Proseminar Spezifikationsformalismen

Vorbesprechung

Thomas Noll

Software Modeling and Verification Group

22. April 2008

Zielsetzung

Spezifikationsformalismen

Spezifikationssprachen = vereinfachte Programmiersprachen

- Beschreibung der "wesentlichen" Eigenschaften eines (Hard- oder Software-) Systems
- Abstraktion von Details
- Formale Semantik
- Anwendungen:
  - schrittweise Verfeinerung
  - Analyse und Verifikation

Zielsetzung

Spezifikationsformalismen

Spezifikationssprachen = vereinfachte Programmiersprachen

- Beschreibung der "wesentlichen" Eigenschaften eines (Hard- oder Software-) Systems
- Abstraktion von Details
- Formale Semantik
- Anwendungen:
  - schrittweise Verfeinerung
  - Analyse und Verifikation

Ziele des Proseminars

- Selbständiges Einarbeiten in ein neues Thema
- Literaturrecherche
- Darstellen des Inhalts in einer wissenschaftlichen Ausarbeitung
- Verständliches Präsentieren
- Teamarbeit (je 2 Studierende)

Anforderungen Ausarbeitung

Ausarbeitung

- selbständiges Verfassen einer ca. 15-seitigen Ausarbeitung
- vollständiges Literaturverzeichnis
- korrektes Zitieren
- Plagiarismus:
  Die nicht gekennzeichnete Übernahme fremder Inhalte führt zum sofortigen Ausschluß.
- Schriftgröße 11pt, übliche Seitenränder
- Sprache Deutsch oder Englisch
- Korrekte Sprache wird vorausgesetzt:
  $\geq$ 10 Fehler pro Seite $\Longrightarrow$ Abbruch der Korrektur

Anforderungen Vortrag

Vortrag

- 45-minütiger (abwechselnder) Vortrag
- Zielgruppengerechte Präsentation der Inhalte
- Übersichtliche Folien:
  - $\leq$ 15 Textzeilen
  - sinnvoller Einsatz von Farben
- Vortrag in Deutsch oder Englisch

Bibliothekseinführung

Einführung in die Literaturrecherche

Vorgesehene Termine:

- Montag, 28.04., 11:30 Uhr
- Dienstag, 29.04., 11:30 Uhr
- Mittwoch, 30.04., 11:30 Uhr
- Freitag, 02.05., 16:30 Uhr

Dauer: ca. zwei Stunden. Die Teilnahme ist verpflichtend.

Deadline

Deadlines

Folgende Termine sind einzuhalten:

- bis Ende Juli 2008: Gliederung vorlegen
- bis Ende August 2008: erste Fassung der Ausarbeitung
- bis Freitag, 19.09.2008: endgültige Fassung der Ausarbeitung
- bis Montag, 29.09.2008: endgültige Fassung der Folien

Blockveranstaltung

Termine der abschließenden Blockveranstaltung

Vorträge als Blockveranstaltung:

- Dienstag, 07.10.2008
- Mittwoch, 08.10.2008

Die Teilnahme an beiden Terminen ist verpflichtend.

Architecture Analysis & Design Language (AADL)

- Society of Automotive Engineers (SAE), seit 1995
- Anwendung: eingebettete Echtzeitsysteme, speziell in der Avionik (Luft- und Raumfahrt)
- Beispiel:

```
system implementation Computer.Impl
subcomponents
  CPU: processor Processor.Impl accesses BUS;
  MEM: memory Memory.Impl accesses BUS;
  BUS: bus Bus.Impl;
  PRC: process Process.Impl stored in MEM;
  THR: thread Thread.Impl running on CPU;
end Computer.Impl;
```

Estelle

- ISO International Standard seit 1989
- Formaler Beschreibungsansatz zur Spezifikation von Kommunikationsprotokollen
- Beispiel:

```
state IDLE, CALLING, CALLED, CONNECTED, CLOSING;
initialize
        to IDLE
        begin end;

trans
        from IDLE
                to CALLING
                        when U.ConReq
                                begin
                                end;
                to CALLED
                        begin
                                output U.ConInd
                        end;
```

Language Of Temporal Ordering Specification (LOTOS)

- ISO 8807 Standard seit 1990
- Spezifikation nebenläufiger und verteilter Systeme (Prozesse und Datentypen)
- Beispiel:

```
SPECIFICATION buffer2 [ get , put ] : noexit
BEHAVIOUR
    ( buffer1 [get ,middle] |[middle]| buffer1 [middle, put])
WHERE
   PROCESS buffer1 [get, put] : noexit :=
      get ?x:data ; put !x ; buffer1 [get, put]
   ENDPROC
TYPE data IS
   SORTS data
   OPNS 0,1  : -> bit
        inc : bit -> bit
   EQNS
   OFSORT bit
   inc(0) = 1 ;
   inc(1) = 0 ;
ENDTYPE
ENDSPEC
```

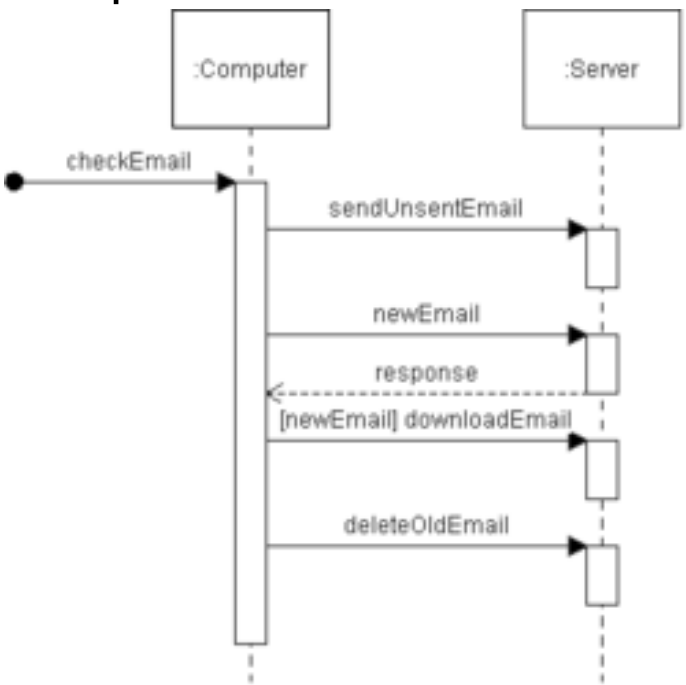
Maude

- Beschreibung des operationellen Systemverhaltens über Termersetzung
- Beispiel:

```
mod MUTEX is
  sorts Name Mode Proc Token Conf .
  subsorts Token Proc < Conf .
  op none : -> Conf [ctor] .
  op __ : Conf Conf -> Conf [ctor assoc comm id: none] .
  ops a b : -> Name [ctor] .
  ops wait critical : -> Mode [ctor] .
  op [_,_] : Name Mode -> Proc [ctor] .
  ops * $ : -> Token [ctor] .
  rl [a-enter] : $ [a, wait] => [a, critical] .
  rl [b-enter] : * [b, wait] => [b, critical] .
  rl [a-exit] : [a, critical] => [a, wait] * .
  rl [b-exit] : [b, critical] => [b, wait] $ .
endm
```

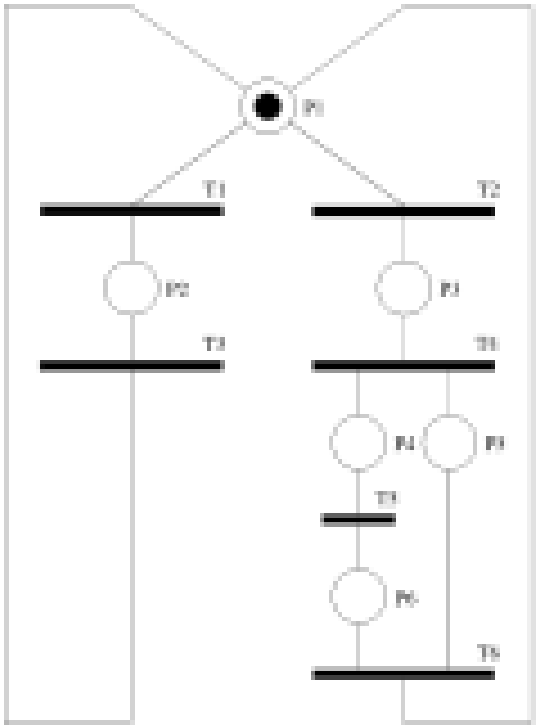
Message Sequence Charts

- ITU-Standard
- Graphischer Formalismus zur Beschreibung der Interaktion von Systemkomponenten
- Hauptanwendung: Telekommunikation
- Beispiel:



Petri-Netze

- Carl Adam Petri, 1960
- Mathematisches Modell nebenläufiger Systeme
- Beispiel:



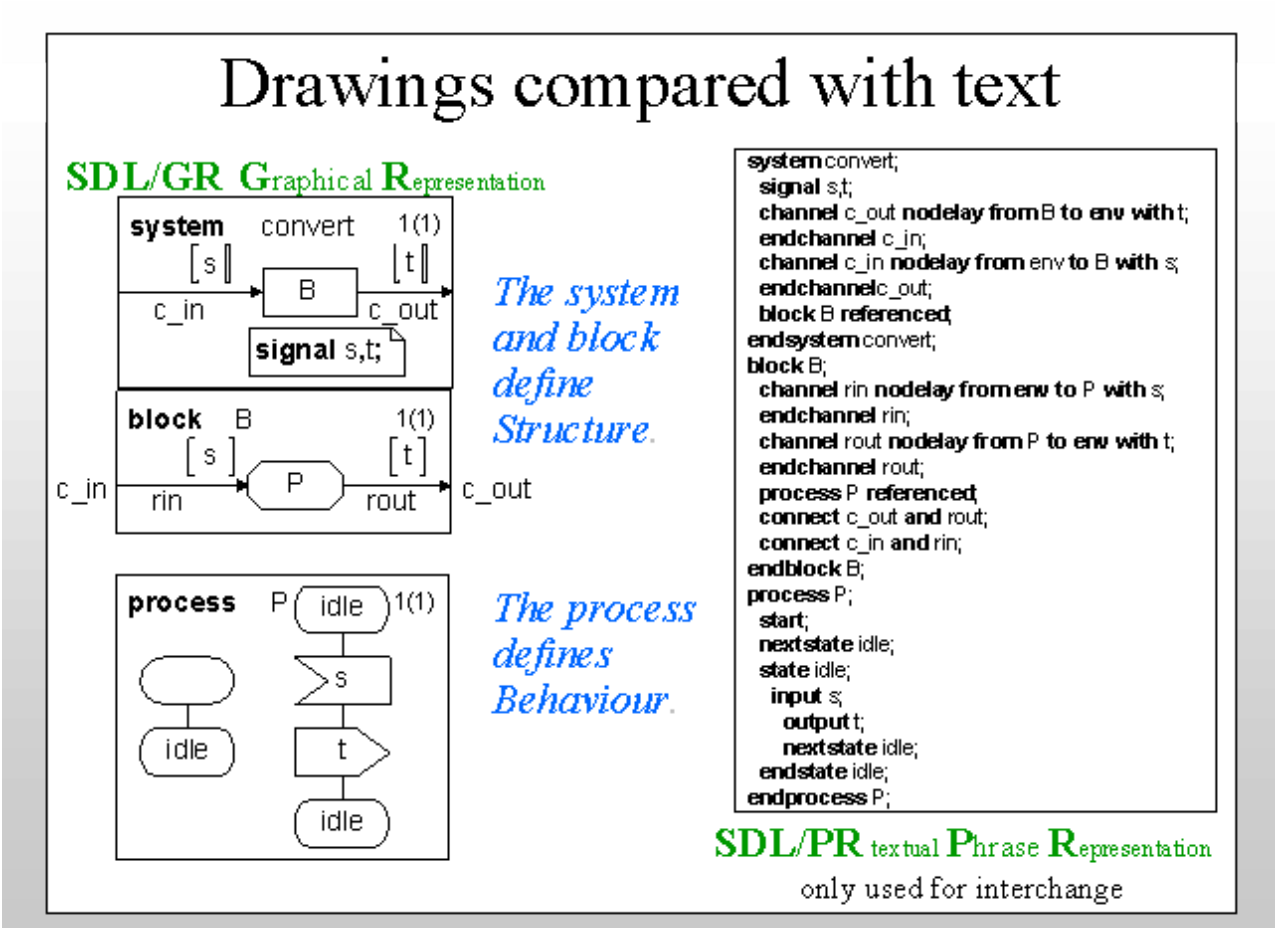
PROcess MEta LAnguage (PROMELA)

- System = Menge kommunizierender Prozesse (Automaten)
- Eingabesprache des Spin-Model-Checkers
- Beispiel:

```
chan to_sndr = [2] of {mtype};
chan to_rcvr = [2] of {mtype};
active proctype Sender()
{again: to_rcvr!msg1 ;
        to_sndr?ack1 ;
        to_rcvr!msg0 ;
        to_sndr?ack0 ;
        goto again}
active proctype Receiver()
{again: to_rcvr?msg1;
        to_sndr!ack1;
        to_rcvr?msg0;
        to_sndr!ack0;
        goto again;}
```

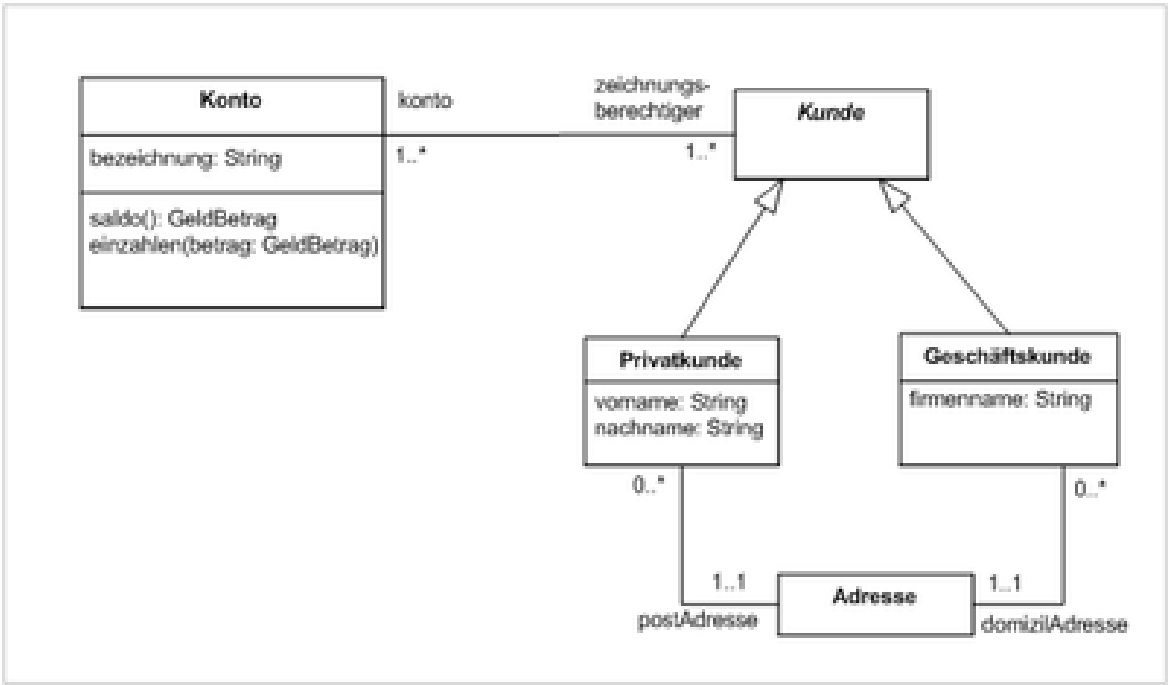
Specification and Description Language (SDL)

- ITU Standard Z.100
- Spezifikation reaktiver und verteilter Systeme als interagierende endliche Automaten
- Beispiel:



Unified Modeling Language (UML)

- Dominierenden Sprache für die Modellierung von Anwendungssystemen
- Beschreibung statischer (Klassendiagramme, Architekturdiagramme, ...) und dynamischer (Zustandsdiagramme, Aktivitätsdiagramme, ...) Aspekte
- Beispiel:



Vienna Development Method (VDM)

- IBM Vienna Laboratory seit 1970
- Formale Beschreibung und Verifikation von Verfeinerungsschritten
- Beispiel:

```
types
  Qelt = token;
  Queue = seq of Qelt;
operations
  ENQUEUE(e:Qelt)
    ext wr q:Queue
    post q = q~ ^ [e];
  DEQUEUE()e:Qelt
    ext wr q:Queue
    pre q <> []
    post q~ = [e]^q;
  IS-EMPTY()r:bool
    ext rd q:Queue
    post r <=> (len q = 0)
```

Very High Speed Integrated Circuit Hardware Description Language (VHDL)

- IEEE Standard 1076 seit 1993
- Hardwarebeschreibungssprache zur spezifikation digitaler Systeme
- Beispiel:

```
ENTITY DFlipflop IS
    PORT(D,Clk: IN Bit;
         Q: OUT Bit);
  END DFlipflop;
  ARCHITECTURE Behav OF DFlipflop IS
      CONSTANT T_Clk_Q: time := 4.23 ns;
  BEGIN
      PROCESS
      BEGIN
         WAIT UNTIL Clk'EVENT AND Clk'Last_Value='0' AND Clk='1';
         Q<=D AFTER T_Clk_Q;
      END PROCESS;
  END Behav;
```

The Z Notation

- Entwickelt am Oxford University Computing Laboratory seit 1975, ISO Standard seit 2002
- Basiert auf Mengentheorie und Prädikatenlogik erster Stufe
- Beispiel: