

Die vergessene Aktivität im RE: die Zusammenführung von Bedürfnissen und Lösungen (oder die Priorisierung von Anforderungen gemäß ihres Beitrags zum Kundennutzen)

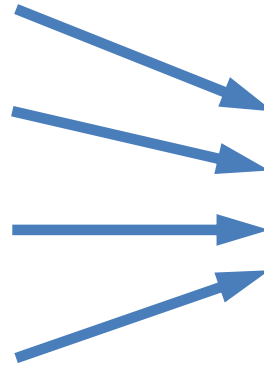
Sixten Schockert



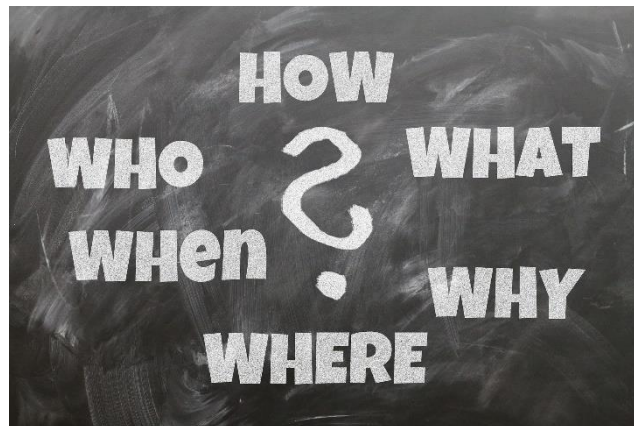
Universität Stuttgart
Lehrstuhl für ABWL und
Wirtschaftsinformatik II

Agenda

- Warum?
- Was (genau)?
- Wie?
- Wohin?



**Zusammenführung
von Bedürfnissen
und Lösungen**



Probleme – Lösungen – Anforderungen

“Das Requirements Engineering als **gestaltende Disziplin** ist dafür verantwortlich, dass mit allen relevanten Stakeholdern

- die **Problemstellungen**, die eine Software lösen soll, definiert und abgestimmt werden,
- **Lösungen** für die identifizierten Problemstellungen definiert und abgestimmt werden,
- **Anforderungen** an die Lösungen definiert und abgestimmt werden.“

➔ „Anforderungen beschreiben die notwendigen Eigenschaften, die ein System aufweisen muss, um die Lösung für ein Problem zu sein. [...] **Anforderungen können erst formuliert werden, wenn Problem und Lösung definiert wurden.**“

Bedürfnisse vs. Lösungen

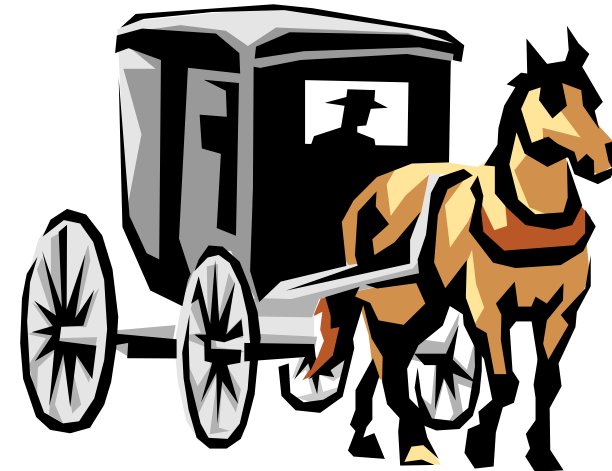
- **Bedürfnisse um 1900**

- Bequeme Fahrt
- Sportliches Aussehen
- Hohe Geschwindigkeit
- Sparsamer Verbrauch

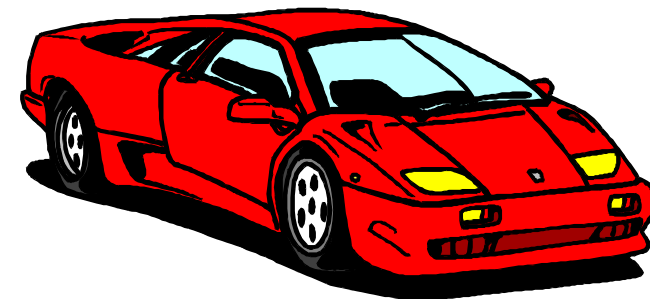
- **Bedürfnisse um 2000**

- Bequeme Fahrt
- Sportliches Aussehen
- Hohe Geschwindigkeit
- Sparsamer Verbrauch

- **Lösung um 1900**



- **Lösung um 2000**



Trennung von Bedürfnissen und Lösungen

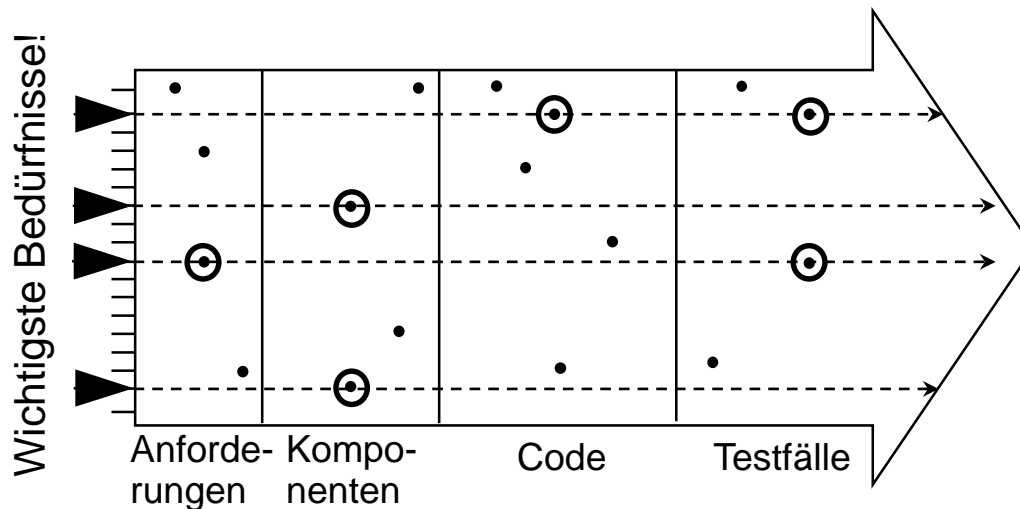
- Klaro! Alte RE-Weisheit!
- What vs. How!



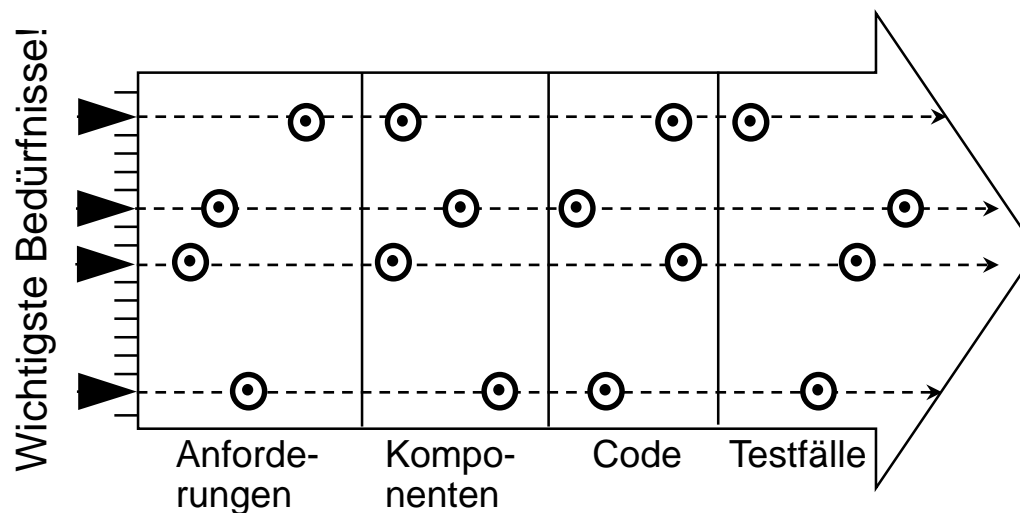
WARUM analytisch trennen?

- zu lösende Probleme identifizieren
- die Stakeholder/Kunden verstehen
- nicht zu früh in Lösungen denken
- Garbage in, Garbage out vermeiden (GiGo-Prinzip)
- verschiedene Perspektiven einnehmen
- Verständlichkeit vs. Präzision
- Bedürfnisse ändern sich langsamer als Lösungen

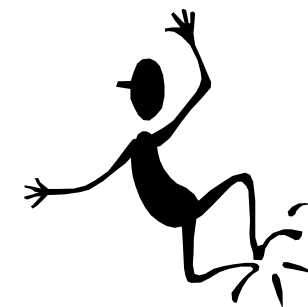
GiGo vermeiden: Kohärente Entwicklung



Nutzen
zufällig,
unregel-
mäßig,
unsicher



Nutzen
bewusst,
regel-
mäßig,
sicher



Zusammenführung von Bedürfnissen und Lösungen

WARUM synthetisch zusammenführen?

- Um wirklich GiGo zu vermeiden, muss man genau an den Lösungen arbeiten, welche die (wichtigsten) identifizierten Probleme lösen
- Liefert die für die Definition und Abstimmung von Anforderungen notwendige Begründung, warum (!) man gerade diese Lösungen zu Anforderungen detailliert und in der Software umsetzt (und nicht andere)
- Gemeinsames Verständnis von Problem- UND Lösungsraum unter allen Stakeholdern erreichen



Priorisierung von Lösungen

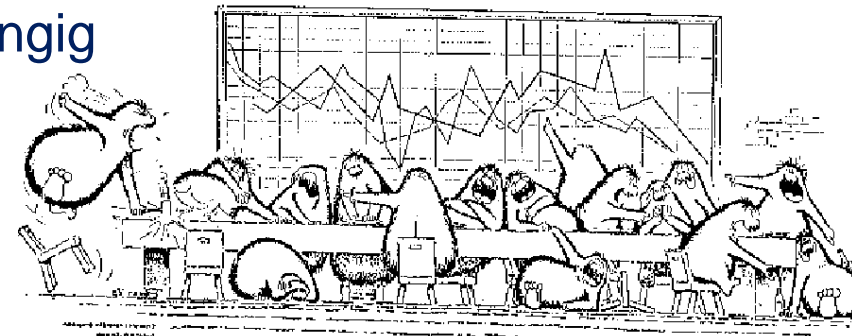
- = **Beurteilung der Wirkungen der Lösungen auf die Bedürfnisbefriedigung**
- Essentiell bei Entwicklung in Inkrementen zur Reduktion von Unsicherheiten und zur Steuerung in agilen Projekten
 - Zentrale Priorisierungskriterien:
 - Aufwand / Kosten / Zeit
 - Risiko / Unsicherheit
 - Abhängigkeiten / Synergien
 - **Business Value / Kunden-/Stakeholdernutzen (!)**

(Siehe für eine Klassifikation von Priorisierungskriterien Riegel, N., Doerr, J.: A Systematic Literature Review of Requirements Prioritization Criteria. In: S.A. Fricker, K. Schneider (Hrsg.): REFSQ 2015, LNCS 9013, pp. 300–317, 2015)

➔ Wenn die Nutzeffekte der Lösungen bei der Auswahl der im nächsten Inkrement umzusetzenden Anforderungen berücksichtigt werden sollen, dann muss man die Lösungen (wieder) mit den Bedürfnissen zusammenführen!

Möglichkeiten der Zusammenführung

- **Verbindung von Problem- und Lösungsraum:**
zweidimensionale Darstellungsmittel nötig für ein „in Beziehung bringen“ von zwei Artefakten
- **Zweck-Mittel-Beziehungen:**
 - 1:1 in Kausalformulierungen (z. B. User Stories)
 - 1:n in Ursache-Wirkungs-Diagrammen (z. B. Maximum-Value-Table)
 - m:n in Matrixdiagrammen (z. B. Priorisierungsmatrix)
- **Zentrale Anforderungen:**
 - Systematisch & strukturiert
 - Nachvollziehbar & personenunabhängig
 - Flexibel & pragmatisch einsetzbar
 - Toolunterstützt (?)



Implizite Zusammenführung

- Mündlich bzw. in Diskussion
- Oft in (User) Stories

Lösung

Bedürfnis

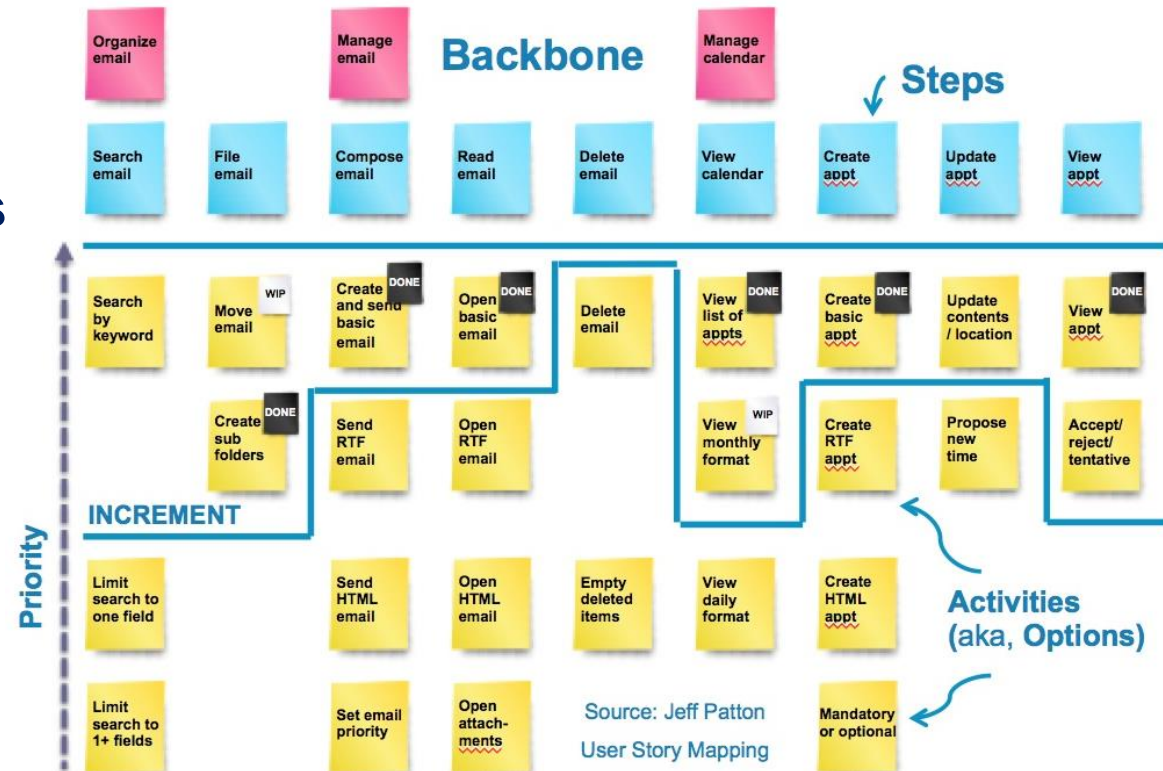
„Als <Rolle> möchte ich <Ziel/Wunsch>, um <Nutzen/Vorteil> zu erreichen“ (Cohn, M.: User Stories: für die agile Software-Entwicklung mit Scrum, XP u.a. (2010))

- ☺ Verständlich, einfach, natürlichsprachig, **verbreitet** (Scrum)
- ☺ Information über Grund eines Lösungswunsches
- ☹ Nur 1:1-Beziehungen zwischen Bedürfnissen und Lösungen
- ☹ Nur **geringe Handlungsunterstützung zur Priorisierung**, denn dazu müssten die verschiedenen Nutzeffekte untereinander priorisiert werden, um auf dieser Basis die besten Lösungen für diese Nutzeffekte zu ermitteln

[gilt analog für Position Statements, die für Epics als Zusammenfassung mehrerer User Stories gebildet werden können]

(User) Story Maps

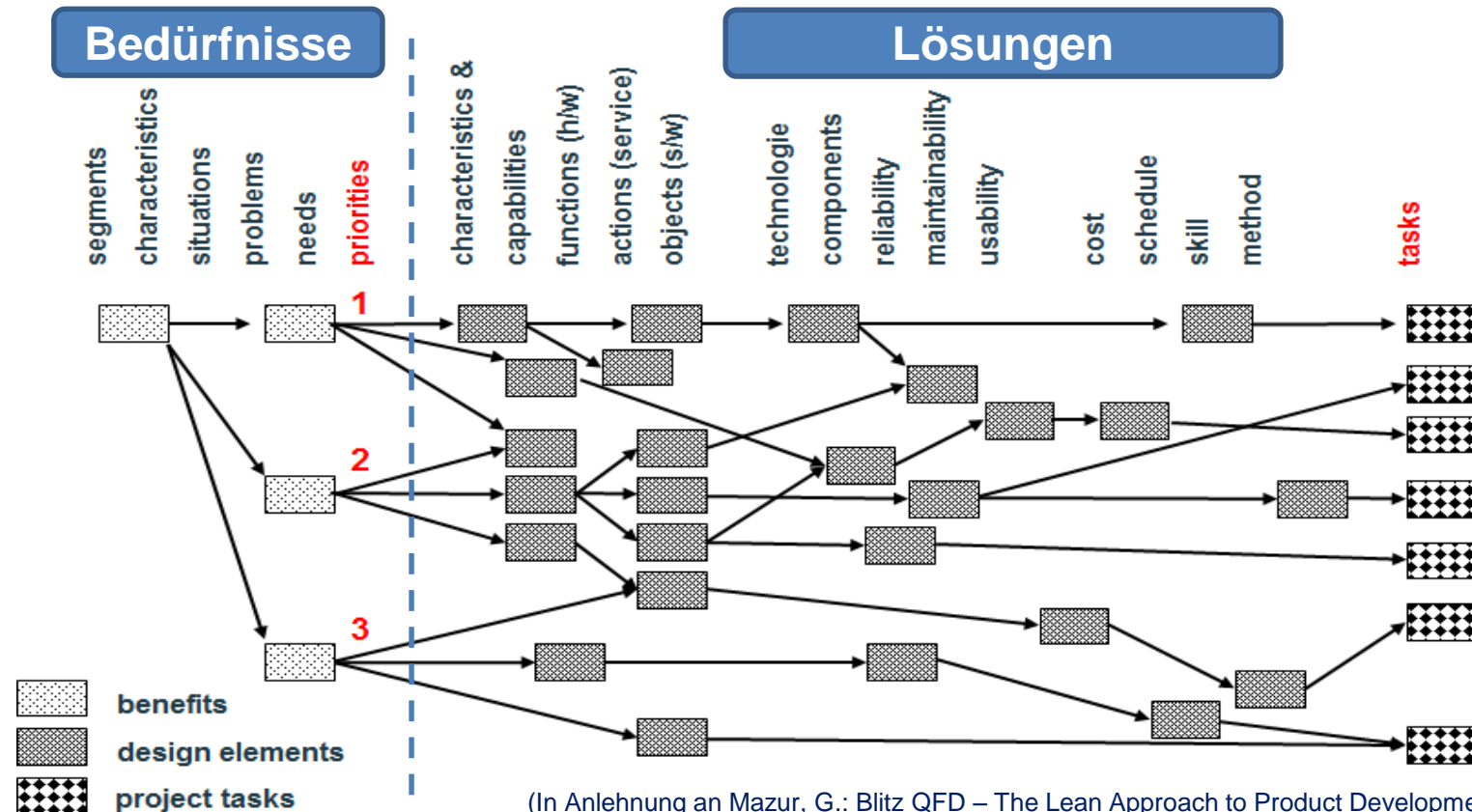
- Organisiert und visualisiert die Stories und Diskussionen
- Fokus auf Lösungen
- Bedürfnisse „versteckt“ in Backbone bzw. in sog. „Outcomes“, welche die Inkremente erfüllen (sollen)



Quelle: <http://cmforagile.blogspot.de/2015/07/story-telling-with-story-mapping.html>;
Patton, J.: User Story Mapping. O'Reilly, Sebastopol CA 2014

- ☺ Sehr flexibel und gestaltbar
- ☺ **Projektmanagement**-Unterstützung (z. B. Releaseplanung)
- ☹ Gefahr der Unübersichtlichkeit und Willkür
- ☹ **Priorisierung oft nur auf Lösungsseite ohne Berücksichtigung der Bedürfnisse, da diese (oft) nur „versteckt“**

Maximum-Value-Table



(In Anlehnung an Mazur, G.: Blitz QFD – The Lean Approach to Product Development. In: Proceedings of the World Conference on Quality and Improvement. Milwaukee WI, ASQ, Mai 2012.)

- ☺ Flexibel einsetzbar, auch „nur“ Projektmanagement („tasks“)
- ☺ Trennung (inkl. Priorisierung der Bedürfnisse) und Zusammenführung
- ☹ **Keine (explizite) Priorisierung auf Lösungsseite, aber Visualisierung durch „Lösungen mit vielen Pfeilen“**
- ☹ Gefahr der Unübersichtlichkeit (insb. bei vielen Beziehungen/Pfeilen)

Priorisierungsmatrix (1)

Problemraum: ①
Nutzeffekte / Kundenbedürfnisse / Marktanforderungen

Priorisierung der Kundenbedürfnisse

Lösungsraum:
Produktanforderungen d. h. insb. funktionale Anforderungen und Qualitätsanforderungen

Zusammenführung:
Priorisierung von Lösungen gemäß ihrer Nutzeffekte (ihres Beitrag zur Bedürfnisbefriedigung) ④

Legende:

Die Erfüllung des Kriteriums [oben] trägt
 9 : zwangsläufig und sehr stark
 3 : merkbar aber evtl. eingeschränkt
 1 : möglicherweise und nur bedingt
 0 : nur mittelbar oder ggf. gar nicht zur Erfüllung der Anforderung [links] bei.

| Kundenanforderungen | Wichtigkeit | Anzeige | Ausfallsicherung | Datenhaltung | Datensicherheit und Datenschutz | Installation und Konfiguration | Kontrolle und Prüfung | Modularität | Schnittstellen und Kommunikation | Überwachung | |
|---------------------------|-------------|---------|------------------|--------------|---------------------------------|--------------------------------|-----------------------|-------------|----------------------------------|-------------|-------|
| Administration | 13,60% | 9 | 3 | 3 | 1 | 9 | 3 | 3 | | 9 | 9 |
| Event Handling | 12,90% | | | 1 | | | 9 | | 9 | 9 | |
| Fachliche Betriebsführung | 12,40% | 9 | 3 | | 3 | 1 | 3 | | 3 | 3 | 1 |
| Flexibilität | 11,10% | | | 3 | | 3 | | 9 | 9 | | 3 |
| Sicherheit | 9,30% | 1 | 3 | 3 | 9 | 1 | 1 | | | 3 | 3 |
| Compliance | 8,00% | | 3 | 3 | 9 | | 1 | | | 1 | |
| Verfügbarkeit | 7,60% | | 9 | | | | 3 | 1 | | 9 | 3 |
| Standards verwalten | 6,50% | | | | | 3 | | 1 | 9 | | 9 |
| Individualisierung | 5,70% | 3 | | 3 | | 3 | | 3 | | | |
| Usability | 5,40% | 9 | | | | 9 | 3 | | | 3 | 3 |
| Abrechnungsunterstützung | 4,40% | 1 | | | | | 9 | | 3 | 3 | |
| Support | 2,20% | 1 | | | | 3 | 1 | | | 1 | 3 |
| Preismodell der Appliance | 0,80% | | | | | | | 3 | | | |
| | 99,90% | | | | | | | | | | |
| absolute Bedeutung | | 3,2 | 2,0 | 1,6 | 2,1 | 2,7 | 2,9 | 1,7 | 3,2 | 4,1 | 3,0 |
| relative Bedeutung | | 11,9% | 7,5% | 5,9% | 7,8% | 10,2% | 11,0% | 6,6% | 12,3% | 15,5% | 11,3% |
| Rang | | 3 | 8 | 10 | 7 | 6 | 5 | 9 | 2 | 1 | 4 |

(Helferich, Pelzl, Schockert: Unterstützung von KMU bei der Erbringung komplexer Mobilitätsservices in Kooperationsnetzwerken. In: 44. Jahrestagung der Gesellschaft für Informatik e.V.: Informatik 2014: Big data – Komplexität meistern)

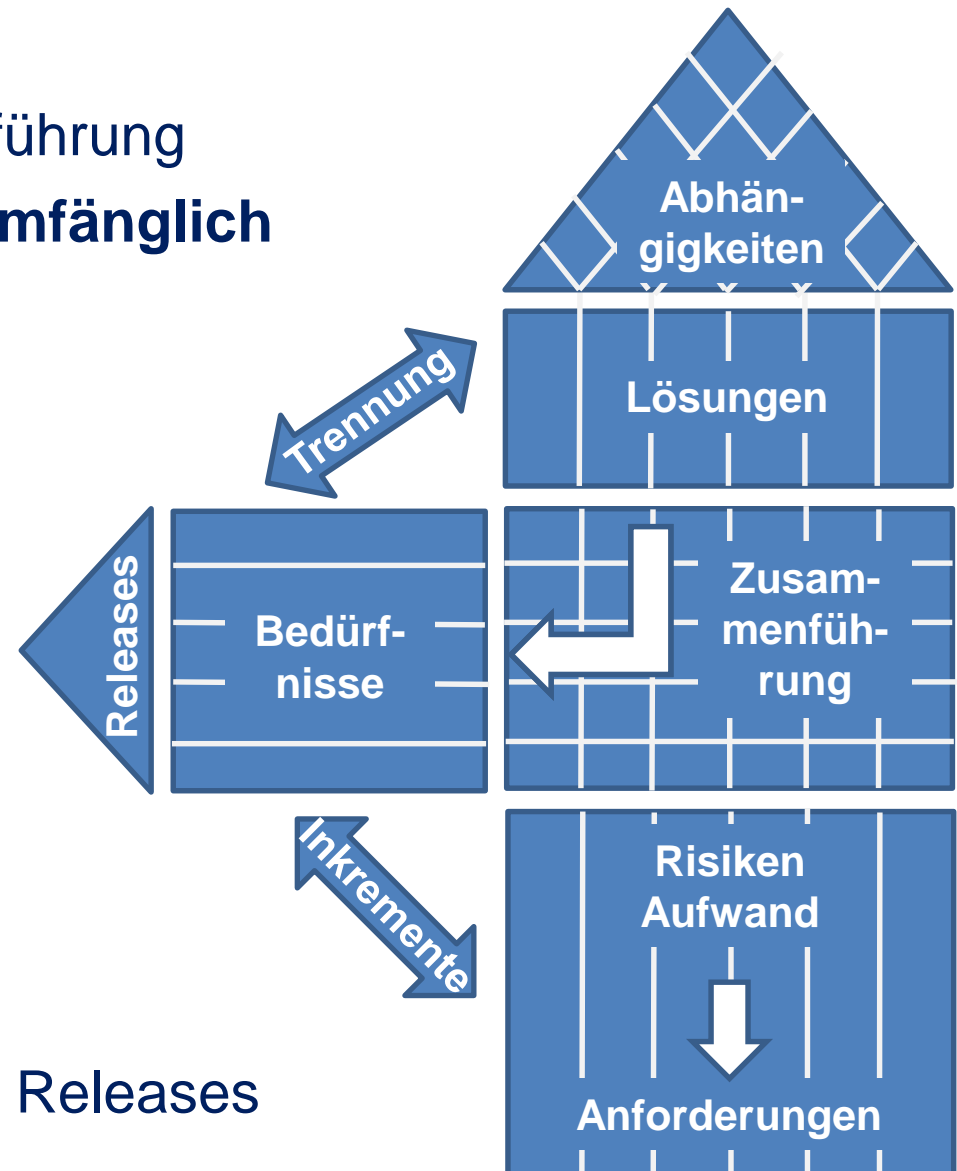
Priorisierungsmatrix (2)

- ☺ Trennung von Bedürfnissen und Lösungen
- ☺ Separate Priorisierung der Bedürfnisse
- ☺ **Zusammenführung:**
Beurteilung der Lösungen im Hinblick auf ihre Nutzeffekte (ihres Beitrags zur Nutzenstiftung / Bedürfnisbefriedigung)
→ **Explizite Priorisierung der Lösungen**
- ☺ n:m-Beziehungen darstellbar d. h. Berücksichtigung von
 - Lösungen, die mehrere Bedürfnisse befriedigen bzw.
 - alternativen Lösungen für dasselbe Bedürfnis
- ☺ Identifikation von Lösungen, die nur eingeschränkten Nutzen bringen bzw. von Bedürfnissen, die nicht adäquat erfüllt werden

- ☹ „Nicht sexy“ – „Old School“ – „Nicht vergnügungssteuerpflichtig“
- ☹ Aufwand-Nutzen-Verhältnis bei vielen Beziehungen?
- ☹ Overengineering durch „Mathematik“

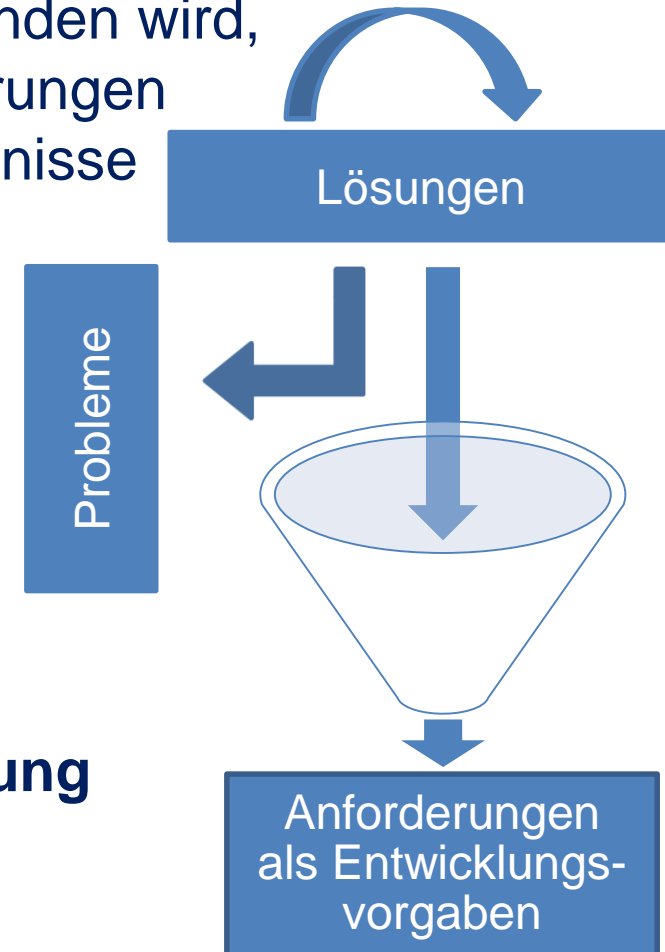
Priority Map

- Trennung und Zusammenführung
- **Priorisierungskriterien umfänglich enthalten**
- Bewertung der Lösungen und Konkretisierung zu Anforderungen als Entwicklungsvorgaben
- Visualisierend
- **Inkrementell wachsend**
- **Kommunikations- und Dokumentationsmittel**
- Situativ gestaltbar
- Steuerung & Kontrolle von Releases und Inkrementen



Fazit und Ausblick

- **Anforderungen detaillieren Lösungen für Probleme**
- Wenn RE als Gestaltungsdisziplin verstanden wird, dann ist für die Begründung von Anforderungen die Wirkung der Lösungen auf die Bedürfnisse essentiell
- **Priorisierung gemäß Nutzeffekten = Zusammenführung von Bedürfnissen und Lösungen**
- Essentielle Aktivität des RE insb. bei inkrementeller Entwicklung und zur Steuerung & Kontrolle agiler Projekte
- **Situative Ausgestaltung und Verwendung** der Möglichkeiten der Zusammenführung von Bedürfnissen und Lösungen nötig



Vielen Dank für Ihre Aufmerksamkeit!

