Basic System Entity Structure Concepts

Faculty of Engineering / Research Group CEA

Thorsten Pawletta

E-Mail: thorsten.pawletta@hs-wismar.de

Web: www.hs-wismar.de / www.cea-wismar.de
SES/MB Modeling Approach

Analyzing and separating the specification of systems allows for conceptual system structures and components to be modeled. Pruning these structures leads to a pruned entity structure, which is then used to build the executable model. Formal modeling and implementation are key steps in this process.

SES/MB Modeling Approach
Formal Modeling
SES/MB Modeling Approach

Formal Modeling

- SES describes permissible structure & parameter variants
SES/MB Modeling Approach

Formal Modeling

- SES describes permissible structure & parameter variants
- MB defines basic dynamic models
Basics of the System Entity Structure (SES)
Basics of the System Entity Structure (SES)

- SES introduced by B.P. Zeigler and J. Rozenblit
Basics of the System Entity Structure (SES)

• SES introduced by B.P. Zeigler and J. Rozenblit
• Amongst others extended by research group CEA (Wismar)
Basics of the System Entity Structure (SES)

- SES introduced by B.P. Zeigler and J. Rozenblit
- Amongst others extended by research group CEA (Wismar)
- SES is a tree structure
Basics of the System Entity Structure (SES)

- SES introduced by B.P. Zeigler and J. Rozenblit
- Amongst others extended by research group CEA (Wismar)
- SES is a tree structure
  - Well defined by axioms
Basics of the System Entity Structure (SES)

• SES introduced by B.P. Zeigler and J. Rozenblit
• Amongst others extended by research group CEA (Wismar)
• **SES is a tree structure**
  - Well defined by axioms
  - Two types of nodes
    - Entity nodes
    - Descriptive nodes
Basics of the System Entity Structure (SES)

- SES introduced by B.P. Zeigler and J. Rozenblit
- Amongst others extended by research group CEA (Wismar)
- **SES is a tree structure**
  - Well defined by axioms
  - Two types of nodes
    - Entity nodes
    - Descriptive nodes

**Entity nodes**
- real or imaginary objects

**Descriptive nodes**
- Aspect
- (Multi-aspect)
- Specialization
Basics of the System Entity Structure (SES)

• SES introduced by B.P. Zeigler and J. Rozenblit
• Amongst others extended by research group CEA (Wismar)

• SES is a tree structure
  ➢ Well defined by axioms
  ➢ Two types of nodes
    ➢ Entity nodes
    ➢ Descriptive nodes
  ➢ Three types of edges (relations between nodes)

<table>
<thead>
<tr>
<th>Entity nodes</th>
<th>Descriptive nodes</th>
</tr>
</thead>
<tbody>
<tr>
<td>real or imaginary</td>
<td>Aspect</td>
</tr>
<tr>
<td>objects</td>
<td>(Multi-aspect)</td>
</tr>
<tr>
<td></td>
<td>Specialization</td>
</tr>
</tbody>
</table>
Basics of the System Entity Structure (SES)

- SES introduced by B.P. Zeigler and J. Rozenblit
- Amongst others extended by research group CEA (Wismar)

- SES is a tree structure
  - Well defined by axioms
  - Two types of nodes
    - Entity nodes
    - Descriptive nodes
  - Three types of edges (relations between nodes)
  - Node/Edge specific attributes

<table>
<thead>
<tr>
<th>Entity nodes</th>
<th>real or imaginary objects</th>
</tr>
</thead>
<tbody>
<tr>
<td>Descriptive nodes</td>
<td>Aspect (Multi-aspect) Specialization</td>
</tr>
</tbody>
</table>
Basics of the System Entity Structure (SES)

- SES introduced by B.P. Zeigler and J. Rozenblit
- Amongst others extended by research group CEA (Wismar)
- SES is a tree structure
  - Well defined by axioms
  - Two types of nodes
    - Entity nodes
    - Descriptive nodes
  - Three types of edges (relations between nodes)
  - Node/Edge specific attributes
  - Global variables, functions, constraints, ...

<table>
<thead>
<tr>
<th>Entity nodes</th>
<th>real or imaginary objects</th>
</tr>
</thead>
<tbody>
<tr>
<td>Descriptive nodes</td>
<td>Aspect (Multi-aspect)</td>
</tr>
<tr>
<td></td>
<td>Specialization</td>
</tr>
</tbody>
</table>
Case Study
Case Study

- Feedback control system
Case Study

- Feedback control system
- Described by transfer functions

\[ G_{Su}(s) = \frac{1}{20 \cdot s + 1} \]
Case Study

- Feedback control system
- Described by transfer functions
- Influenced by disturbances

\[ G_{Su}(s) = \frac{1}{20 \cdot s + 1} \]

\[ G_{Sz}(s) = \frac{1}{10 \cdot s + 1} \]
Case Study

- Feedback control system
- Described by transfer functions
- Influenced by disturbances
- Measurable disturbances
  → Compensated with feedforward control

\[
G_{Su}(s) = \frac{1}{20 \cdot s + 1} \\
G_{Sz}(s) = \frac{1}{10 \cdot s + 1} \\
G_{St}(s) = \frac{20 \cdot s + 1}{10 \cdot s + 1}
\]
Case Study (2)

- Two system structure variants
  - Without feedforward control: \( \text{feedforward}=0 \)
  - With feedforward control: \( \text{feedforward}=1 \)
Case Study (2)

- Two system structure variants
  - Without feedforward control: \( \text{feedforward}=0 \)
  - With feedforward control: \( \text{feedforward}=1 \)
- For every structure variant
  - Different parameter configurations of PID controller (we consider two)
Case Study (2)

Design objective:
Find best control configuration.

• Two system structure variants
  ➢ Without feedforward control: \( \text{feedforward} = 0 \)
  ➢ With feedforward control: \( \text{feedforward} = 1 \)

• For every structure variant
→ Different parameter configurations of PID controller
   (we consider two)
SES/MB-based Modeling of the Case Study
SES/MB-based Modeling of the Case Study

**SES (excerpt)**
- ctrlSys
  - ctrlSysDEC
    - cplg
  - feedforwardCtrl
    - sourceSys
    - feedbackSys
      - mb='MB/Constant'
      - mb='MB/Feedback'
    - selection
  - feedforwardCtrlSPEC
    - NONE
    - fc

**MB**
- b(s) / a(s)
  - Step
  - TransferFunction
  - Constant
  - Feedback
  - PID
  - Add
SES/MB-based Modeling of the Case Study

SES (excerpt)

ctrlSys

ctrlSysDEC

cplg

feedforwardCtrl

sourceSys

feedbackSys

{k=0}

{mb='MB/Constant'}

{mb='MB/Feedback'}

selection

NONE

fc

MB

Step

TransferFunction

b(s)

a(s)

Constant

Feedback

PID

Add

Variability Modeling and Simulation Using Multiple Simulators
SES/MB-based Modeling of the Case Study

 SES (excerpt)

ctrlSys

ctrlSysDEC

cplg

feedforwardCtrl

sourceSys

feedbackSys

\{ mb='MB/Constant' \}

\{ mb='MB/Feedback' \}

feedbackCtrlSPEC

selection

NONE

fc

...

MB

Step

TransferFunction

b(s)

a(s)

Constant

Feedback

PID

Add
SES/MB-based Modeling of the Case Study

Systems

Structures

Components

Separation of specification

SES (excerpt)

ctrlSys

ctrlSysDEC
cplg

feedforwardCtrl

sourceSys

feedbackSys

{mb='MB/Constant'
{k=0}}

{mb='MB/Feedback'}

feedforwardCtrlSPEC

selection

NONE

fc

MB

Step

TransferFunction

Constant

Feedback

PID

Add

b(s)

a(s)
SES/MB-based Modeling of the Case Study

Systems

Separation of Specification

Structures

Components

SES (excerpt)

- ctrlSys
- ctrlSysDEC
- cplg
- feedforwardCtrl
- sourceSys
- feedbackSys

MB

- Step
- TransferFunction
- Constant
- Feedback
- PID
- Add
SES/MB-based Modeling of the Case Study

**SES**

- ctrlSys
  - ctrlSysDEC
    - cplg

- feedforwardCtrl
- sourceSys
- feedbackSys
  {mb= 'MB/Constant' k=0}

- feedforwardCtrlSPEC selection

**MB**

- \( \frac{b(s)}{a(s)} \)

- Step
- TransferFunction

- Constant
- Feedback
- PID
- Add
SES/MB-based Modeling of the Case Study

**SES**
- **ctrlSys**
- **ctrlSysDEC**
- **cplg**
- **feedforwardCtrl**
- **sourceSys**
- **feedbackSys**
  - mb='MB/Constant' \( \{k=0\} \)
  - mb='MB/Feedback'

**MB**
- **Step**
- **TransferFunction**
  - **b(s)**
  - **a(s)**

**Components**

**Systems**
- separation of specification

**Structures**

**Components**
- feedforward controller \( G_{ff}(s) \)
- disturbance \( G_{d}(s) \)
- PID controller
- process unit \( G_{pu}(s) \)
- PT1
- setpoint
- disturbance \( G_{d}(s) \)
- manipulated variable
- controlled variable

**feedforward controller**
- **PID controller**
- **process unit**
- **setpoint**
- **disturbance**
- **manipulated variable**
- **controlled variable**

---

**SES (excerpt)**
- **feedforwardCtrl**
- **sourceSys**
- **feedbackSys**
  - mb='MB/Constant' \( \{k=0\} \)
  - mb='MB/Feedback'

**MB**
- **Constant**
- **Feedback**
- **PID**
- **Add**

---

**NONE**

**fc**
SES/MB-based Modeling of the Case Study

**Systems**

- **Structures**
  - ctrlSys
  - ctrlSysDEC
  - cplg
- **Components**
  - sourceSys
  - feedbackSys

**SES (excerpt)**

- feedforwardCtrl
- sourceSys
- feedbackSys

**MB**

- Step
- TransferFunction

- Constant
- Feedback
- PID
- Add

- b(s)
- a(s)
- \( G_{ul}(s) \)
- \( G_{ul}(s) \) PT1
- setpoint
- disturbance
- manipulated variable
- controlled variable
- \( G_{ul}(s) \) PT1
- feedback controller PID
- disturbance
- feedforward controller \( G_{ul}(s) \)

...
More Detailed Extract of the SES

**SES**

- SESVAR = {feedforward}
- SemanticCondition = {feedforward in [0,1]}

**ctrlSys**

**ctrlSysDEC**

- {cplg1}

**feedforwardCtrl**

- {mb='MB/Constant', k=0}

**feedforwardCtrlSPEC**

- specrule:
  - feedforward == 0 → NONE
  - feedforward == 1 → fc

- NONE
- fc

**sourceSys**

- {mb='MB/Feedback'}

**feedbackSys**

- {mb='MB/PID', k=1, Ti=1, Td=0}

**ctrlPIDSys**

- {mb='MB/TransferFunction', b={1}, a={20,1}}

**procUnitSys**

- sourceDist
  - {...}

**sourceDist**

- tfDist
  - {...}

**tfDist**

- addDist
  - {...}
More Detailed Extract of the SES

\[
\text{SES} \quad \text{SESVAR} = \{ \text{feedforward} \} \\
\text{SemanticCondition} = \{ \text{feedforward in } [0,1] \}
\]

- **ctrlSys**
  - **ctrlSysDEC** \{cplg1\}
    - **feedforwardCtrl** \{mb='MB/Constant', k=0\}
    - **sourceSys** \{mb='MB/Feedback'\}
    - **feedbackSys** \{mb='MB/Feedback'\}

- **feedforwardCtrlSPEC** \{specrule: feedforward == 0→NONE, feedforward == 1→fc \}
  - **NONE**
  - **fc**
    - **fcDEC** \{cplg2\}

- **addFeedforward** \{mb='MB/Add'\}
- **tfFeedforward** \{mb='MB/TransferFunction'\}

**Key**
- **Type**
  - **Aspect** DEC
  - **Specialization** SPEC
- **Suffix Edge**
Variability Modeling and Simulation Using Multiple Simulators

More Detailed Extract of the SES

SES

\[
\text{SESVAR}=\{\text{feedforward}\}
\]
\[
\text{SemanticCondition}=\{\text{feedforward in } [0,1]\}
\]

ctrlSys

ctrlSysDEC

{cplg1}

feedforwardCtrl

\{mb='MB/Constant', k=0\}

sourceSys

\{mb='MB/Feedback'\}

feedbackSys

ctrlPI/DSys

\{mb='MB/PID', k=1, Ti=1, Td=0\}

procUnitSys

\{mb='MB/TransferFunction'\}

sourceDist

\{...\}

tfDist

\{...\}

addDist

\{...\}

feedforwardCtrlSPEC

\{specrule: feedforward == 0→NONE, feedforward == 1→fc\}

NONE

fc

fcDEC

\{cplg2\}

addFeedforward

\{mb='MB/Add'\}

tfFeedforward

\{mb='MB/TransferFunction'\}
More Detailed Extract of the SES

**SES**
- `SESVAR={feedforward}`
- `SemanticCondition={feedforward in [0,1]}`

- `ctrlSys`
- `ctrlSysDEC {cplg1}`

- `feedforwardCtrl {mb='MB/Constant', k=0}`
- `sourceSys {mb='MB/Feedback'}`
- `feedbackSys {mb='MB/TransferFunction'}`

- `feedforwardCtrlSPEC {specrule: feedforward == 0→NONE, feedforward == 1→fc }`
- `NONE`, `fc`

- `fcDEC {cplg2}`

- `addFeedforward {mb='MB/Add'}`
- `tfFeedforward {mb='MB/TransferFunction'}`

- `ctrlPIDSys {mb='MB/PID', k=1, Ti=1, Td=0}`
- `procUnitSys {mb='MB/TransferFunction'}`
- `sourceDist {...}`
- `tfDist {...}`
- `addDist {...}`
More Detailed Extract of the SES

SES

SESVAR={feedforward}
SemanticCondition={feedforward in [0,1]}

ctrlSys

ctrlSysDEC {cplg1}

feedforwardCtrl
{mb='MB/Constant' k=0}
feedbackSys
{mb='MB/Feedback'}

feedforwardCtrlSPEC
{specrule:
feedforward == 0→NONE
feedforward == 1→fc  }

NONE fc

fcDEC {cplg2}

addFeedforward {mb='MB/Add'}
tfFeedforward {mb='MB/TransferFunction'}

ctrlPIDSys
{mb='MB/PID'
 k=1
 Ti=1
 Td=0 }

procUnitSys
{mb='MB/TransferFunction'
 k=1
 b={1}
 a={20,1} }

sourceSys
{mb='MB/Constant'}

sourceDist
{...}

tfDist
{...}

addDist
{...}
More Detailed Extract of the SES

**SES**
- SESVAR={feedforward}
- SemanticCondition={feedforward in [0,1]}

- ctrlSys
  - ctrlSysDEC {cplg1}

- feedforwardCtrl
  - sourceSys {mb='MB/Constant', k=0}
  - feedbackSys {mb='MB/Feedback'}

- feedforwardCtrlSPEC
  - specrule:
    - feedforward == 0 → NONE
    - feedforward == 1 → fc

- NONE fc

- fcDEC {cplg2}

- addFeedforward {mb='MB/Add'}
- tfFeedforward {mb='MB/TransferFunction'}

**Key**
- Type
- Suffix
- Edge
- Aspect DEC
- Specialization SPEC
Variability Modeling and Simulation Using Multiple Simulators

More Detailed Extract of the SES

SES

- SESVAR={feedforward}
- SemanticCondition={feedforward in [0,1]}

ctrlSys

- ctrlSysDEC={cplg1}

- Coupling attribute cplg1 is specified using an SES function depending on feedforward

- feedforwardCtrl
  - sourceSys={mb='MB/Constant', k=0}
  - feedbackSys={mb='MB/Feedback'}

- feedforwardCtrlSPEC={specrule:
  feedforward == 0 → NONE
  feedforward == 1 → fc}

  - NONE
  - fc

- fcDEC={cplg2}

- addFeedforward={mb='MB/Add'}

- tfFeedforward={mb='MB/TransferFunction'}

- ctrlPIDSys={mb='MB/PID', k=1, Ti=1, Td=0}
  - procUnitSys={mb='MB/TransferFunction'}
  - sourceDist={...}
  - tfDist={...}
  - addDist={...}

- Key
  - Type
  - Suffix
  - Edge

  - Aspect: DEC

  - Specialization: SPEC
More Detailed Extract of the SES

- Coupling attribute $cplg1$ is specified using an SES function depending on feedforward.
- Different parameter settings are not shown in this extract of SES.
SES/MB Modeling Approach

Deployment

- Systems
  - Structures
  - System Entity Structure
  - Pruned Entity Structure
  - formal linkage
  - pruning

- Components
  - Model Base
  - Executable Model
  - build

separation of specification
Outlook and Software Tool Support

The pruning and build processes are presented in the supplementary material for Chapter 18.6 in detail. The case study introduced here is revisited, Software tools supporting the SES/MB approach are introduced, the SES developed for the case study is pruned, and models are generated.