

Verteidigung der Bachelorarbeit:

# “Formalisierung von Requirements durch Nutzung von Templates”

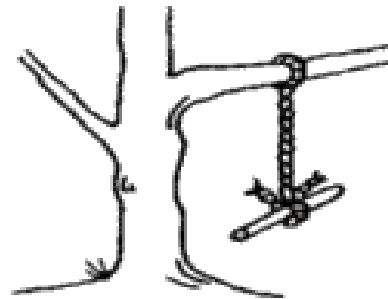
von Christian Kühl



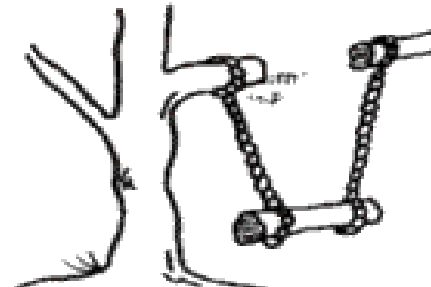
10.02.2011

# Motivation

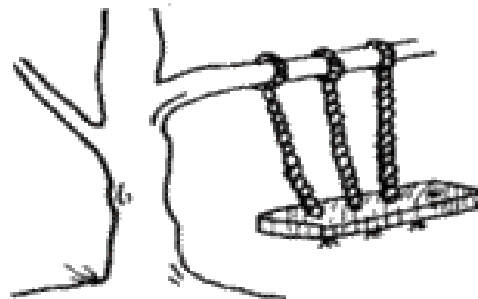
- Warum Anforderungen verbessern?



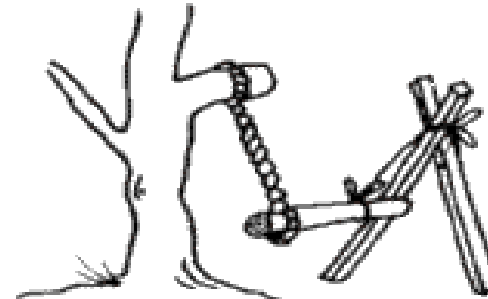
**What the user asked for**



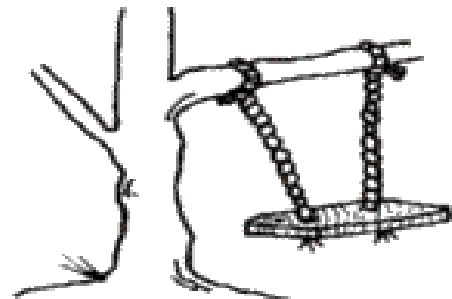
**How the analyst saw it**



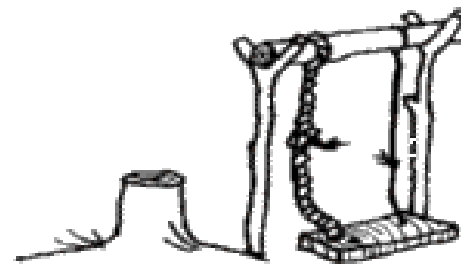
**How the system was designed**



**As the programmer wrote it**



**What the user really wanted**



**How it actually works**

# Gliederung

- Anforderungen
  - Was ist das?
  - Qualitätskriterien nach ISO 26262
  - Beispielanforderung
- Templates
  - Was sind Templates?
  - Vorhandene Methoden
  - Verwendete Methodik
  - Anwendung der Methodik an einem Beispiel



# Anforderung: Was ist das?

Eine Anforderung ist:

Eine Bedingung oder Fähigkeit, die ein System oder Teilsystem erfüllen muss, beziehungsweise von einem Benutzer (Person oder System) zur Lösung eines Problems oder Erreichung eines Ziels benötigt wird.

Quelle: Klaus Pohl & Chris Rupp



# Qualitätskriterien nach ISO 26262

- eindeutig
- verständlich
- atomar
- intern konsistent
- prüfbar
- machbar



# Qualitätskriterien Fortsetzung

- hierarchisch strukturiert
- organisatorisch strukturiert gemäß eines Schemas
- vollständig
- extern konsistent
- nicht redundant
- wartbar



# Ziel

- Die Verwendung von Templates soll folgende Qualitätskriterien positiv beeinflussen
  - Atomar
  - Verständlich
  - Prüfbar
  - Eindeutig



# Beispiel Anforderung

Das System soll das Licht anschalten.

- Was wissen wir:
  - Irgendein System soll irgendein Licht anschalten
- Noch unklar:
  - Was für ein System ist es
  - Um was für ein Licht handelt es sich
  - Gibt es Bedingungen, wann das Licht aktiviert werden soll?
  - ...





# Beispielanforderung

- Besser:
  - Solange eine Tür des Autos geöffnet ist und der Lichtschalter für die Innenbeleuchtung auf automatisch steht, soll die Software die Innenbeleuchtung anschalten.



10.02.2011

# Gliederung

- Anforderungen
  - Was ist das?
  - Qualitätskriterien nach ISO 26262
  - Beispielanforderung
- **Templates**
  - Was sind Templates
  - Vorhandene Methoden
  - Verwendete Methodik
  - Anwendung der Methodik an einem Beispiel



# Template (Schablone)

- Allgemein:
  - Eine Schablone ist ein ausgeschnittenes Muster zur Herstellung gleichgestaltiger Dinge.
- Anforderungsschablone:
  - Eine Anforderungsschablone (Requirements Template), auch Satzschablone genannt, ist ein Bauplan für die syntaktische Struktur einer einzelnen Anforderung.

Quelle: Klaus Pohl & Chris Rupp

# Methoden

- Formaler Ansatz
- Reviewbehafteter Ansatz
- “Mittelweg”



# Formaler Ansatz

- Mathematik basierender Ansatz zur Spezifikation, Entwicklung und Verifikation von Software- und Hardwaresystemen
- Vorteile:
  - Durch hohe Disziplin schnell Fehler finden
  - Systeme und Anforderungen präziser entwickelt
  - Eindeutige, verständliche Anforderungen für Kenner
- Nachteile:
  - Großer Zeitaufwand
  - Unverständlich für alle anderen
- Nutzen für die Entwicklung von kritischen Systemen
- Beispielprogramm: Estrel:

```
module ABRO:  
input A, B, R;  
output O;  
loop  
  [ await A || await B ];  
  emit O  
each R  
  
end module
```

# Reviewbehafteter Ansatz

- DESIRe
- Definierung kritischer Wörter (System, use, immadietly,...)
- Definierung von Fragen zu den Wörtern
- Nach Fertigstellung Überprüfung der Anforderungen
- Dadurch Validierung der Anforderung
- Aber verhindert nicht das Schreiben von schlechten Anforderungen



# Verwendete Methodik

- Einteilung in Kategorien von Anforderungen
  - Performance
  - Structure
  - Diagnosis
  - Operation Modes
  - ...
- Templates für den Satzbau einer Anforderung

# Aufbau eines Templates

- Eigene kleine Templatemetasprache entwickelt
- Besitzt folgende Bausteine:
  - Variablen
  - Optionale Parts
  - Listen



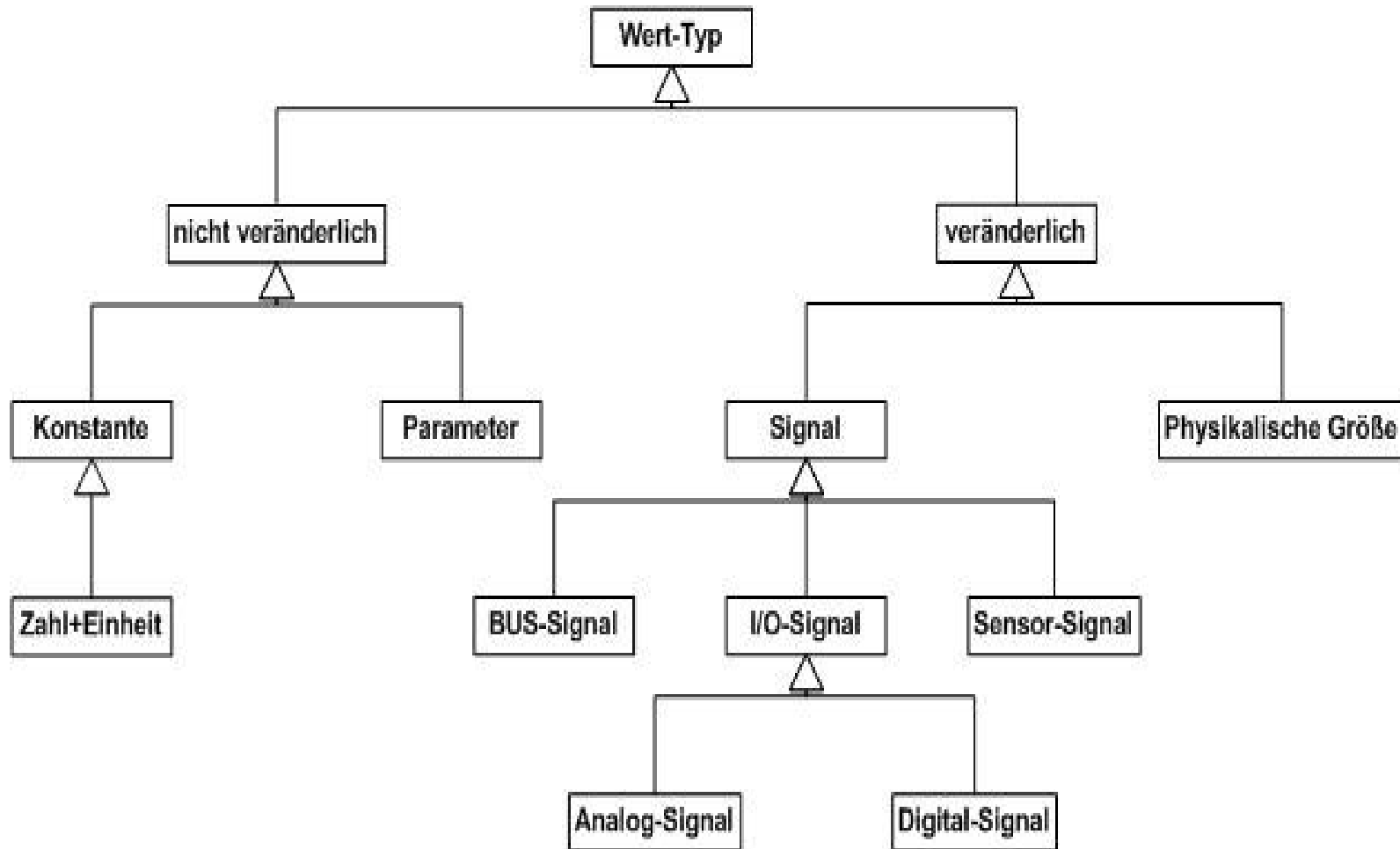


# Variablen

- Variablen werden durch [ ] dargestellt
- Eine Variable repräsentiert meist eine Klasse von weiteren Variablen
- Durch formale Regeln (BNF) Unterklassen definiert
- Auf tiefster Ebene Ersetzung durch realen Wert



# Beispiel Wert-Typ



# Optionale Parts und Listen

- Optionale Parts
  - werden durch { } repräsentiert
  - Notwendig, sonst Templateexplosion
- Listen
  - Durch [List of (Typ)] repräsentiert
  - Verwendung z.B. bei Reaktionen



# Beispiel

- ~~{For every incoming [message], the} {The} [component] shall check if [Condition] and, {in case of violation trigger the appropriate reaction as specified in the fault database} {perform the following reactions:[List of (Reactions)]}~~
  - [component] → Hardware
  - [Condition] → the temperature of the CPU is bigger than 80°C
  - [List of (Reactions)] → increase the engine output of the ventilator
- The Hardware shall check if the temperature of the CPU is bigger than 80°C and, perform the following reactions:
  - Increase the engine output of the ventilator

# Ausblick

- Automatische Testfallgenerierung
- Modellgenerierung
- Vollständigkeit von Anforderungen besser gewährleisten



# Vielen Dank

- Noch Fragen?



10.02.2011