Entwurf, Simulation und Implementierung ereignisdiskreter Steuerungen mit $PDEVS_{RCP}$ Version 2.0 (WIP)

Birger Freymann, Artur Schmidt, Sven Pawletta,
Sven Hartmann, Thorsten Pawletta

Research Group
CEA
Computational
Engineering & Automation

Birger Freymann
birger.freymann@hs-wismar.de
www.mb.hs-wismar.de/cea/

Agenda

1. Motivation
2. Rapid Control Prototyping Approach
3. PDEVS & $PDEVS_{RCP}$ Formalism
4. Open Problems of $PDEVS_{RCP}$
5. $PDEVS_{RCP}$ 2.0 Formalism
6. Application Example
7. Summary and Outlook

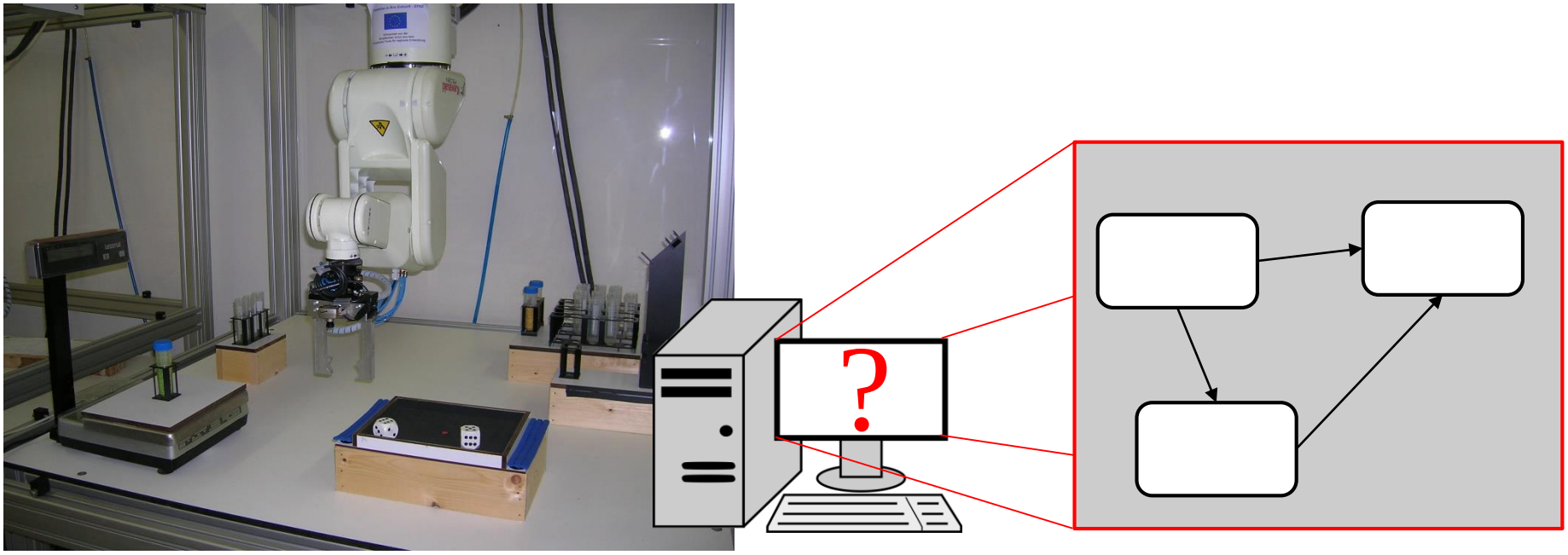
Agenda

1. **Motivation**
2. Rapid Control Prototyping Approach
3. PDEVS & $PDEVS_{RCP}$ Formalism
4. Open Problems of $PDEVS_{RCP}$
5. $PDEVS_{RCP}$ 2.0 Formalism
6. Application Example
7. Summary and Outlook

- **Nowadays control development is complex**
  testing, maintenance, extension, …
- **Ad-hoc implementations are often not possible**

➔ Design methodology:

Rapid Control Prototyping (RCP) approach by Abel (RWTH)

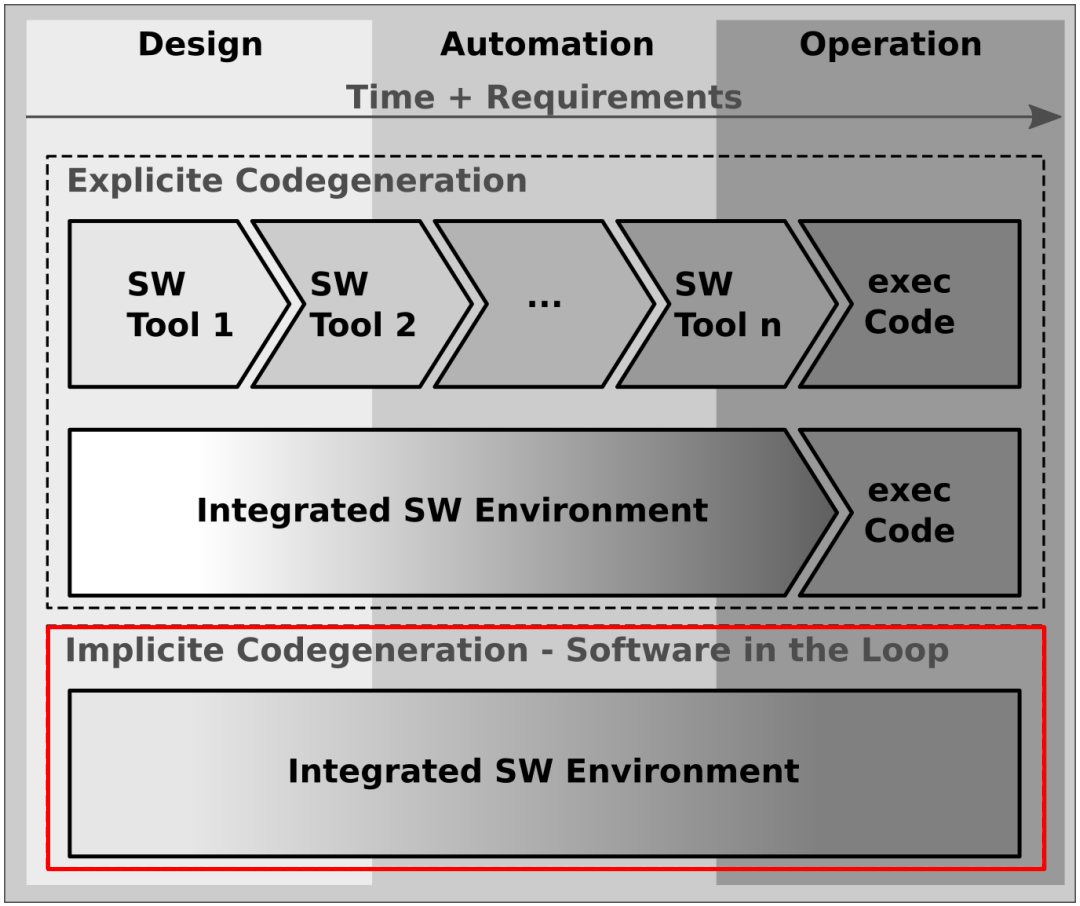


Agenda

1. Motivation
2. **Rapid Control Prototyping Approach**
3. PDEVS & $PDEVS_{RCP}$ Formalism
4. Open Problems of $PDEVS_{RCP}$
5. $PDEVS_{RCP}$ 2.0 Formalism
6. Application Example
7. Summary and Outlook

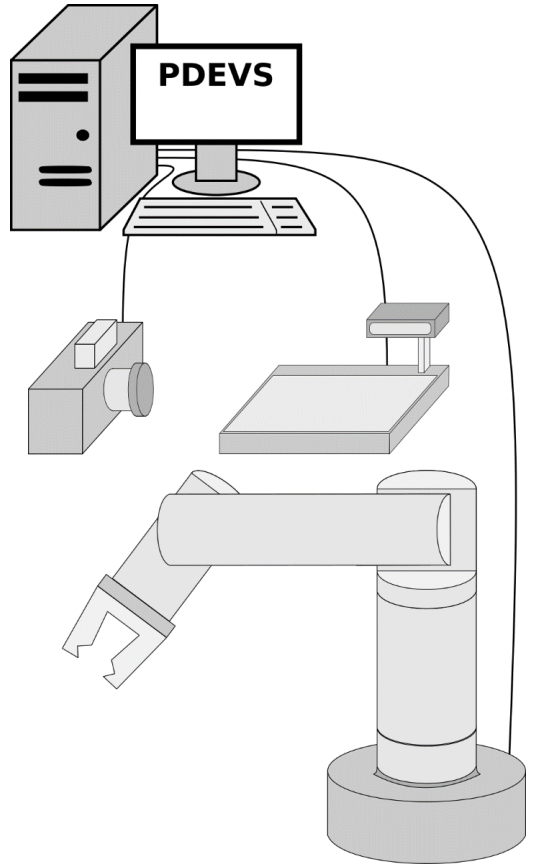
Rapid Control Prototyping Approach

- Model based approach
- Continuous control development
- Avoidance of re-implementations



event based M&S approach !

usage of PDEVS for real time control ?



Agenda

1. Motivation
2. Rapid Control Prototyping Approach
3. **PDEVS & $PDEVS_{RCP}$ Formalism**
4. Open Problems of $PDEVS_{RCP}$
5. $PDEVS_{RCP}$ 2.0 Formalism
6. Application Example
7. Summary and Outlook

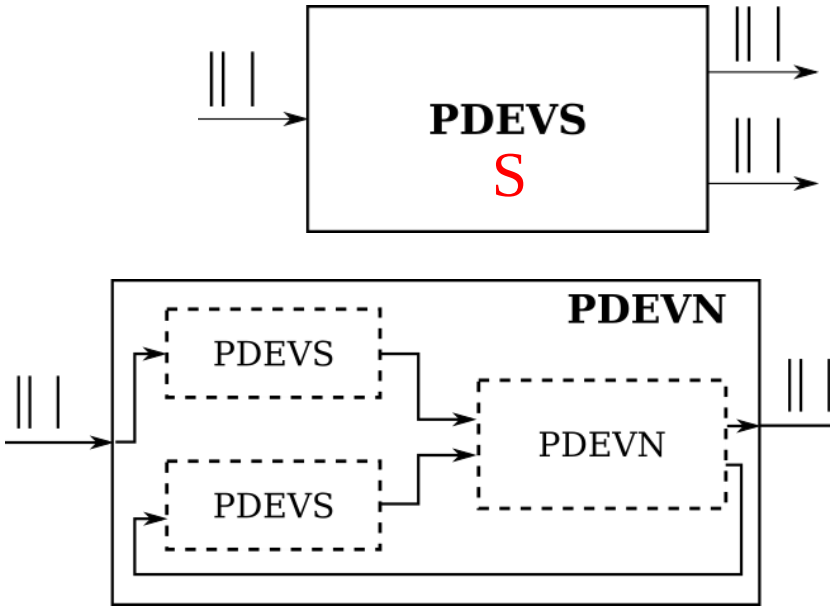
Parallel Discrete EVent System Specification

PDEVS (Chow 1996) is an extension of DEVS (Zeigler 1976)

- Event-oriented modeling approach based on modular hierarchical model specification
- PDEVS - two system types:

atomic PDEVS
dynamic behavior

coupled PDEVS (PDEVN)
composition of *atomic* or *coupled* PDEVS



PDEVS & $PDEVS_{RCP}$

$$PDEVS = \{X, Y, S, \delta_{int}, \delta_{ext}, \delta_{con}, \lambda, ta\}$$

$$PDEVS_{RCP} = \{X, Y, S, \delta_{int}, \delta_{ext}, \delta_{con}, \lambda, ta, A\} \quad \text{(RG CEA)}$$

$$X = X_{model} \cup X_{clock}$$

$$X_{model} = \{(p, v) | p \in IPorts, v \in X_p\}$$

$$\boldsymbol{X_{clock}} = \{(\text{"}\boldsymbol{clock}\text{"}, \boldsymbol{v}) | \boldsymbol{v} \in \mathbb{R}^+\}$$

$A = \{a_1, a_2, \ldots, a_n\}$ set of executable activities

$$a_n = (a_i, [ti_{min}, ti_{max}])$$

$\lambda : S \rightarrow Y \times A$ combined output and activity function

Agenda

1. Motivation
2. Rapid Control Prototyping Approach
3. PDEVS & $PDEVS_{RCP}$ Formalism
4. **Open Problems of $PDEVS_{RCP}$**
5. $PDEVS_{RCP}$ 2.0 Formalism
6. Application Example
7. Summary and Outlook

Open problems:

(i) Specification of **activities**

(ii) Specification of the **Real Time Clock** (RTC)

(iii) How are activities specified within atomic PDEVS$_{RCP}$ components ?

(iv) Dynamic is limited ($ta(s) \in \{0, \infty\}$)

$$PDEVS_{RCP} = \{X, Y, S, \delta_{int}, \delta_{ext}, \delta_{con}, \lambda, ta, A\}$$?

$$X_{clock} = \{("clock", v) | v \in \mathbb{R}^+\}$$

atomic RTC
PDEVS$_{RCP}$? → WCT

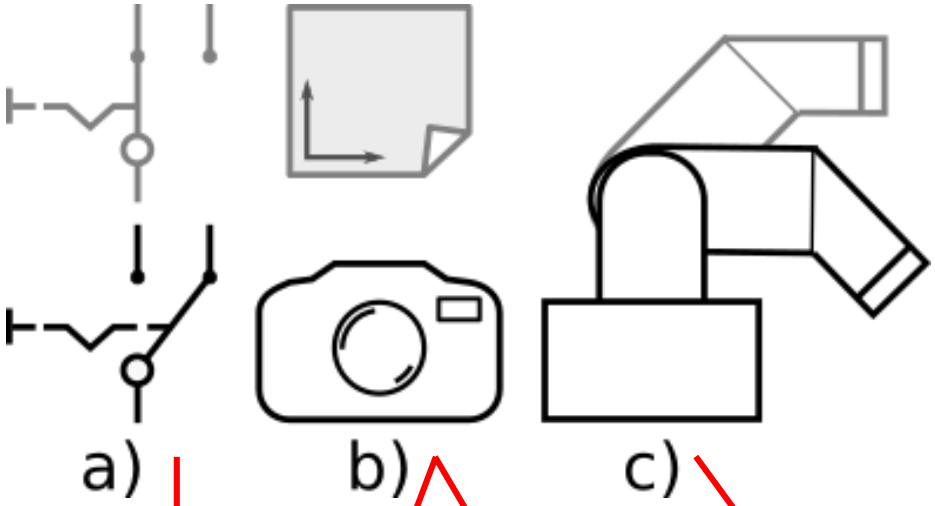
Dynamic PDEVS components with

- $ta(s) \in \{0, \infty\}$ ✓
- $ta(s) \in \mathbb{R}^+$ X → $ta(s) \in \{0, \infty\}$ → **re-implementation**

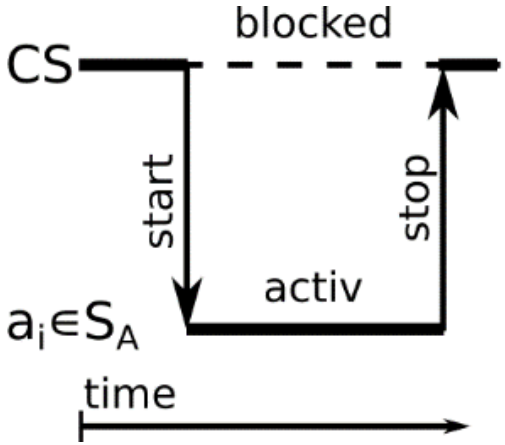
Agenda

1. Motivation
2. Rapid Control Prototyping Approach
3. PDEVS & $PDEVS_{RCP}$ Formalism
4. Open Problems of $PDEVS_{RCP}$
5. **$PDEVS_{RCP}$ 2.0 Formalism**
6. Application Example
7. Summary and Outlook

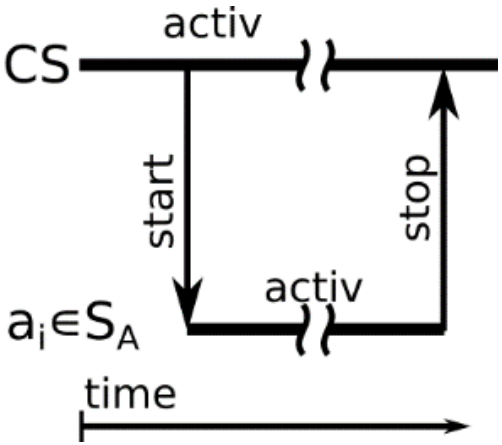
activities



synchronous



asynchronous



PDEVS$_{RCP}$ 2.0 Formalism

activity

$\boldsymbol{a_i = (M, X, S, Y, f_{ext}, f_{int}, f_{out})}$

$\boldsymbol{M}$	Set of activity methods
$\boldsymbol{X}$	Set of input values
$\boldsymbol{S}$	Set of states
$\mathbf{Y}$	Set of output values
$\boldsymbol{f_{ext}}$	$f_{ext}: S \times X \rightarrow S$
$\boldsymbol{f_{int}}$	$f_{int}: M \times S \rightarrow S$
$\boldsymbol{f_{out}}$	$f_{out}: S \rightarrow \mathrm{Y}$

$\boldsymbol{getWCT = (\mathrm{M}, X, S, Y, f_{ext}, f_{int}, f_{out})}$

$M = \{readSystemClock\}$

$X = \emptyset \Rightarrow f_{ext}$ not defined

$S = \{t_{Value}\}$ with $s_0 = 0$

$Y = \{t_{wct}\}$

$f_{int}(readSystemClock, t_{Value})$

$t_{Value} := readSystemClock$

$f_{out}(t_{Value})$

$t_{WCT} := t_{Value}$

$\boldsymbol{\mathrm{PDEVS}_{\mathrm{RCP}}\ 2.0 = (X, Y, S_{RCP}, \delta_{int}\delta_{ext}, \delta_{con}, \lambda_{RCP}, ta)}$

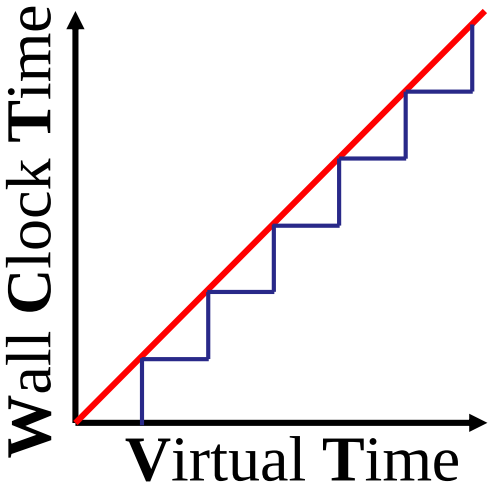
$S_{RCP} = S \cup S_A$

$S_A = \{a_1, \ldots, \boldsymbol{a_i}, \ldots, a_n\}$ //set of activities

$\lambda_{RCP}: S_{RCP} \rightarrow Y \times S_A$ // combined output and activity function

Real Time Clock (RTC)

atomic RTC
PDEVS$_{RCP}$ 2.0



$\boldsymbol{RTC} = (\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{S_{RCP}}, \boldsymbol{\delta_{int}}\boldsymbol{\delta_{ext}}, \boldsymbol{\delta_{con}}, \boldsymbol{\lambda_{RCP}}, \boldsymbol{ta})$
$X = \emptyset, Y = \emptyset \Rightarrow \ \delta_{ext}, \delta_{con}$ not defined
$S_{RCP} = S \cup S_A; S = \{\sigma, t_{Last}, t_{WCT}\}; S_A = \{\boldsymbol{getWCT}\}$
$s_{RCP,0} = (0,0,0, getWCT)$
$ta(\sigma, t_{Last}, t_{WCT}, getWCT) := \ \sigma$
$\lambda_{RCP}(\sigma, t_{Last}, t_{WCT}, getWCT)$
$\quad \boldsymbol{getWCT.f_{int}(getWCT.M, getWCT.S)}$
$\delta_{int}(\sigma, t_{Last}, t_{WCT}, getWCT)$
$\quad \boldsymbol{t_{WCT} := getWCT.f_{out}(getWCT.S)}$
$\quad if\ t_{WCT} - \ t_{Last} < 0.1$
$\quad$ yes: $\sigma := 0$ **//VT advance = 0**
$\quad$ no: $\sigma := 0.1; t_{Last} := t_{WCT}$ **//VT advance = 0.1**

Agenda

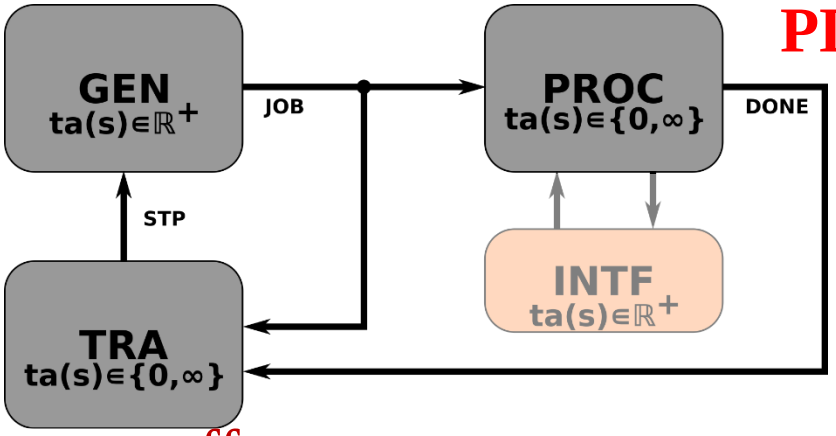
1. Motivation
2. Rapid Control Prototyping Approach
3. PDEVS & $PDEVS_{RCP}$ Formalism
4. Open Problems of $PDEVS_{RCP}$
5. $PDEVS_{RCP}$ 2.0 Formalism
6. **Application Example**
7. Summary and Outlook

Application Example

input puffer

robot

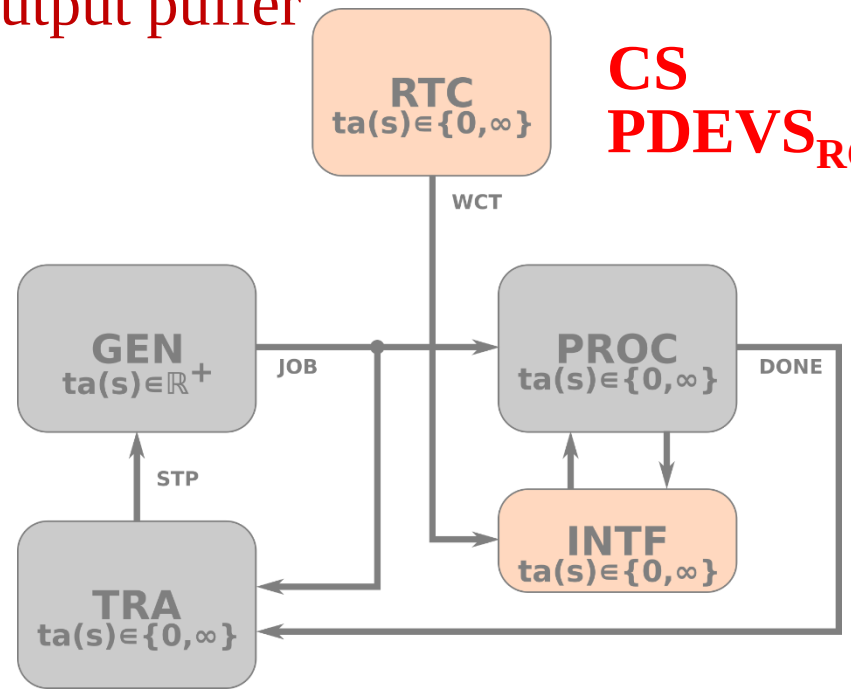
SM
PDEVS

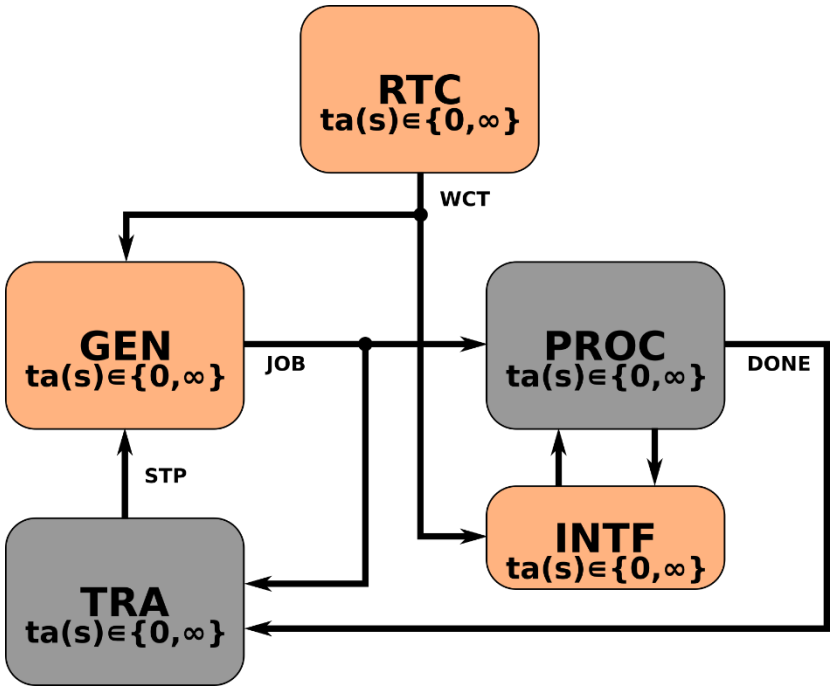


output puffer

GEN	Generator
PROC	Processor
TRA	Transducer
INTF	Interface
RTC	Real Time Clock
SM	Simulation Model
CS	Control Software

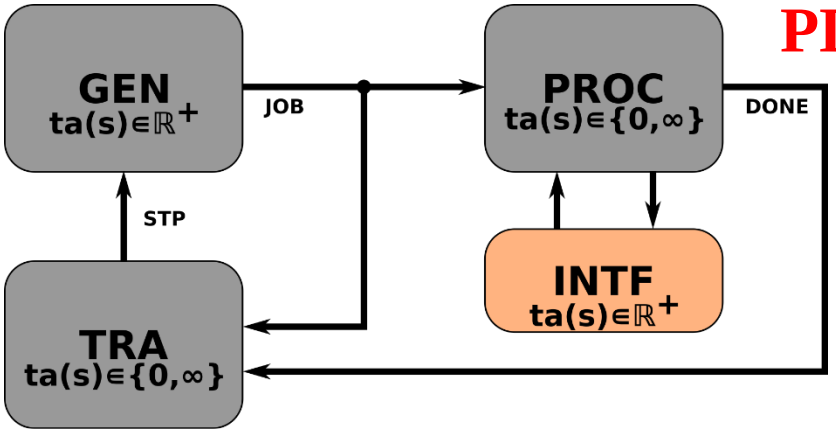
CS
PDEVS$_{RCP}$





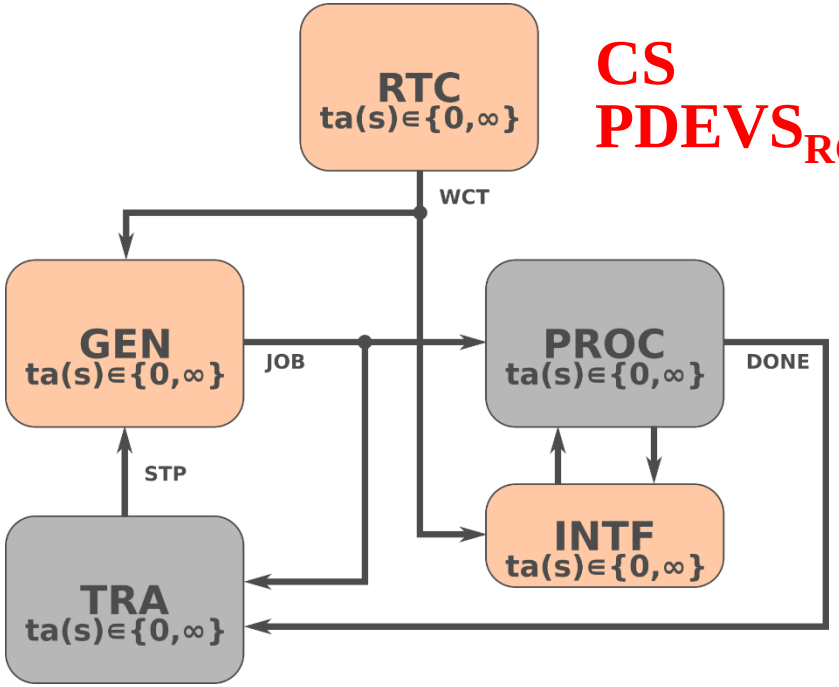
Application Example

SM
PDEVS

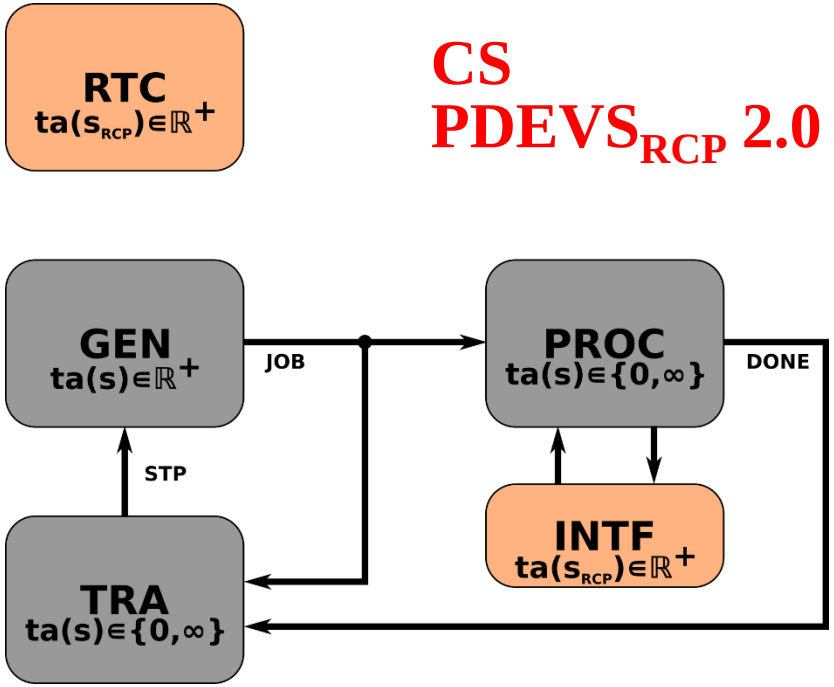


GEN	Generator
PROC	Processor
TRA	Transducer
INTF	Interface
RTC	Real Time Clock
SM	Simulation Model
CS	Control Software

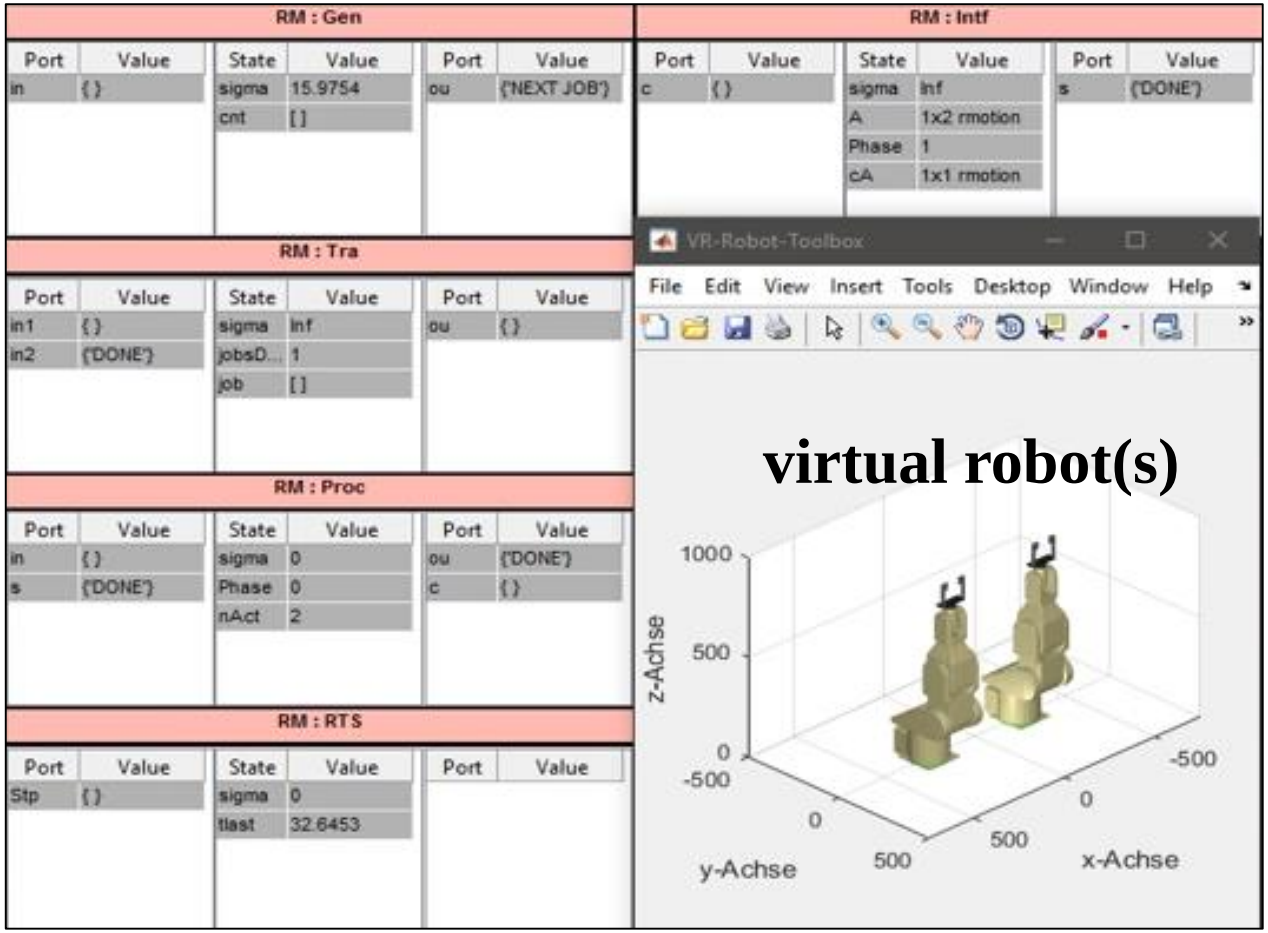
CS
PDEVS$_{RCP}$



CS
PDEVS$_{RCP}$ 2.0



CS based on PDEVS$_{RCP}$ 2.0 (MATLAB/DEVS)



Agenda

1. Motivation
2. Rapid Control Prototyping Approach
3. PDEVS & $PDEVS_{RCP}$ Formalism
4. Open Problems of $PDEVS_{RCP}$
5. $PDEVS_{RCP}$ 2.0 Formalism
6. Application Example
7. **Summary and Outlook**

Summary and Outlook

Summary:

- **Analysis of first version of $PDEVS_{RCP}$**
- **Highlighting of open problems of $PDEVS_{RCP}$**
- **Definition and implementation of $PDEVS_{RCP}$ 2.0**
- **Comparision of both $PDEVS_{RCP}$ variants**

Outlook:

- **Classification of $PDEVS_{RCP}$ 2.0 in context of system therory**
- **Usage of the new concept for robot control development**

Thank you for your attention

[1] D. Abel, A. Bollig, Rapid Control Prototyping, Methoden und Anwendung, Springer, 2006.
[2] G. Maletzki, Rapid Control Prototyping komplexer und flexibler Robotersteuerungen auf Basis des SBC-Ansatzes. Diss., Uni. Rostock/HS Wismar, 2014.
[3] B.P. Zeigler, H. Praehofer, T.G. Kim, Theory of modeling and simulation 2nd ed., Acad. Pr, 2000.
[4] S.M. Cho, T.G. Kim, Real Time Simulation Framework for RT-DEVS Models, Trans. Soc. Comput. Simul. Int. 18 (4) (2001) 203–215.
[5] J.L. Risco-Martín, S. Mittal, J.C. Fabero, P. Malagón, J.L. Ayala, Real-time Hardware/Software Co-design Using Devs-based Transparent M&S Framework, in: Proc. of the Summer Computer Conf., 45:1-45:8.
[6] T. Schwatinski, T. Pawletta, S. Pawletta, C. Kai-ser: Simulation-based development and operation of controls on the basis of the DEVS for-malism. Proc. 7th EUROSIM Conf., Vol.2, Prag, Czech Rep., 2010, 8 pages.
[7] A.C.-H. Chow, Parallel DEVS: A Parallel, Hierarchical, Modular Modeling Formalism and Its Distributed Simulator, Trans. Soc. Comput. Simul. Int. 13 (2) (1996) 55–67.
[8] B. Freymann, T. Pawletta, S. Pawletta, Multi-Robotersteuerungen mit variablen Interaktionsprinzipien auf Basis des Simulation Based Control Frameworks und dem Dicrete Event System Specification Formalismus, Proc. ASIM STS/GMMS WS, ARGESIM Rep. No.50, 2015, 67–77.
[9] B. Freymann, T. Pawletta, T. Schwatinski, S. Pawletta, Modellbibliothek für die Interaktion von Robotern in der MATLAB/DEVS-Umgebung auf Basis des SBC-Frameworks, Proc. ASIM STS/GMMS WS, ARGESIM Rep. No.42, 2014, 199–208.

Backup

CLOSURE UNDER COUPLING