

# Internet Briefing Agile SW-Entwicklung



6. Februar 2007  
Peter Stevens, Principal Consultant

# Agenda



# Agenda

- » Kurzportrait namics
- » Die Herausforderung des Projektmanagements
  - Was kann schief gehen?
  - Das Beispielprojekt „Schaukel“
- » Was ist „agile“ Software-Entwicklung?
- » Wie arbeitet man agil?
  - Das Beispielprojekt „Amsterdam“
- » Erfahrungen beim ersten Grossprojekt: Publicconnect (WLC)

# Kurzportrait namics.



## Eckdaten zu namics.

- » Marktführender Schweizer Internet-Dienstleister, starke Präsenz in Deutschland
- » Fokus
  - Konzeption und Implementierung nutzergerechter, effizienter und begeisternder Internet-Anwendungen
  - Methodisch fundiertes Vorgehen, pragmatische Umsetzung
- » Zahlen und Fakten
  - gegründet 1995 als Spinoff der Universität St. Gallen
  - 180 Mitarbeiter, Umsatz 2005 CHF 20,4 Mio.
  - Standorte: Bern, Frankfurt, Hamburg, München, St. Gallen, Zug, Zürich
  - Mehrheitsaktionär PubliGroupe SA, Minderheit beim Management (13 Partner)

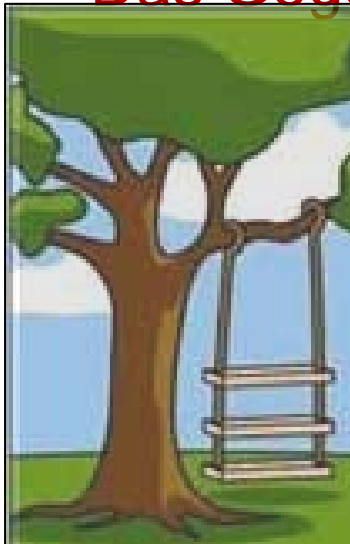
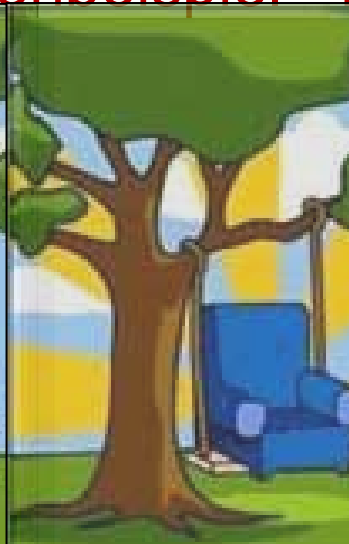
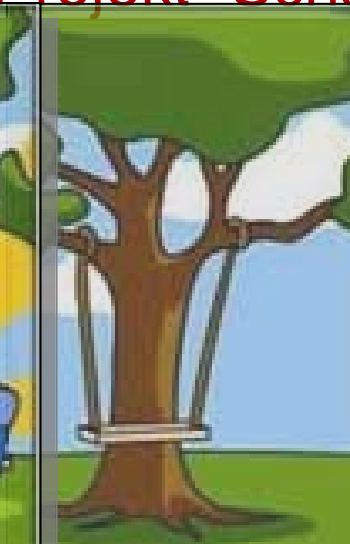

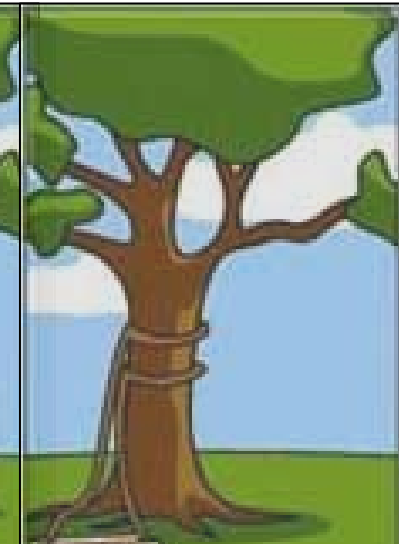
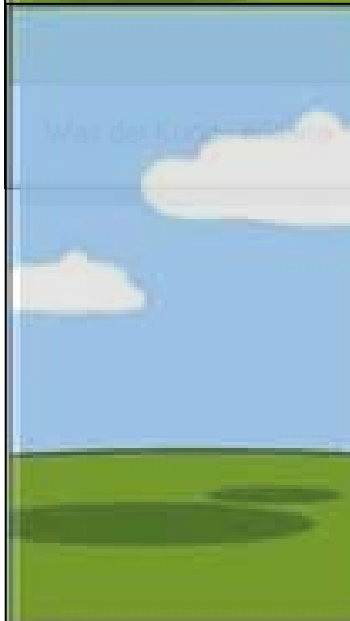
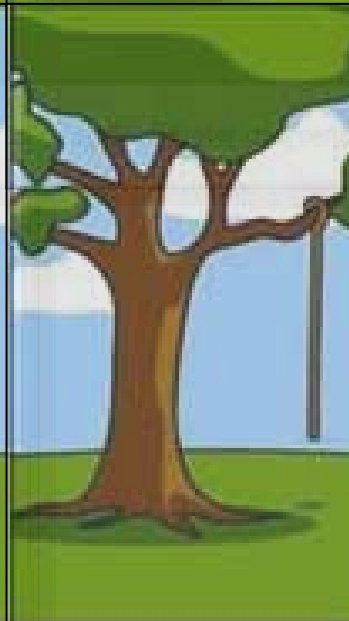

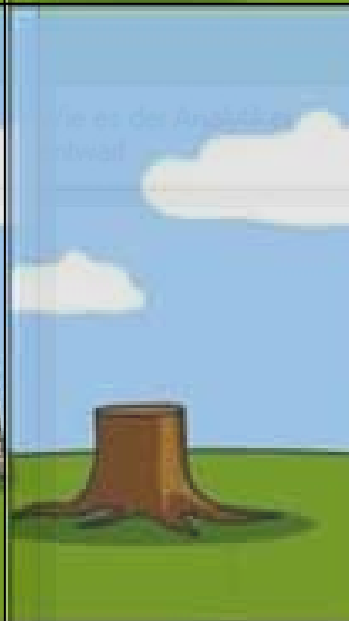
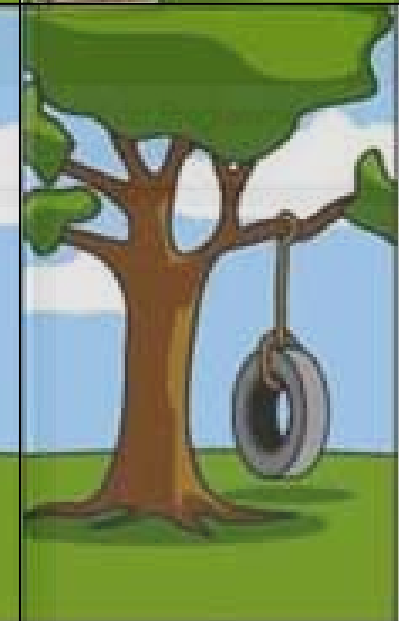
# Projekt-Management



# Die tägliche Herausforderung

- » 50% aller Projekte haben Probleme
  - Budget
  - Termine
  - Funktionalität
  - Qualität
- » Bei erfolgreichen Projekten
  - 66% der Funktionen werden von den Usern nie benutzt
- » Wenn es schief geht...

# Das Gegenbeispiel - Projekt "Schaukel"

				
Was der Kunde wollte	Was installiert wurde	Was dem Kunden in Rechnung gestellt wurde	Wie es gewartet wurde	Was der Kunde wirklich gebraucht hätte
				
Wie das Projekt definiert wurde	Was installiert wurde	Was dem Kunden in Rechnung gestellt wurde	Wie es gewartet wurde	Was der Kunde wirklich gebraucht hätte

# Was ist agile Software-Entwicklung?

## Das agile Manifesto

Menschen und Interaktion

Prozesse und Werkzeuge

Funktionierende Software

Ausführliche Dokumentation

Zusammenarbeit mit dem Kunden

Vertragsverhandlungen

Auf Änderungen reagieren

Einen Plan befolgen

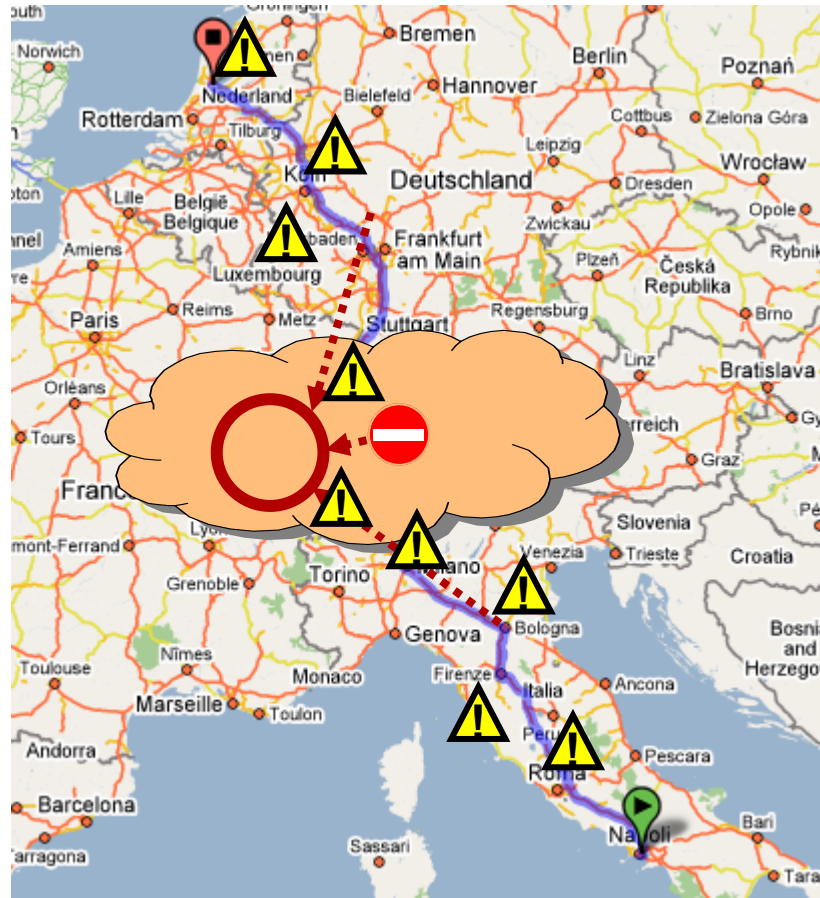


# Agile Methodologien

- » Scrum ← 
- » XP (eXtreme Programming)
  - Communication, Simplicity, Feedback, Courage, Respekt
  - TDD – Test Driven Development
  - Pair Programming
- » Others
  - DSDM
  - Crystal

# Wie fährt man von Neapel nach Amsterdam? – klassisch

- » Händler kauft 20t Kaffee in Neapel
- » Er will die Ware nach Amsterdam bringen, um sie zu verkaufen



Risiken:

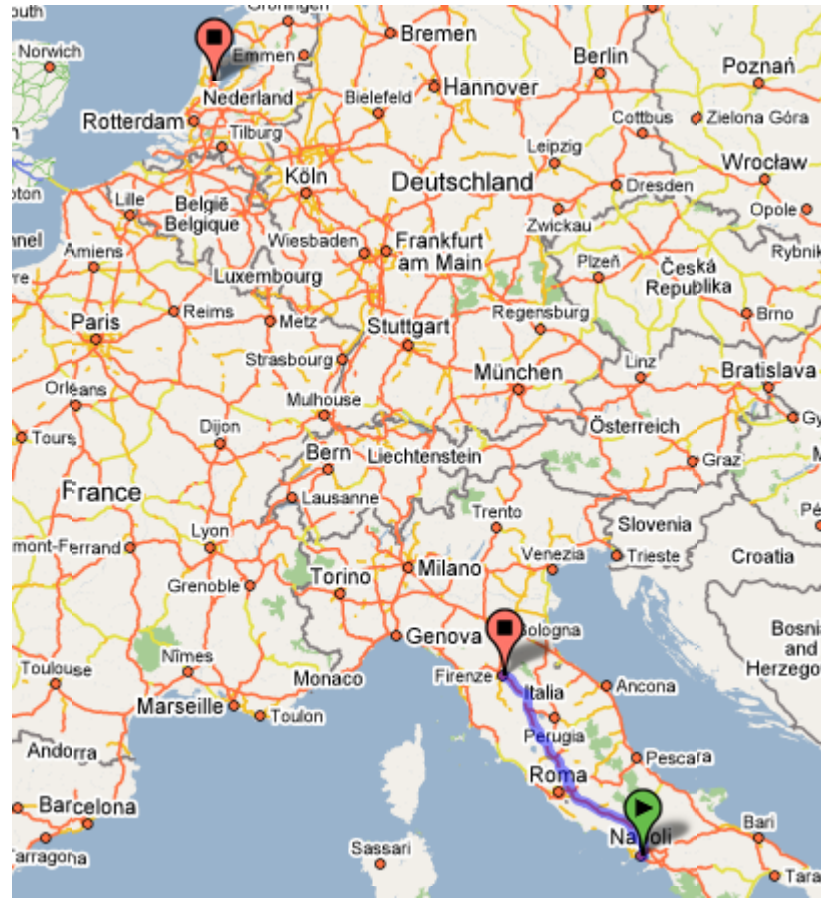
Verkehr 

Wetter 

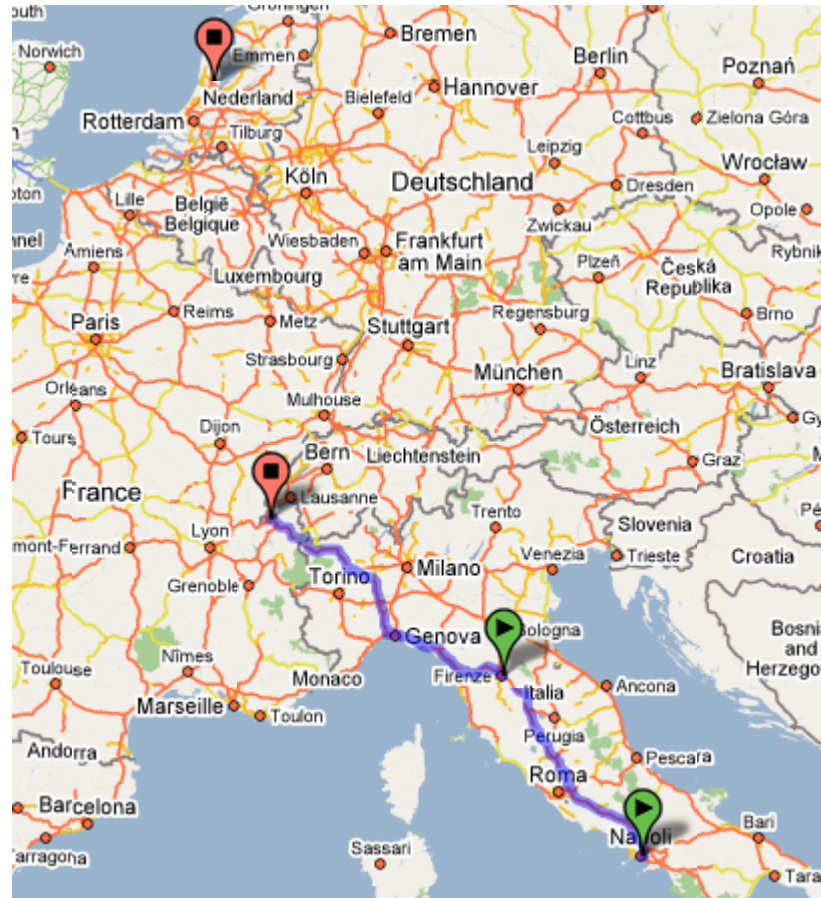
Sperrungen 

Kunde 

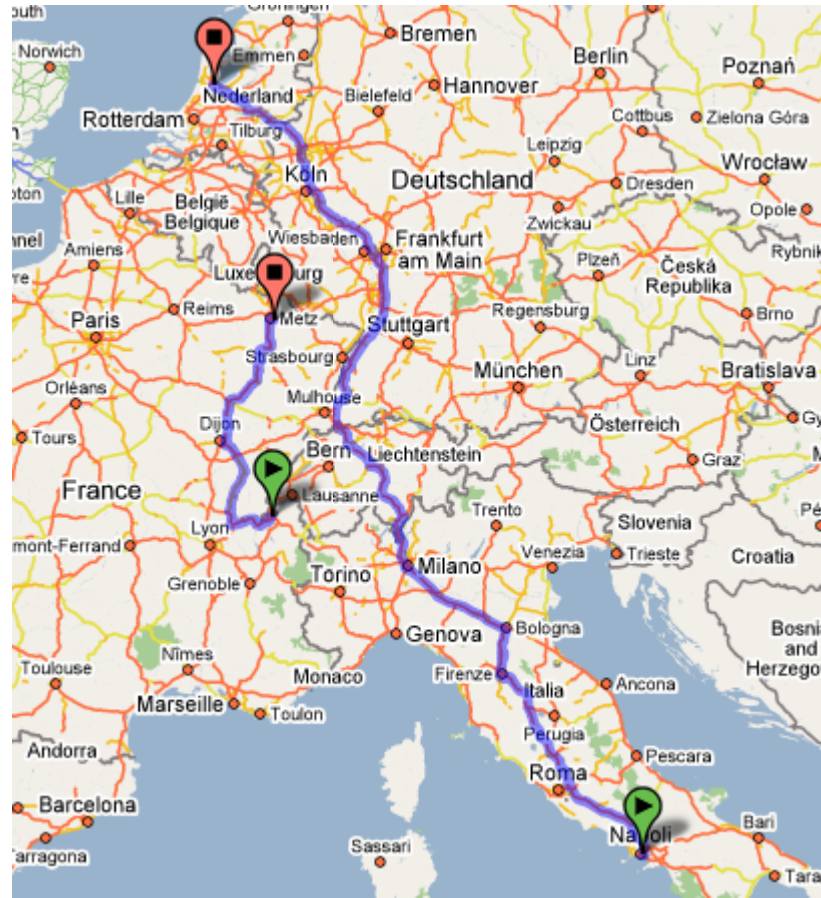
# Wie fährt man von Neapel nach Amsterdam? – nach Scrum



# Wie fährt man von Neapel nach Amsterdam? – nach Scrum



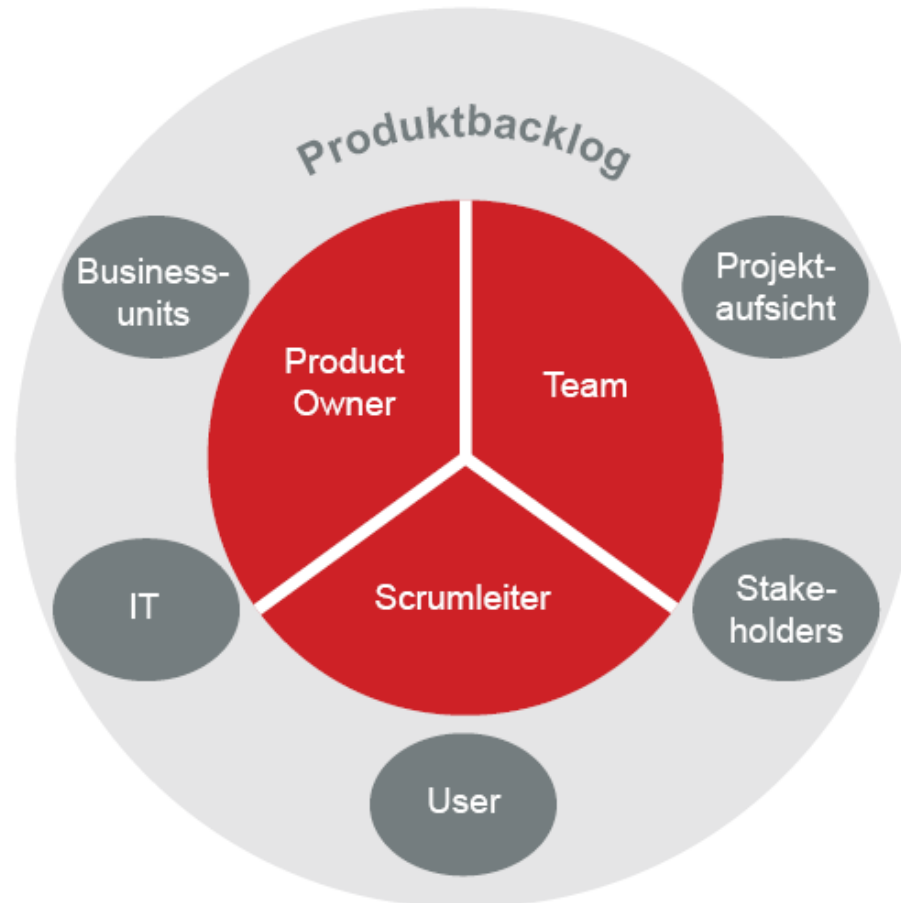
# Wie fährt man von Neapel nach Amsterdam? – nach Scrum





# Was ist Scrum?

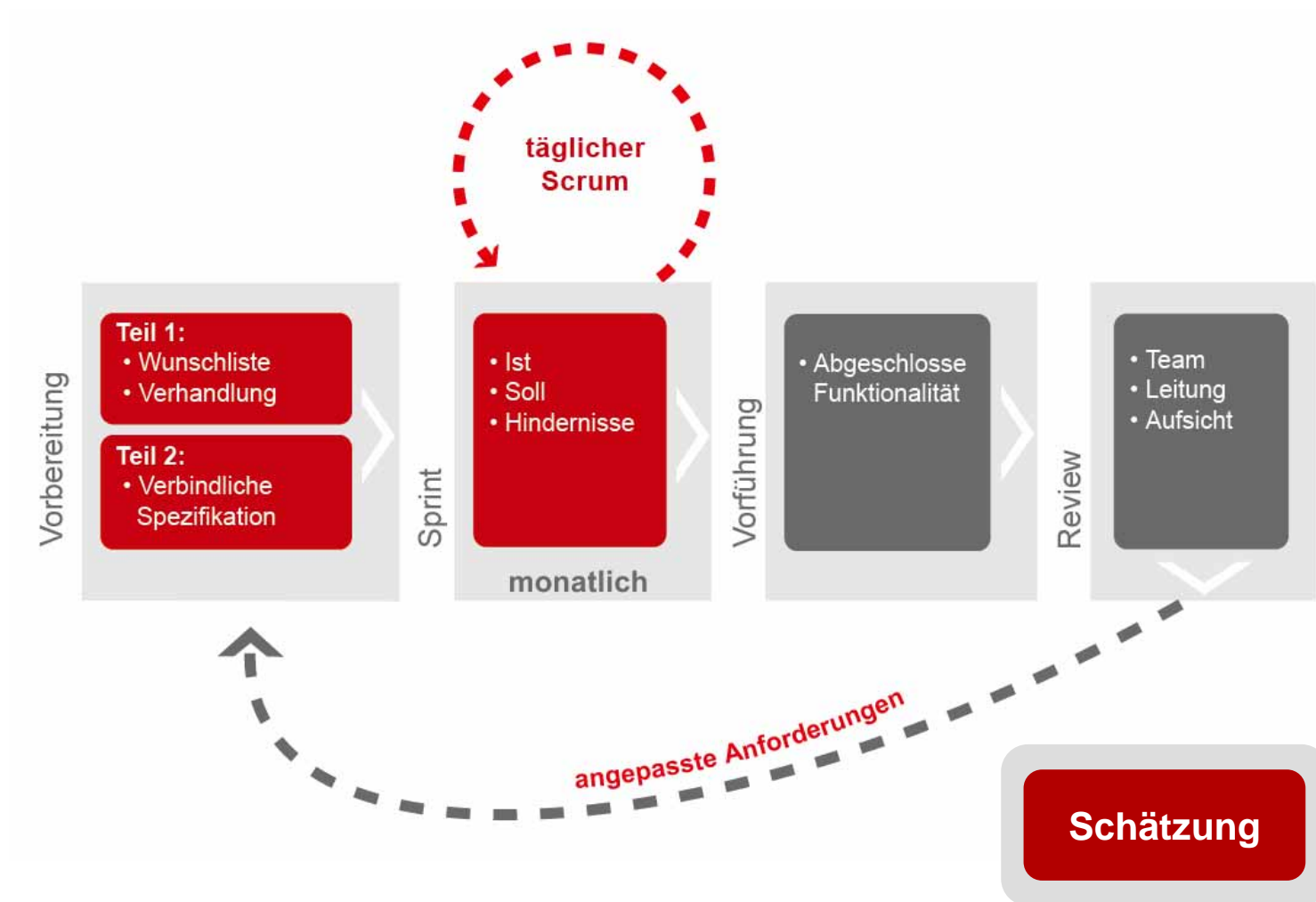
- » Scrum ist
  - „just an interaction model for teams and stakeholders“
  - eine PL-Methodik, welche stark auf das Erreichen von geschäftsrelevanten Zielen fokussiert ist (ROI)
  - eine Sammlung von einfachen Arbeitstechniken, Strukturen, Rollen und Methoden für das Projektmanagement im Rahmen agiler Software-Entwicklung
  
- » Eine Entwicklungsperiode (Sprint) wird auf 30 Tage terminiert
  
- » Jede Aufgabe wird im Team gelöst
  
- » Am Ende des Sprints: Die entwickelten Funktionen werden präsentiert - echte Funktionalität, keine Screen-Shots
  
- » Nach der Präsentation: Feedback einholen, Ideen und Vorschläge für die nächsten 30 Tage sammeln

# Rollen



-  Schweine (verpflichtet)
-  Hühner (involviert)

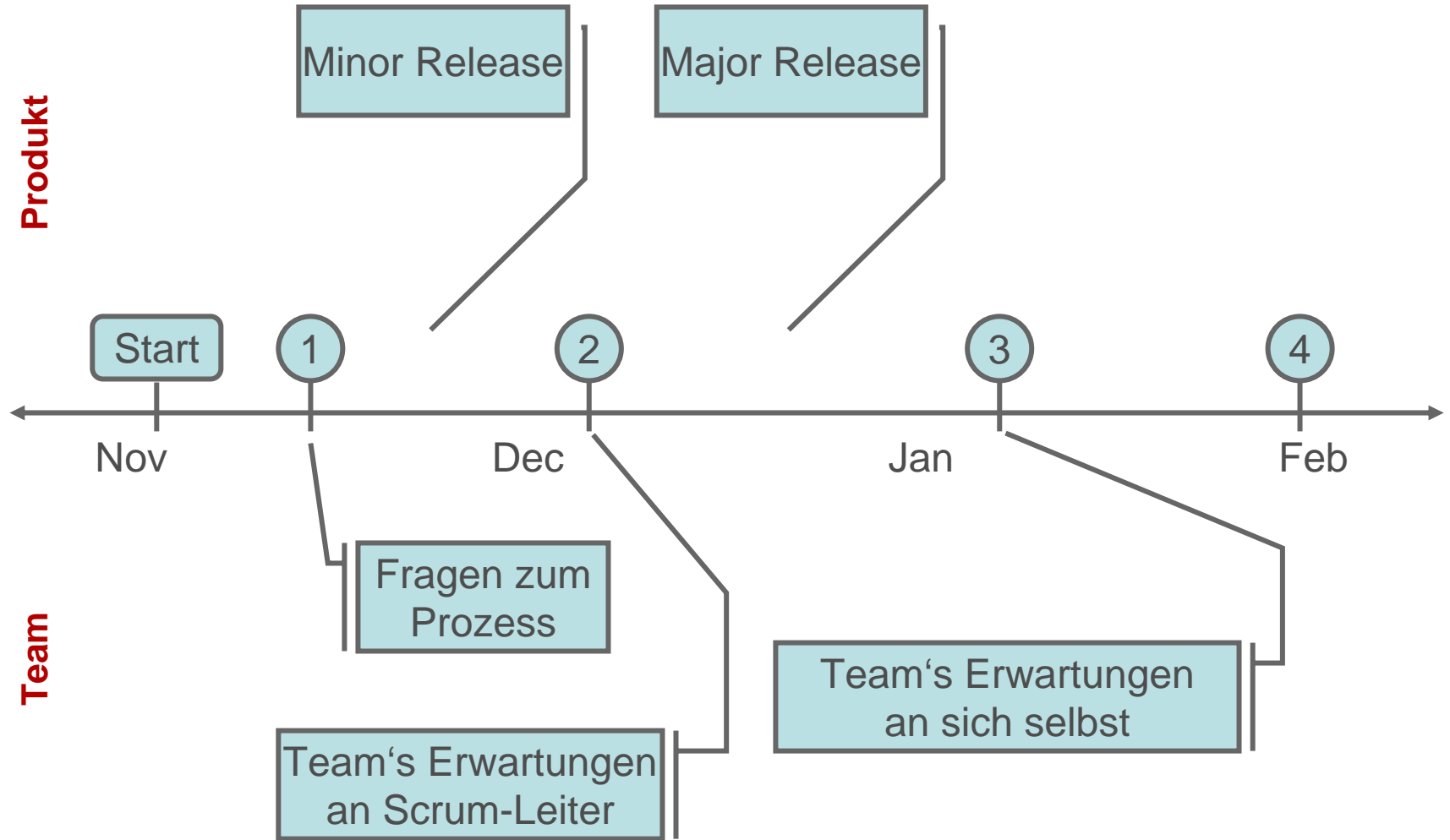
# Sitzungen



# Einführung von Scrum beim WLC-Projekt

- » Das Projekt
  - läuft seit 1½ Jahren
  - ASP-Lösung für Kleinanzeigen
  - NZZExecutive.ch
- » Kunden und Team
  - überzeugen und einarbeiten
  - Was heisst „Done“?
  - Team sagt zu, mit Scrum zu arbeiten
  - Wie arbeitet man unter Scrum?
  - Einheitliche Bug- & Feature-Liste erstellen
  - Aufwand im Team schätzen (Grösse/Komplexität, keine AT)
  - Aufgaben von Kunden priorisieren
- » Organisation
  - Kunde stellte Produkt-Owner
  - namics stellte Scrummaster und Team
  - Team mehrheitlich 100%
- » Planung vom ersten Sprint
  - Zielsetzung des Kunden
  - Dauer
  - Ergebnisse
  - Was heisst release-bar?
  - Mini-Wasserfälle

# Einführung von Scrum beim WLC-Projekt



# Scrum aus Team-Sicht

## Vorteile

- » Kommunikation mit dem Kunden
  - Fokus auf brauchbare Lösungen
  - Ziel immer vor Augen
  - Verantwortung kann nicht abgeschoben werden
- » Der Kunde entscheidet „was“
  - ist auch ein „Schwein“
  - hat absolute Kostentransparenz
  - kann jederzeit das Ergebnis steuern.
- » Das Team entscheidet „wie“
  - genießt offene Interaktion
  - spezifiziert im Gespräch mit dem Kunden
  - baut das, was der Kunde möchte
- » Früh erkennbar, ob Ziele realistisch sind

## Nachteile

- » Nicht jeder Entwickler möchte mit dem Kunden diskutieren
- » Sprint-Overhead (Meeting-Marathons)
- » Bereitschaft Verantwortung zu übernehmen
- » Respekt vor Refactoring

## Weitere Hinweise

- » Trotz Arbeiten im Takt sicherstellen, dass das Fernziel des Kunden noch erreicht werden kann.

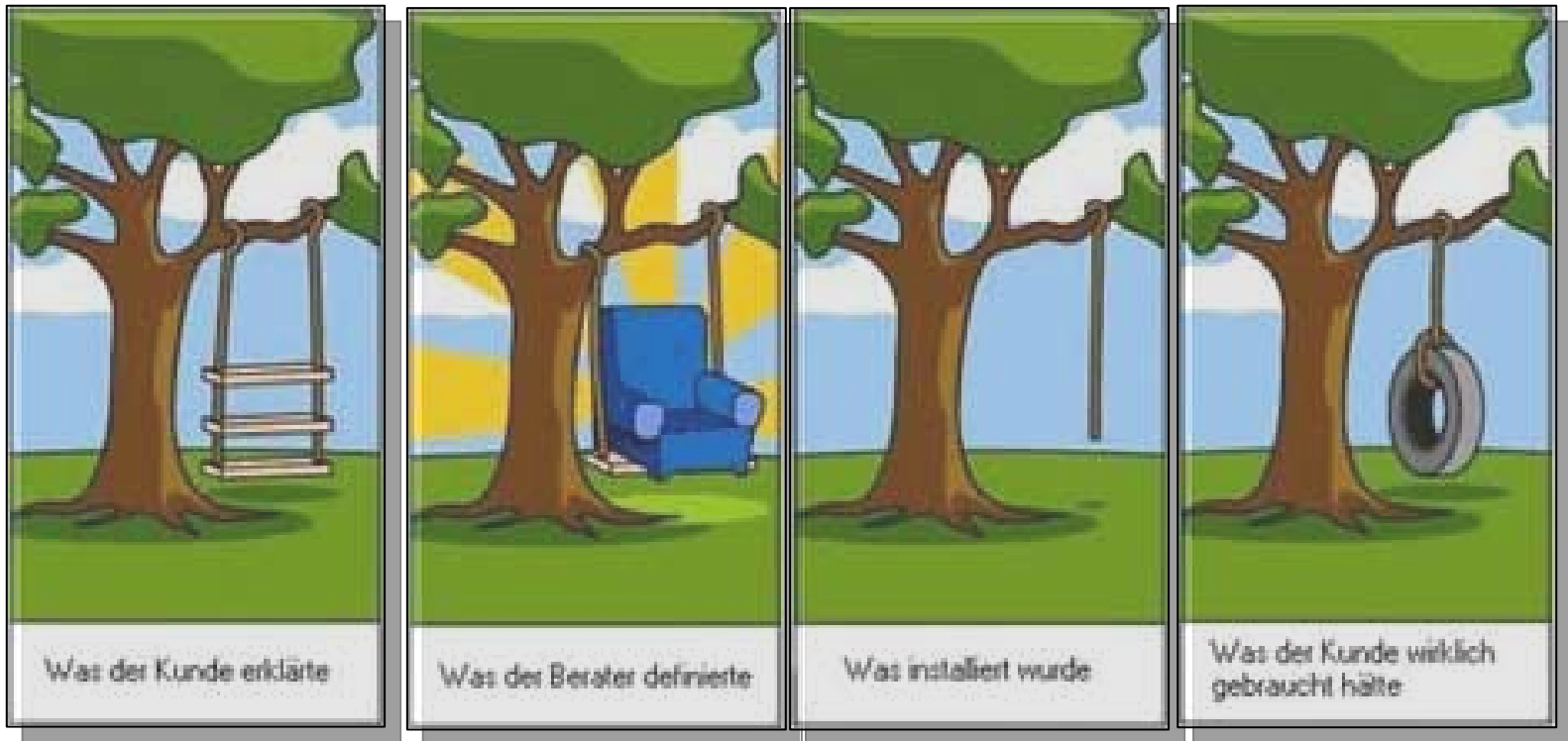
## Kundensicht - Nachteile

- » Verzicht auf kurzfristige Flexibilität
- » Business-Plan & Budget
  - Risiko mehr beim Kunden (vorher sagt man mehr als man eigentlich sagen dürfte)
  - Budgetieren von grösseren Vorhaben ist schwieriger - es fallen mehr Kosten an, bis man weiss, was es wirklich kostet
  - Nicht Finanz-Abteilungs-konform
- » Product-Owner >> „Auftraggeber“
- » Anfangsskepsis / Ungewissheit / interne Akzeptanz
  - Bezahlen wir, um Versuchskaninchen zu spielen?

## Kundensicht - Vorteile

- » Diskrepanz zwischen Wunsch und Realität wird am Anfang bereinigt
  - nach Sprint-Verhandlung kann jeder „in Ruhe“ arbeiten
  - geliefert wird, was vereinbart wird
  - richtige Erwartungen
- » Resultat-Orientierung
  - Resultate nach jedem Sprint
  - Leistungen des Teams bzw. des Lieferanten sind klar
- » Kommunikation
  - Team versteht Kunden besser
  - Zusammenarbeit im Team
  - mitdenken
- » Qualität hat sich verbessert
  - Termin-Treue
  - Kosten-Treue
- » Transparenz & Kontrolle
  - steigert Vertrauen in Lieferanten
  - Reporting an Management & Kunden
- » Fahrtakt – wie ein Zug
  - eine Krise dauert höchstens einen Sprint lang
- » Klare Methodik
  - „berechenbares Verhalten“ des Lieferanten
  - Konsistenter, jedoch ertragbarer Druck
  - keine Krisen

# Das Beispiel-Projekt “Schaukel” unter Scrum



**Sprint 1**

**Sprint 2**

# Mehr Informationen



## Mehr Informationen

- » [www.namics.com/letstalk?about=Scrum](http://www.namics.com/letstalk?about=Scrum)
- » [blog.namics.com](http://blog.namics.com)
- » Scrum-Kurs bei namics:
  - 14./15.5.07 Certified Scrum Master Training
  - 16.5.06 Certified Scrum Leader Training
  - [www.namics.com](http://www.namics.com)

Besten Dank für Ihre Aufmerksamkeit.



[peter.stevens@namics.com](mailto:peter.stevens@namics.com)