

# Anforderungsmanagement / Anforderungs-Engineering

# Agenda

- Kurze Vorstellung des Referenten
- Problemstellung / Ausgangslage
- Den Kunden verstehen?
- Checkliste für die Anforderungsermittlung
- Wofür steht RE/RM?
  - Welche Vorteile habe ich durch RE/RM?
  - Welche Werkzeuge können hier unterstützen?
  - Kostenfreie Tools?
- Diskussion  
Software-Entwicklung/Web-Entwicklung und RE/RM?
- Quellen

# Kurze Vorstellung des Referenten



- Name Jan Witte
- Freiberufler im Bereich RE/RM (seit ca. 7 Jahren im Bereich Anforderungsmanagements/Engineerings aktiv)
- aktuell eingesetzt bei:
  - Cross Content Media
  - T-Systems Enterprise Services (Wolfsburg)
  - Audi AG (Ingolstadt)
  - Volkswagen (Wolfsburg)
- Kontaktdaten
  - [jwitte@cross-content.com](mailto:jwitte@cross-content.com)
  - [extern.jan.witte@audi.de](mailto:extern.jan.witte@audi.de)
  - [jan.witte@t-systems.com](mailto:jan.witte@t-systems.com)
- Mobil: 0171/6415078

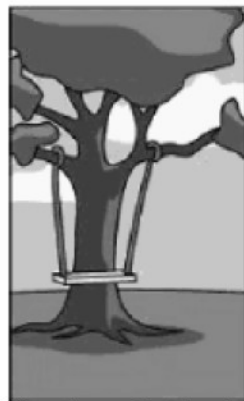
In folgenden Organisationen vertreten:

- Gesellschaft für Systems Engineering e.V.
- INCOSE
- [www.re-wissen.de](http://www.re-wissen.de)
- Systems Engineering Society of Australia (SESA)
  - *The Systems Engineering Society of Australia (SESA) is a Technical Society of Engineers Australia. Technical Societies provides a forum for mutual technical development, networking, expanding and sharing knowledge with like-minded professionals. Members of Technical Societies provide expert opinion on a broad range of issues and have a capacity to exert considerable influence about professional practice in a changing world market*
- ASQF
  - *Der ASQF e.V. ist ein gemeinnütziger Verein und dient dem Austausch von Erfahrungen, Kenntnissen und Ideen auf den Gebieten Software-Entwicklung und Qualitätsmanagement.*

# Problemstellung / Ausgangslage



How the customer explained it



How the project leader understood it



How the engineer designed it



How the programmer wrote it



How the sales executive described it



How the project was documented



What operations installed



How the customer was billed



How the helpdesk supported it



What the customer really needed

- Häufig gemachter Fehler:
  - der Kunde bekommt das, was er braucht, aber nicht das, was er will ;-)

Worauf sollte geachtet werden:

- Realistische Projektpläne, die sauber heruntergebrochen werden
- Zusammenstellung der Anforderungen und Bewertung hinsichtlich unterschiedlicher Sichtweisen
- Vorschlag zur Priorisierung der Anforderungen (kritische Terminpläne können so eingehalten werden)
- Risikomanagement mit dem Kunden (Einbauen von Puffern reicht nicht aus)
- Kommunikation
- Interessen der einzelnen Stakeholder

# Checkliste für die Anforderungsermittlung

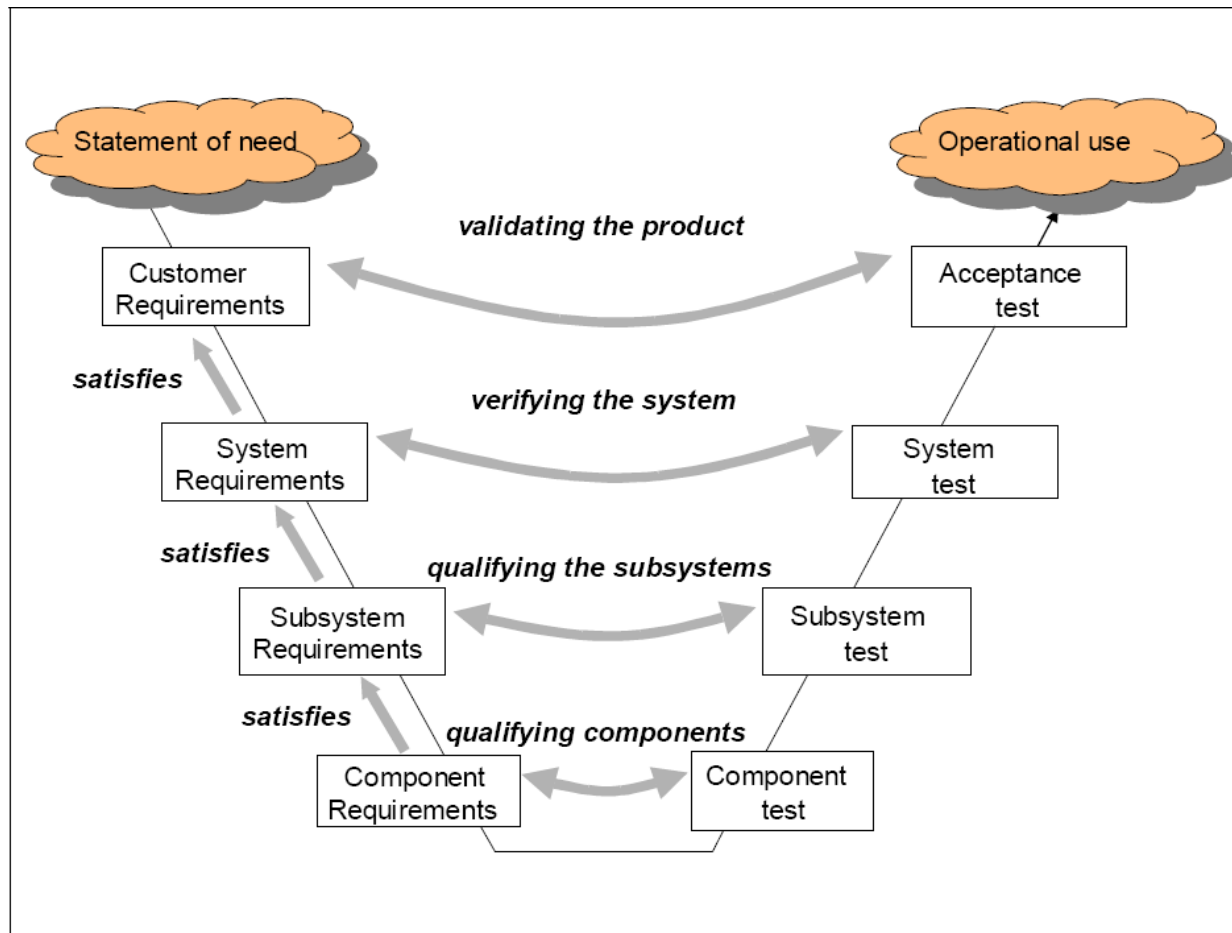
- Wer ist der Kunde für das zu entwickelnde System?
- Welches Problem soll das System lösen?
- Welchen Nutzen kann der Kunde aus einer erfolgreichen Lösung ziehen?
- Was exakt muss das Projekt liefern?
- Kommen die Anforderungen von der richtigen Seite?
- Wer (Personen und Gruppen) ist am Erfolg der Lösung interessiert?
  - Gibt es Konflikte zwischen den Parteien?
- Wie sieht die Umfeld eines Systems aus?
- Sind die Anforderungen bekannt und verständlich? → und auch messbar und testbar?

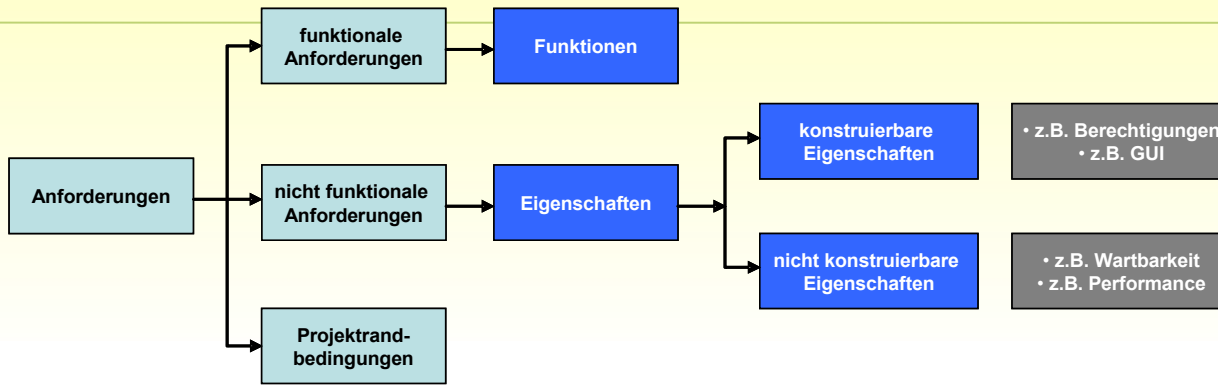
# Checkliste für die Anforderungsermittlung

- Sind die späteren Benutzer bekannt?
- Wie werden die Benutzer mit dem System kommunizieren?
- Sind die Anforderungen technisch umsetzbar?
- Wie viel Zeit steht für das Projekt zur Verfügung?
  - Was passiert, wenn sich das Projekt verzögert?
- Welche Anforderungen können sich (wie stark) verändern?
- Wie wird sich die Umgebung während des Projekts ändern?
- Welche Probleme könnten durch das neue System entstehen?

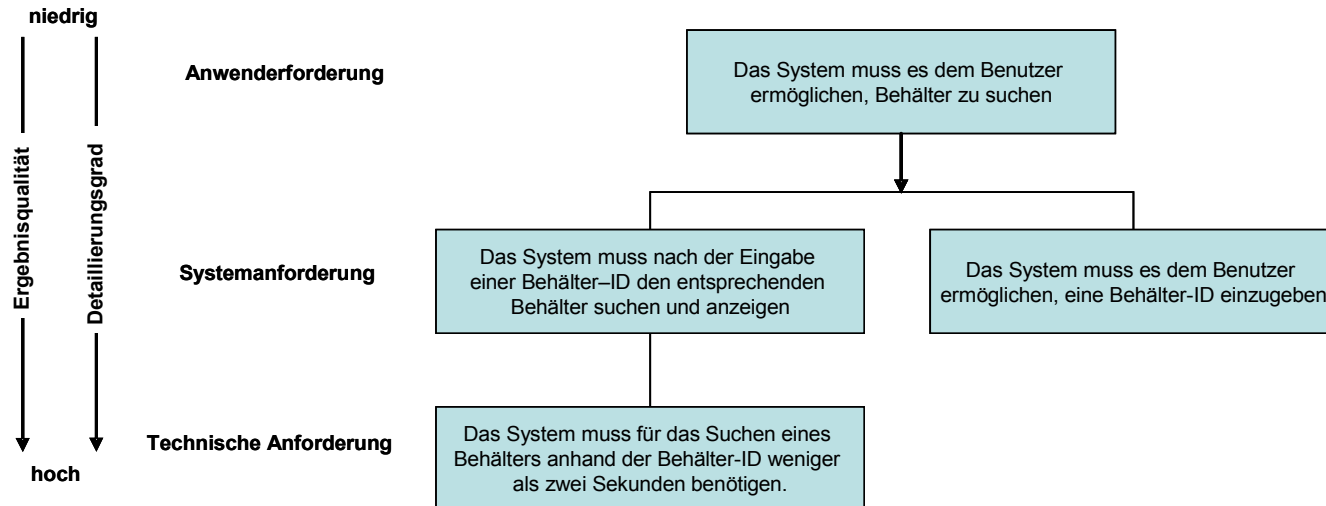
- RE steht für „Requirements Engineering“ (Erheben von Anforderungen)
  - Iterativer Prozess zur systematischen Annäherung und Bestimmung der Anforderungen (Requirements)
  - Process of finding out, analyzing, documenting, and checking requirements [Sommerville]
  - Ziel ist es, die Anforderungen des Auftraggebers an das zu entwickelnde System zu ermitteln
    - Diese Disziplin lässt sich in folgende Bereiche aufteilen:
      - Anforderungserhebung
      - Anforderungssanalyse
      - Anforderungsspezifikation
      - Anforderungsverifikation und Validierung
- RM steht für „Requirements Management“ (Verwalten der Anforderungen)
  - Steuerung, Kontrolle und Verwaltung von Anforderungen

# Requirements in the V-Model





### Beispiel:



- Scope (vorgegebenen Zweck) festlegen
  - Ziele erheben (der einzelnen Stakeholder)
  - Stakeholder und Quellen identifizieren
- Nichtfunktionale Anforderungen erheben (z.B. Performanz, Usability, Sicherheit, Zuverlässigkeit, rechtliche Anforderungen usw.)
- Funktionale Anforderungen erheben

**Frage: Welche Methoden könnten hier sinnvoll sein ?**

# Kriterien, die eine Anforderung erfüllen muss (ein Auszug)

- **vollständig**
  - alle Anforderungen des Kunden müssen explizit beschrieben sein, es darf keine impliziten Annahmen des Kunden über das zu entwickelnde System geben.
- **eindeutig definiert / abgegrenzt**
  - präzise Definitionen helfen, Missverständnisse zwischen Entwickler und Auftraggeber zu vermeiden.
- **verständlich**
  - damit sowohl der Auftraggeber als auch der Entwickler mit vertretbarem Aufwand die gesamte Anforderung lesen und verstehen können.
- **atomar**
  - es darf nur eine Anforderung pro Abschnitt oder Satz beschrieben sein. Das Kriterium für ein "Atom" sollte die Entscheidbarkeit einer Anforderung sein.
- **identifizierbar**
  - jede Anforderung muss eindeutig identifizierbar sein (z. B. über eine Kennung oder Nummer).
- **nachprüfbar**
  - die Anforderungen sollten mit Abnahmekriterien verknüpft werden, damit bei der **Abnahme** geprüft werden kann, ob die Anforderungen erfüllt wurden (Ableitung von Testfällen aus den Abnahmekriterien).

# Motive zur Verwendung natürlicher Sprache



- Naheliegend
  - Ausdrucksmächtig
  - Ohne zusätzliche Ausbildung les- und schreibbar
  - Heute das am häufigsten verwendete Mittel zur Formulierung von Anforderungen
- Probleme:
  - unübersichtlich
  - fehlerträchtig
  - schwierig zu prüfen

- Die Qualität natürlichsprachige Anforderungsspezifikation lässt sich systematisch verbessern durch
  - geeignete Strukturierung des Dokuments
    - Einzelanforderungen in kleinen Einheiten fassen: eine Kernaussage pro Einzelanforderung, jede Einzelanforderung als einen Satz formulieren
    - Einzelanforderungen strukturieren (zum Beispiel durch Ordnen in Kapiteln und Abschnitten)
  - Regeln zur sprachlichen Formulierung von Anforderungen
  - kontrollierten Umgang mit Redundanz
  - Vermeidung von Weak-Words (schneller, schöner, automatisch, circa,....)
  - konsequente Verwendung eines Glossars

- Daten modellieren
- Machbarkeit überprüfen
- Nutzungsmodell erstellen
- Interaktionsmodell erstellen
- 
- Anforderungsauswirkung analysieren

- Verfolgbarkeit sicherstellen
  - Ziel ist es, die Verfolgbarkeit von Anforderungen sicherzustellen, um deren Verfeinerung / Realisierung in späteren Entwicklungsprodukten nachvollziehen zu können.
- Kundenanforderungen dokumentieren
  - Ziel ist es, die Anforderungen der Kunden zu dokumentieren.
- Anforderungen messbar und testbar beschreiben
  - Ziel ist es, Anforderungen überprüfbar zu spezifizieren.
- Entwickleranforderungen dokumentieren
  - Ziel ist es, die Anforderungen aus Entwicklersicht zu dokumentieren, um so eine optimale Kommunikation mit den Entwicklern bezüglich der Anforderungen zu ermöglichen.
- Standards und Dokumentenstruktur benutzen
  - Ziel ist es, durch Verwendung einer Dokumentenstruktur und Erfüllung von Dokumentationsstandards die Qualität und insbesondere die Verständlichkeit und Vollständigkeit eines Anforderungsdokuments zu erhöhen

- Anforderungen formal überprüfen
  - Ziel ist es, Anforderungen mit Hilfe formaler, mathematischer Techniken zu überprüfen, um dadurch insbesondere Konsistenz und Vollständigkeit sicherstellen zu können.
- Prototyping
  - Ziel ist es, ein späteres System oder Teile eines späteren Systems (z.B. Benutzerschnittstelle) exemplarisch zu implementieren, um hiermit frühzeitig Qualitätseigenschaften überprüfen und die Kommunikation mit den Stakeholdern unterstützen zu können.
- Usability überprüfen
  - Ziel ist es, die Gebrauchstauglichkeit eines Systems sicherzustellen.
- Anforderungen reviewen
  - Ziel ist es, die Anforderungen systematisch zu inspizieren, um die Qualität eines Systems bereits in frühen Phasen der Softwareentwicklung sicherzustellen.
- Tests zu Anforderungen vorbereiten
  - Ziel ist es, frühzeitig Testfälle für die Testphase vorzubereiten, um damit Feedback zur Überprüfbarkeit der Anforderungen zu erhalten.

- **Project Drivers**

1. The Purpose of the Project
2. The Client, the Customer, and Other Stakeholders
3. Users of the Product

## **Project Constraints**

4. Mandated Constraints
5. Naming Conventions and Definitions
6. Relevant Facts and Assumptions

## **Functional Requirements**

7. The Scope of the Work
8. The Scope of the Product
9. Functional and Data Requirements

## **Nonfunctional Requirements**

10. Look and Feel Requirements
11. Usability and Humanity Requirements
12. Performance Requirements
13. Operational and Environmental Requirements
14. Maintainability and Support Requirements
15. Security Requirements
16. Cultural and Political Requirements
17. Legal Requirements

## **Project Issues**

18. Open Issues
19. Off-the-Shelf Solutions
20. New Problems
21. Tasks
22. Migration to the New Product
23. Risks
24. Costs
25. User Documentation and Training
26. Waiting Room
27. Ideas for Solutions

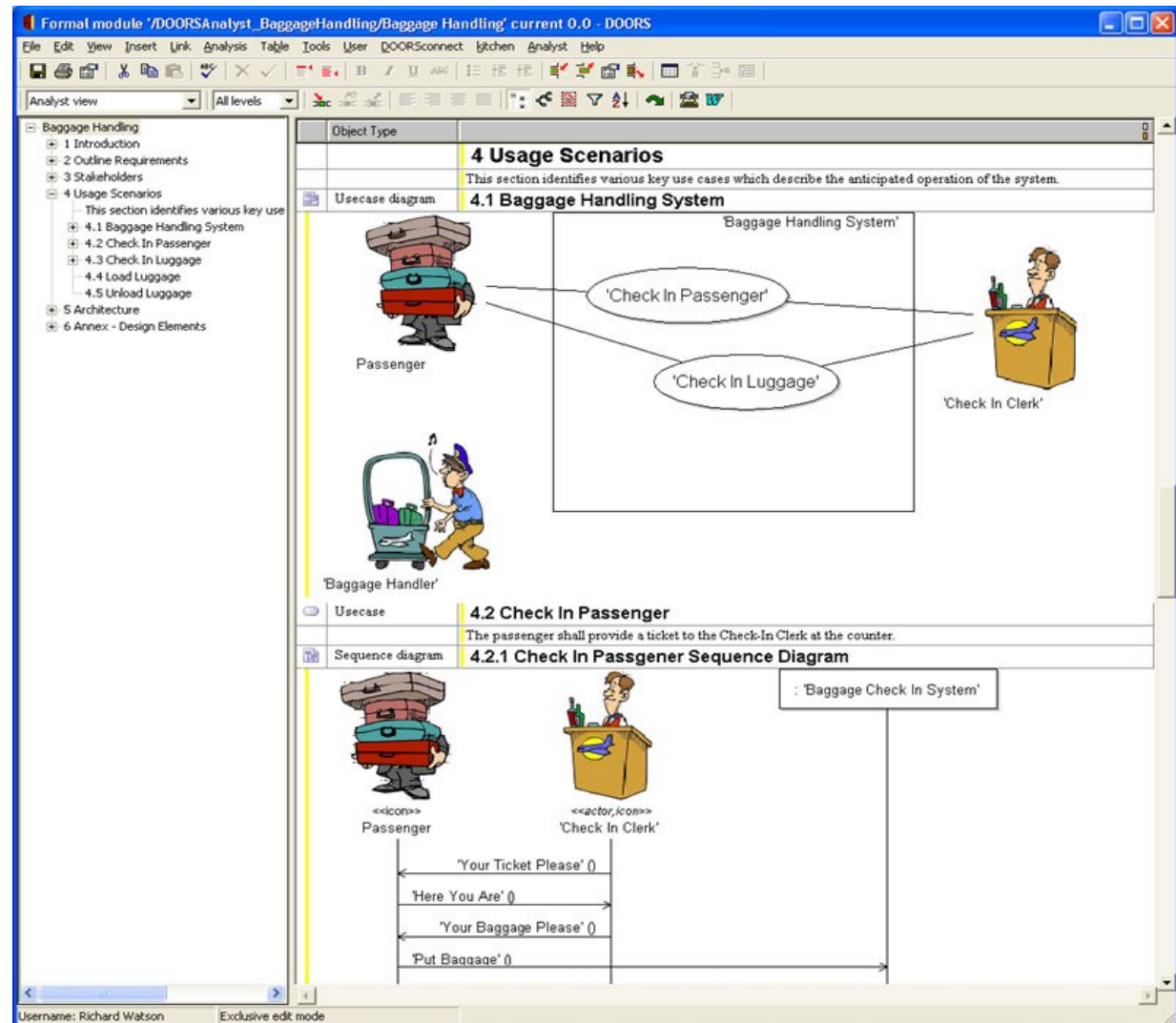
# Welche Vorteile habe ich durch RE/RM?

- Schaffung eines gemeinsamen Verständnis über ein zu entwickelndes System zwischen Auftragnehmer und Auftraggeber
- Qualitätssteigerung durch genauere Testspezifikationen
- Kostenreduktion
- Entwicklungszeit-Verkürzung
- Steigerung der Planungssicherheit
- Bessere Kommunikation der Projekt-Teams
- effiziente und fehlerarme Entwicklung
  
- Erfolgreiche Projekte ;-)
- Zitat Uni-Essen (<http://sse.informatik.uni-essen.de/>)

*„Requirements Engineering betrachten wir als Prozess, dessen Ziel es ist, eine Brücke zwischen Anwendern, Nutzern, Geldgebern, etc. und Entwicklern von Softwaresystemen aufzubauen und aufrecht zu erhalten. Requirements Engineering ist somit ein kontinuierlicher, kreativer Prozess, der immer wieder aus vagen und widersprüchlichen Ideen hinreichend vollständige und konsistente Systemspezifikationen unter Berücksichtigung vorhandener technischer und sozialer Kontexte gewinnen muss“*

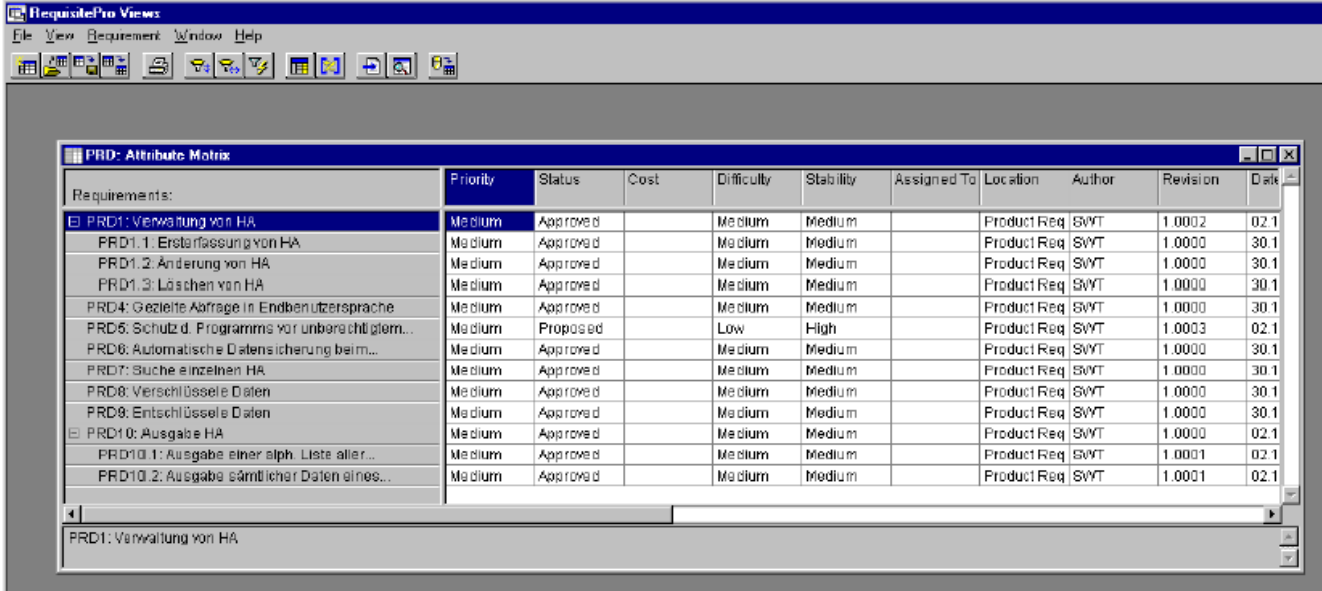
# Welche Werkzeuge können hier unterstützen?

- Telelogic DOORS Markführer im Bereich des RM. Mittlerweile von IBM gekauft.



# Welche Werkzeuge können hier unterstützen?

- Requisite Pro (IBM)

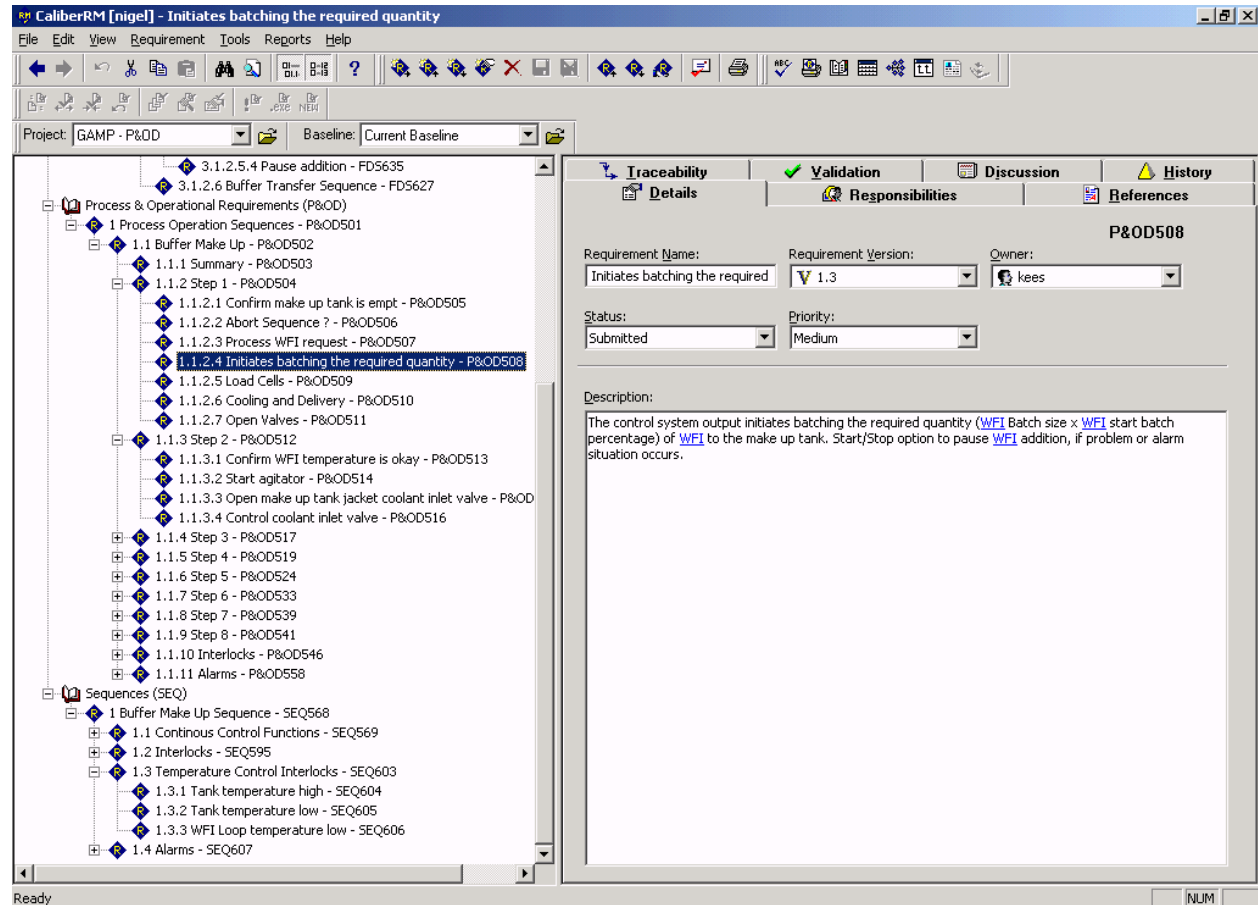


The screenshot shows the 'Requisite Pro Views' application window. The main content is a table titled 'PRD: Attribute Matrix'. The table has columns for Requirements, Priority, Status, Cost, Difficulty, Stability, Assigned To, Location, Author, Revision, and Date. The requirements listed include PRD1: Verwaltung von HA and its sub-requirements, PRD4: Gezielte Abfrage in Endbenutzersprache, PRD5: Schutz d. Programms vor unberechtigtem..., PRD6: Automatische Datensicherung beim..., PRD7: Suche einzelnen HA, PRD8: Verschlüsselt Daten, PRD9: Entschlüsselt Daten, PRD10: Ausgabe HA, and PRD10 sub-requirements.

Requirements:	Priority	Status	Cost	Difficulty	Stability	Assigned To	Location	Author	Revision	Date
PRD1: Verwaltung von HA	Medium	Approved		Medium	Medium		Product Req	SVYT	1.0002	02.1
PRD1.1: Ersterfassung von HA	Medium	Approved		Medium	Medium		Product Req	SVYT	1.0000	30.1
PRD1.2: Änderung von HA	Medium	Approved		Medium	Medium		Product Req	SVYT	1.0000	30.1
PRD1.3: Löschen von HA	Medium	Approved		Medium	Medium		Product Req	SVYT	1.0000	30.1
PRD4: Gezielte Abfrage in Endbenutzersprache	Medium	Approved		Medium	Medium		Product Req	SVYT	1.0000	30.1
PRD5: Schutz d. Programms vor unberechtigtem...	Medium	Proposed		Low	High		Product Req	SVYT	1.0003	02.1
PRD6: Automatische Datensicherung beim...	Medium	Approved		Medium	Medium		Product Req	SVYT	1.0000	30.1
PRD7: Suche einzelnen HA	Medium	Approved		Medium	Medium		Product Req	SVYT	1.0000	30.1
PRD8: Verschlüsselt Daten	Medium	Approved		Medium	Medium		Product Req	SVYT	1.0000	30.1
PRD9: Entschlüsselt Daten	Medium	Approved		Medium	Medium		Product Req	SVYT	1.0000	30.1
PRD10: Ausgabe HA	Medium	Approved		Medium	Medium		Product Req	SVYT	1.0000	02.1
PRD10.1: Ausgabe einer alph. Liste aller...	Medium	Approved		Medium	Medium		Product Req	SVYT	1.0001	02.1
PRD10.2: Ausgabe sämtlicher Daten eines...	Medium	Approved		Medium	Medium		Product Req	SVYT	1.0001	02.1

# Welche Werkzeuge können hier unterstützen?

- CaliberRM (Borland)



# Weitere Tools

- Integral Requisite Analyzer (IRqA®)
- RTM (Serena)
- CARE (Sophist Group)
- Reqtify (TNI)

- Community Edition von "Trend/Analyst"
  - Mit TREND/Analyst hat GEBIT Solutions ein Requirement-Werkzeug der nächsten Generation entwickelt. Es richtet sich gleichermaßen an Business-Analysten/Requirements Engineers als auch an Software-Ingenieure, die ein integriertes Werkzeug für die Erfassung, Verwaltung und Umsetzung von Anforderungen suchen.
- Serena Prototype Composer
  - „Mit dem Prototype Composer hat Serena Software ein neues, kostenloses Virtualisierungs-Tool vorgestellt. Hiermit können Business-Analysten Aussehen und Funktion geplanter Anwendungen vorab simulieren, auch wenn in einem frühen Projektstadium noch keine einzige Zeile Programmcode fertiggestellt ist. Davon profitieren nicht nur Business-Analysten und Programmierer. Auch Endanwender können mit dem Prototyp interagieren wie mit der späteren Applikation und ihre Anregungen so frühestmöglich in den Entwicklungsprozess einfließen lassen. Damit überwindet der Serena Prototype Composer die schon fast sprichwörtlichen Abstimmungsschwierigkeiten zwischen Anwendern und Programmierern. Er bietet eine zeitgemäße Alternative zu langen Meetings, endlosen Spezifikationen und riesigen Word-Dokumenten, die früher die Zusammenarbeit zwischen den Abteilungen eher erschwert als unterstützt haben.“

- Universität Zürich, Anforderungsspezifikation mit natürlicher Sprache

<http://www.ifi.unizh.ch/req/fileadmin/downloads/teaching/course>

- Systematisches Requirements Management (Christof Ebert)

- Anforderungen ermitteln, spezifizieren, analysieren und verfolgen

- Wikipedia

- <http://www.re-wissen.de>

- <http://www.informatik.hs-mannheim.de/~knauber/MSc-MSI-06/06->

- <http://www.telelogic.de>