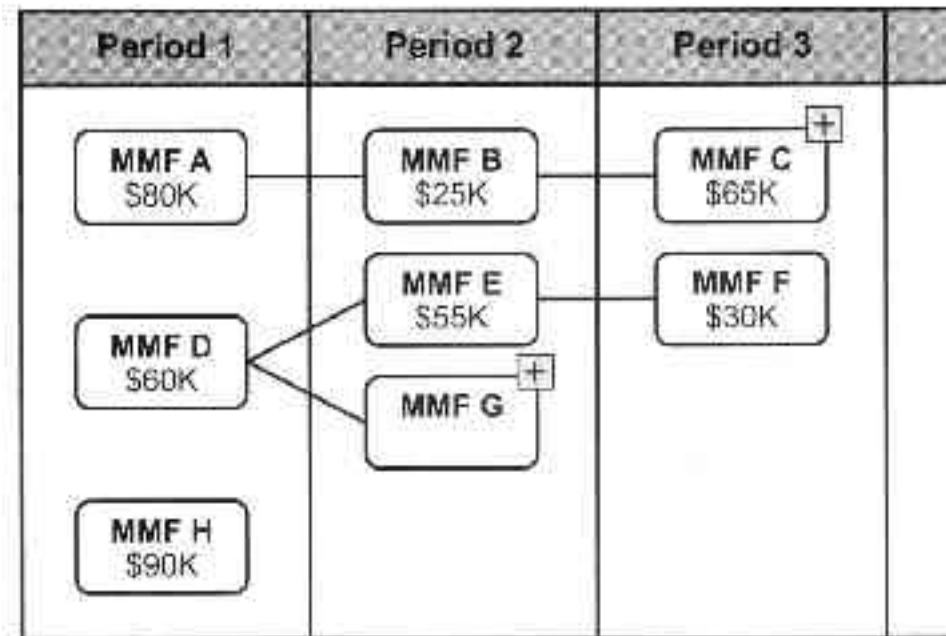
SANPV: Sequence Adjusted NPV

IFM – Grundlagen: Sequencing...

Das Beispiel zeigt 4 **Strands**, wobei der 4. nur aus einem MMF besteht.

Der MMF G hat nur einen qualitativen Nutzen.

MMF C hat einen gemischten Nutzen.



⊕ Intangible benefits

IFM – Grundlagen: MMF & AE

Nun sind ggf. einzelne Module gemeinsame Voraussetzung für mehrere MMFs. Diese sind aber nicht vermarktbar. IFM nennt diese Elemente:

Architectural Elements - AE

AEs werden genauso wie **MMFs** behandelt, sie liefern lediglich keine Returns.

Ein MMF hat meist einen **kalkulierbaren** und einen **rein qualitativen Nutzen (Intangibles)** : Entweder das Eine, das Andere oder in einem Mischungsverhältnis.

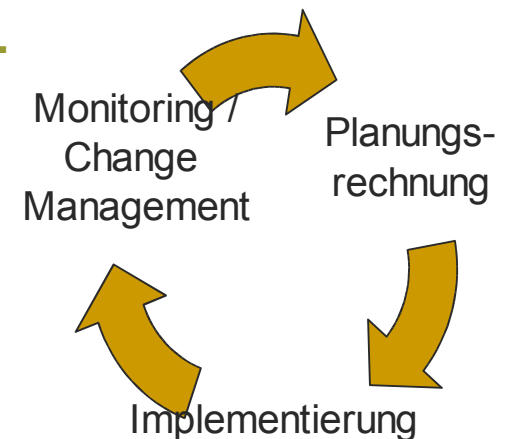
IFM benutzt die **Pairwise Comparison Table**, um den qualitativen Nutzen zu quantifizieren und damit rechenbar zu machen.

IFM – Grundlagen: Inkrementelles ...

IFM bezieht **Risiken** mit in die Kalkulation ein: Es werden **Faktoren** geschätzt, und diese werden **quantifiziert** eingerechnet.

Die **Look-Ahead Projektion** geht über eine zu bestimmende **Anzahl von Perioden** bei der Ermittlung von SANPV und damit **Planungsentscheidungen**.

Allerdings kann nach jeder Periode das Verfahren **neu gerechnet** werden, um **Störungen** und **Änderungen** zu berücksichtigen. So werden **Fehlentwicklungen** sehr schnell sichtbar und **Korrekturmaßnahmen** können eingeleitet werden.



IFM – Kompatibilität

IFM ist neutral zu Architektur und kompatibel zu anderen Methodologien. Es unterstützt RUP, Six SIGMA, XP, SCRUM und Feature Driven Development (FDD):

IFM behandelt nur Makro-Building Blocks – Die Entwicklung der jeweiligen MMFs und AEs ist Sache der weiteren. Das Mapping der Begriffe und Methoden wird weitgehend erläutert.

Doch ist IFM auch kompatibel mit in Deutschland gebräuchlichem Projekt-Controlling ???

IFM setzt Project Accounting voraus

In Deutschland ist es gängige Praxis, Die **Projektkosten** in Form von **Mittelabfluss / Budget** zu überwachen, seltener eine **Vollkostenrechnung**. **Unüblich** ist dagegen, den **Projektnutzen** als **periodengerechte Revenues** dem Projekt **gutzuschreiben** ... also **vollwertiges Project Accounting**. → **Voraussetzung für Self Funding**

Was nun zunächst wie eine **Hürde** aussieht, entpuppt sich im zweiten **Blick** als **Chance**:

Mit einem **Projekt Accounting** löst sich das Projekt aus seiner **Stellung** als **Kostenfresser** und wird zum **Leistungsträger**!

ROI informed Decision Making wird möglich.

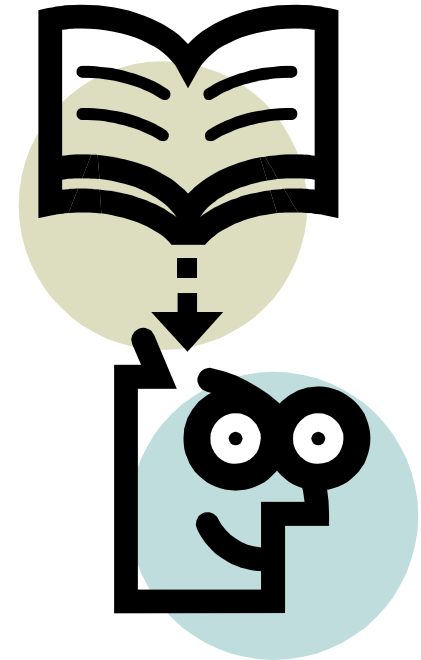
IFM Konsequenzen

Die beabsichtigte Nutzung von IFM führt zu:

- Project Accounting – Nutzenbewusstsein
- höherer Wirtschaftlichkeit
- Effizientes Gegensteuern bei Fehlentwicklungen
- IFM kompatibles Requirements Management

Es werden nicht mehr nur die Kosten für ein Modul ermittelt, sondern der Nutzen eines MMF wird in einer Ertragsreihe dargestellt.

→ Mehr erfolgreiche Projekte!



Praktische Projektvergabe mit IFM

Anforderer nennen ihren IT-Bedarf – nach MMFs – und bewerten diesen mit entsprechenden Reihen an Revenues.

- Werden die Nutzenreihen zu hoch angesetzt, so muss der Anforderer seine überzogenen Zahlen später rechtfertigen.
- Setzt er sie gar nicht oder zu niedrig an, muss er dies ebenfalls begründen – er könnte als Pessimist und Blockierer seinen Ruf einbüßen.

Zur IFM-kompatiblen Bedarfsfeststellung erhält die Fachabteilung Unterstützung von IT-Koordinatoren / Requirement-Coaches.

Projekte (oder einzelne MMFs) **werden ausgeschrieben**, intern wie extern.

Wird ein **unternehmensinterner Leistungserbringer** gewählt, so werden die positiven **Erträge** zwischen AN und AG **geteilt**.

Externe Leistungserbringer könnten über einen **Bonus-Schlüssel** beteiligt werden.

Wirtschaftliche Gesichtspunkte

Software-Entwicklung ist dann effektiv, wenn sie den Bedarf befriedigt, die Qualität im notwendigen Rahmen sicherstellt, den Prozess überwacht und alle vermeidbaren Tätigkeiten unterlässt.

So kann aus einem ungeliebten Aufwandsbudget eine profitable Investition werden! Kostenvorteile durch agile Methoden müssen durch solides Requirements Management verständlich und nutzbar werden.

Erforderlich ist ein Umdenken seitens der Auftraggeber und der Auftragnehmer. Der Wille zur Effizienz muss mit methodischer Reife gepaart werden.

Beachtung der gezeigten Schritte hilft, dies zu erreichen. Der Mehraufwand durch intensive Spezifikations- und Testarbeit ist unvermeidbar, um eine geordnete und zielgerechte Bedarfsdeckung zu erreichen. Anderenfalls führen die Reibungsverluste erfahrungsgemäß zu einem weit höherem Aufwand.