MODELING OF COMPLEX AUTOMATION SYSTEMS USING COLORED STATE CHARTS

Olga Fengler
Wolfgang Fengler
Bernd Däne
Vesselka Duridanova
Ilmenau Technical University
Department of Computer Architectures
E-mail: olga.fengler@tu-ilmenau.de
Web: http://tin.tu-ilmenau.de/ra

TU Ilmenau

Outline

1. Motivation

2. State Charts

3. Colored State Charts

4. Conversion into High Level Petri Nets

5. Example model

6. Conclusion und Perspectives

# Motivation

State Charts:

many similar objects
→ many elements
→ very large and disturbing diagrams

Extending State Charts:

- ✓ Reducing complexity
- ✓ Partly introducing the well established concept of High Level Petri Nets
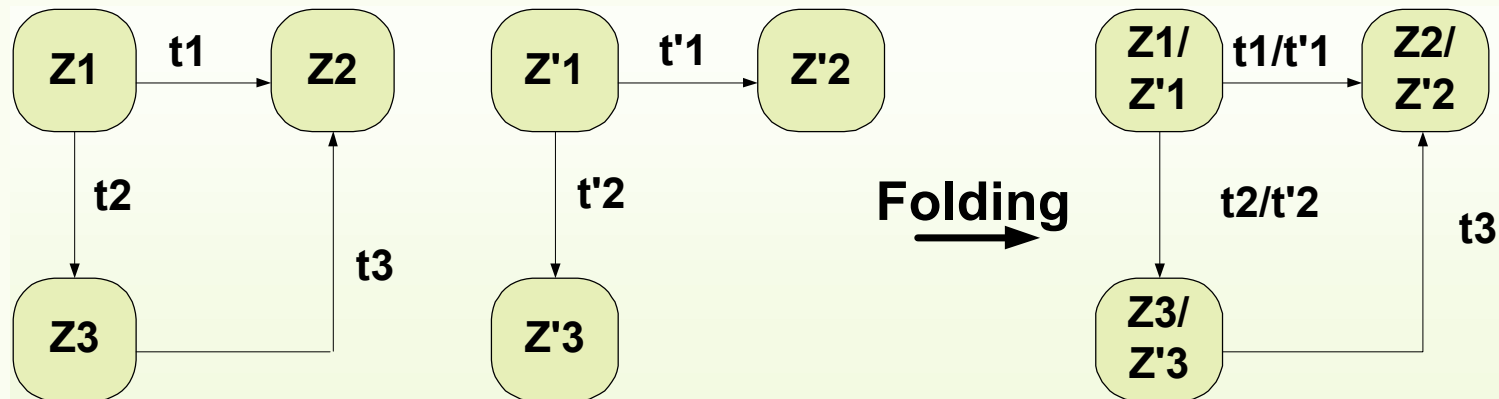- ✓ Compact representation of similar parallel processes or objects

State Charts

➢ **Graphical language for processes and sequences**

➢ **Usable even in early stages of system development**

➢ **Formally defined, textual description available**
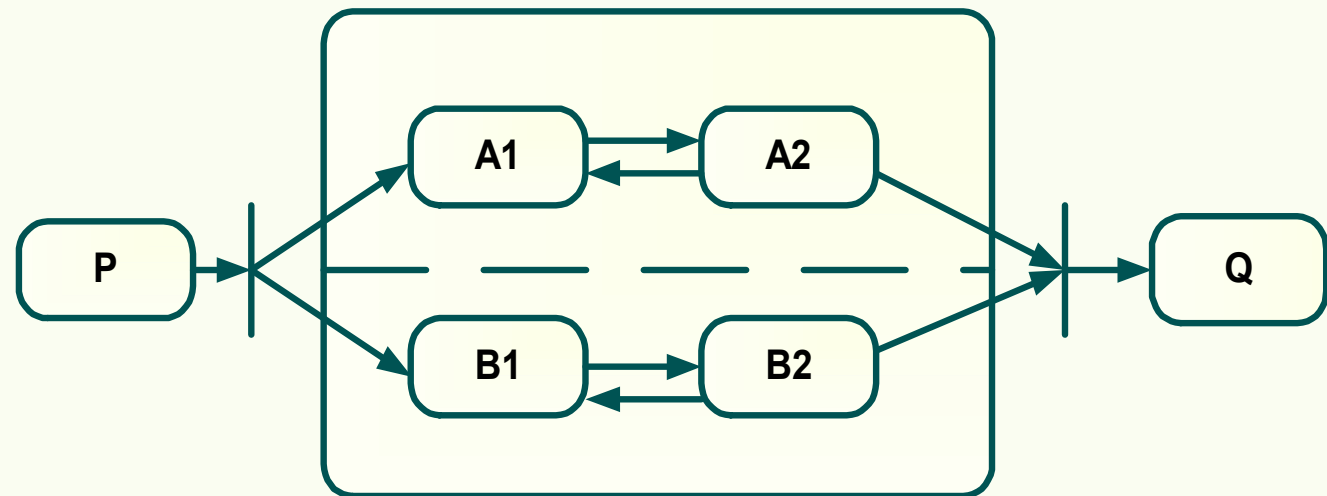
Colored State Charts

„ Folding“

Matching states of similar content and structure

```
Z1  --t1-->  Z2        Z'1  --t'1-->  Z'2          Z1/        --t1/t'1-->   Z2/
 |           ^          |             ^            Z'1                      Z'2
 t2          |          t'2           |             |                       ^
 |           t3         |             |    Folding  t2/t'2                   |
 v           |          v             |    ----->   |                       t3
Z3 ----------+         Z'3            Z'3           Z3/  ---------------------+
                                                    Z'3
```
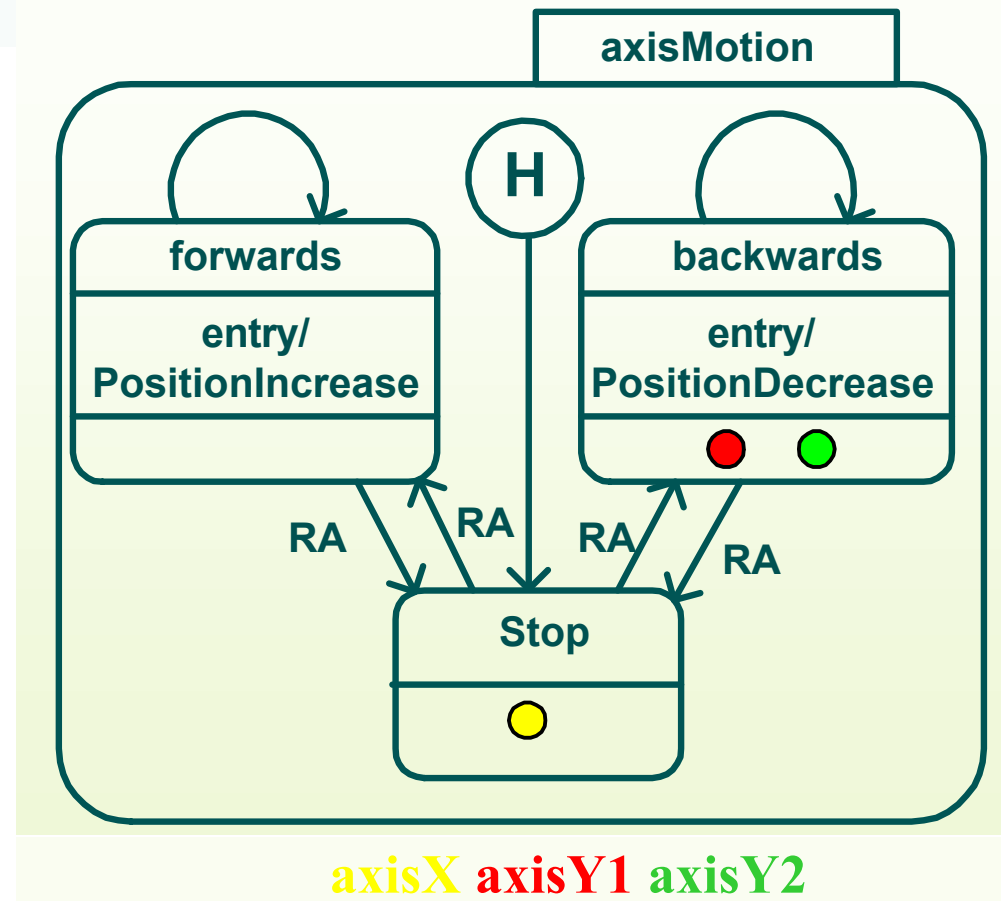
Folding

- some subclasses
- a class and it's subclasses
- some objects of just one class
- some objects of several subclasses

Colored State Charts

- **Example represents three objects from one parental class**
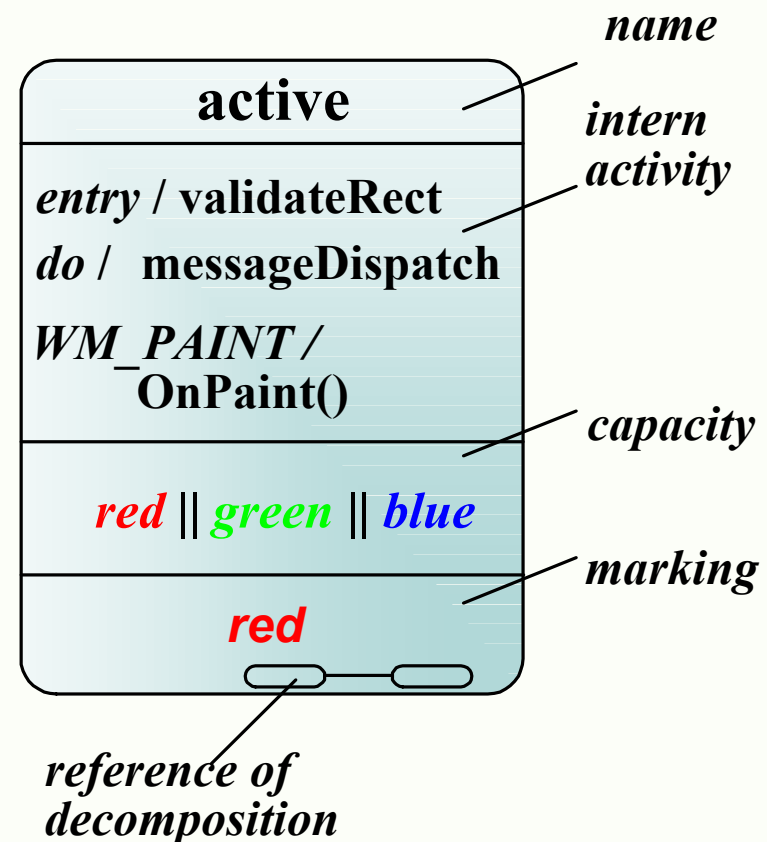
- **There are complex states, events, transitions**

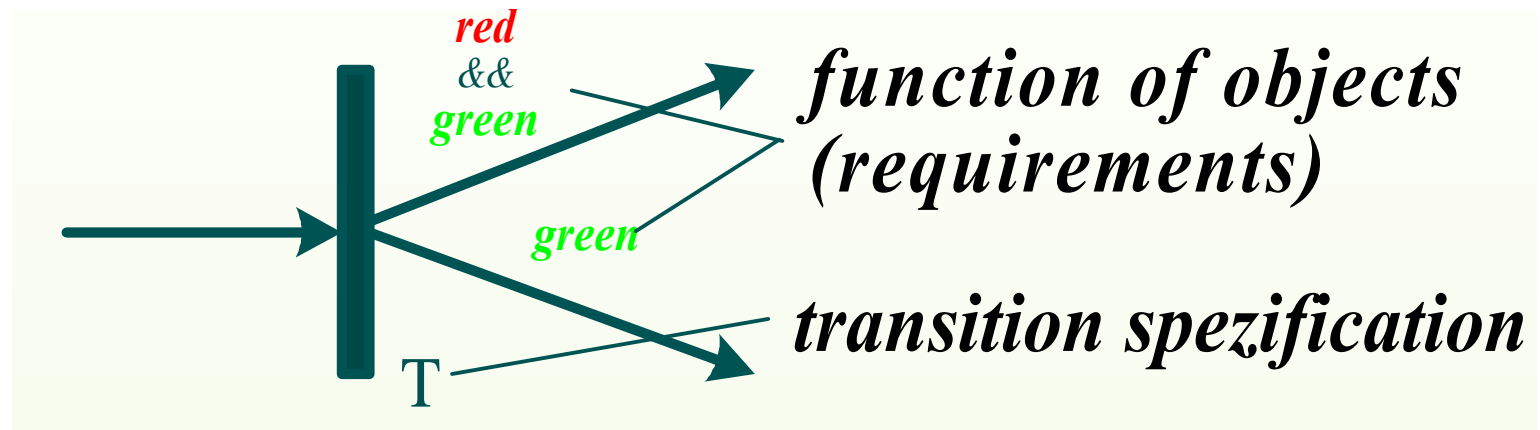- **Activities with time interval properties are available**

axisMotion

H

forwards
entry/
PositionIncrease

backwards
entry/
PositionDecrease

RA   RA   RA   RA

Stop

axisX axisY1 axisY2

Colored State Charts

State

- **Folded from seve**
- **ral local states**
- **Complex capacity**
- **Current marking**

active
entry / **validateRect**   *do* / **messageDispatch**   *WM_PAINT* /   **OnPaint()**
red || green || blue
red

name

intern activity

capacity

marking

reference of decomposition

Colored State Charts

Transitions

event{red && green}[requirement{red}]/action

state1 → state2

red && green → function of objects (requirements)

green → transition spezification

T

Colored State Charts

event1{red && green}/action1

red
green

state1

red && green

event 2{red}/action2

state2

red && green

event 3{green}/action3
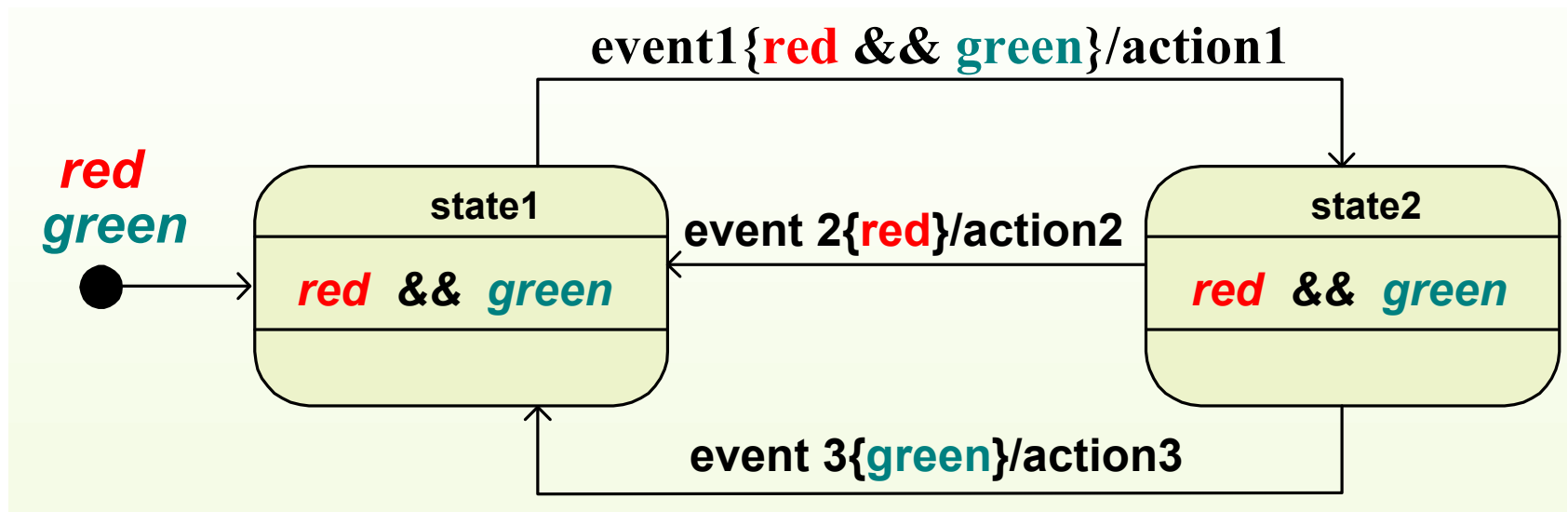
- **Colored states**
- **Complex transitions**

Conversion into High Level Petri Nets

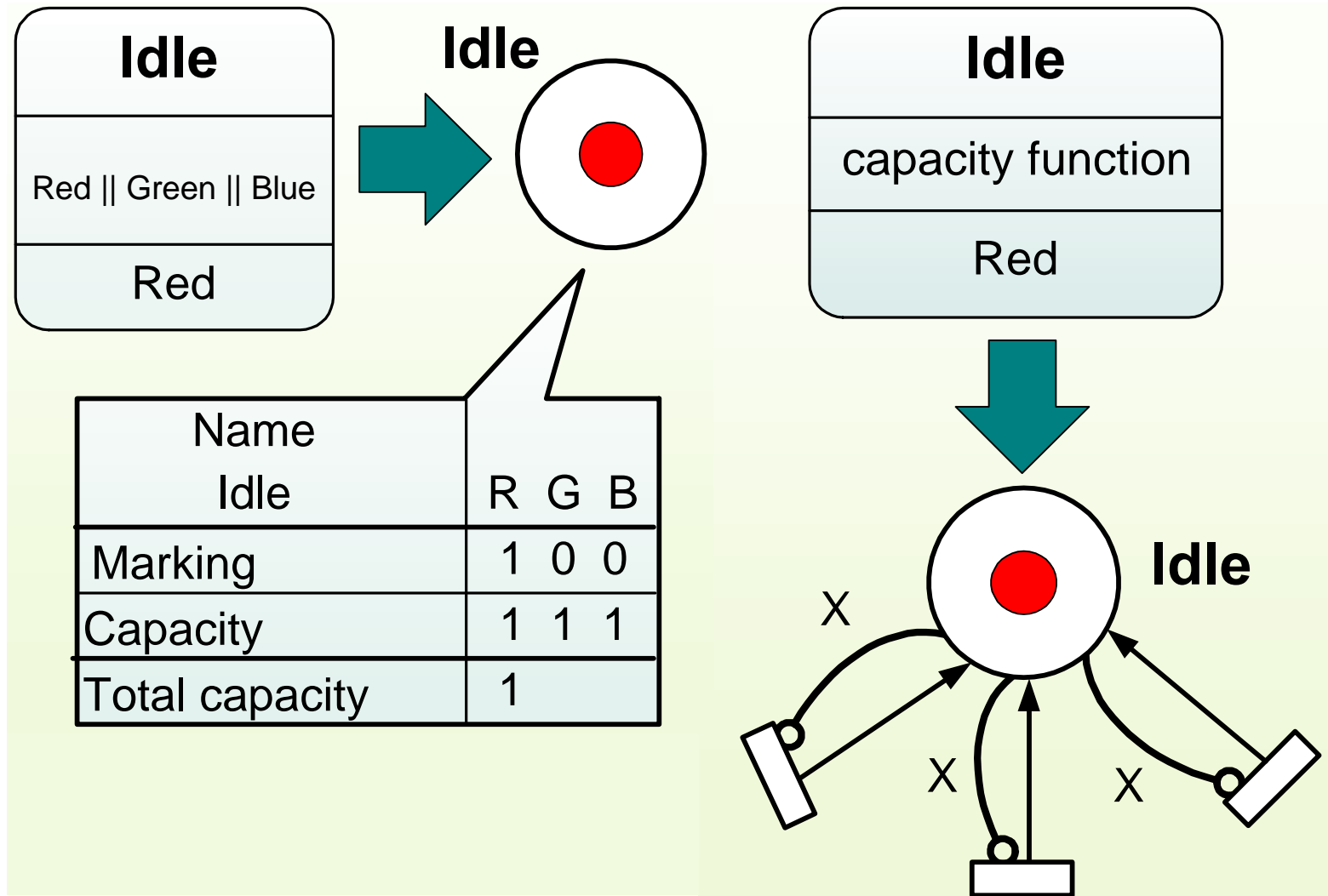## Transformation of the structure:

✓ decomposition into single modules

✓ transformation using existing algorithms

✓ "coloring" the net

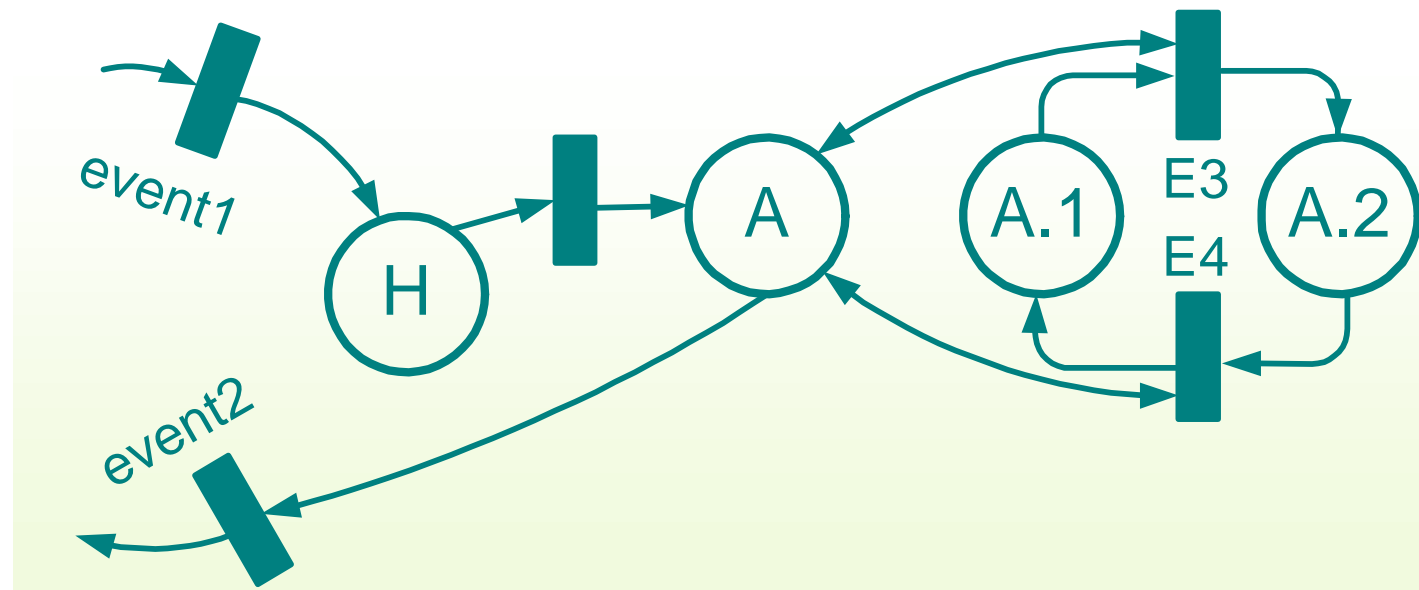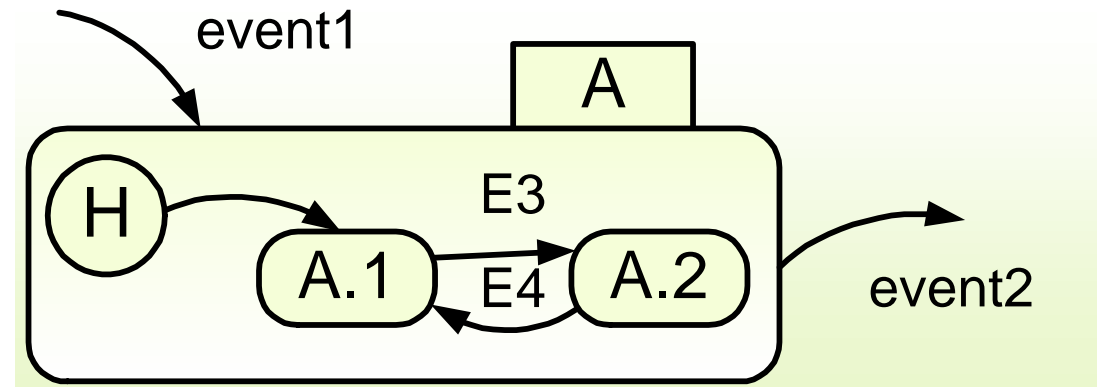✓ transformation of color dependencies

✓ transformation of state capacities

Key aspects of the transformation are:

✓ transformation of the place capacity,

✓ definition of the firing modes for transitions,

✓ transformation of hierarchical structures,

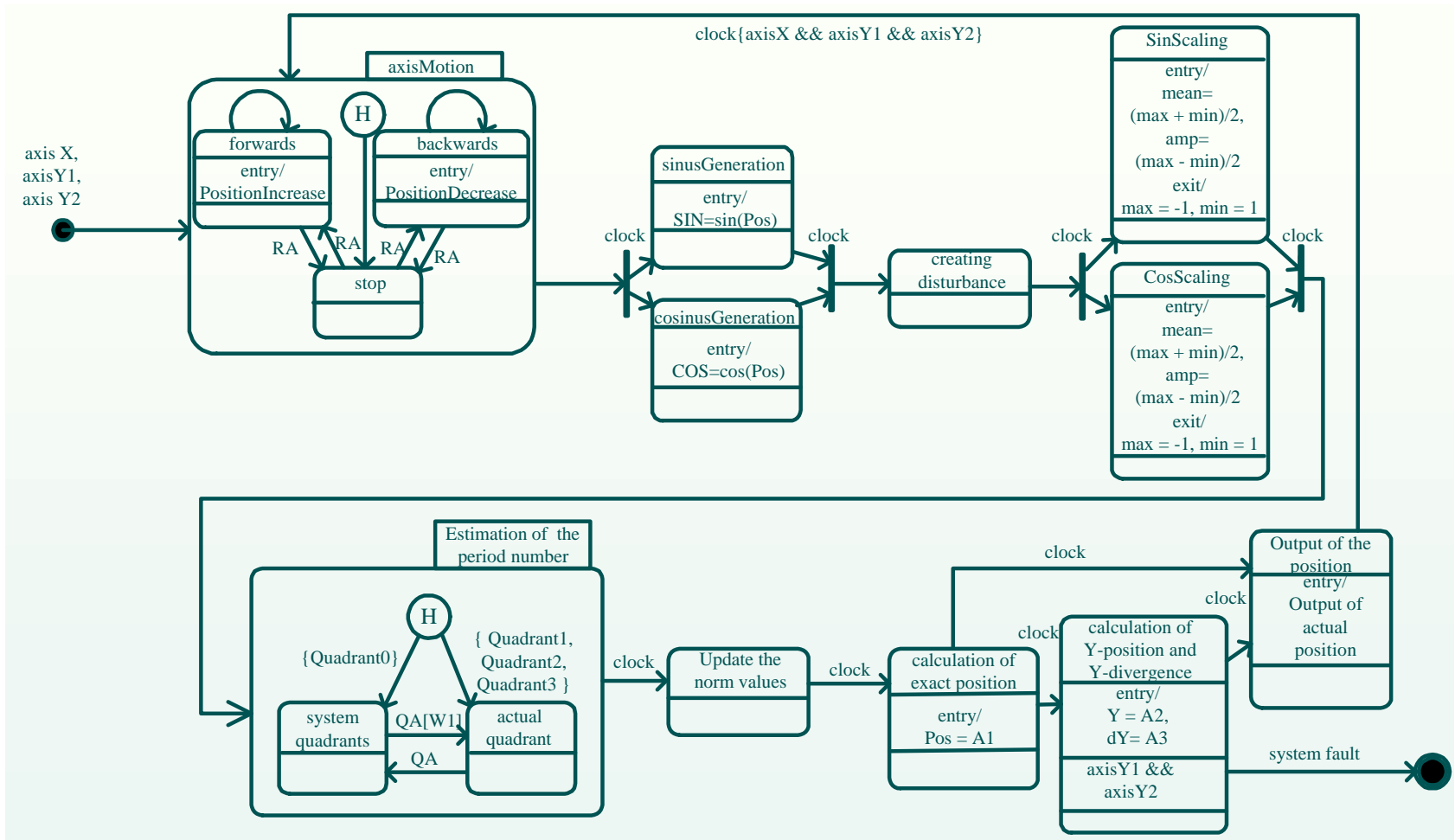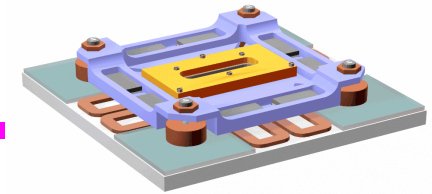✓ transformation of time events and timed transitions
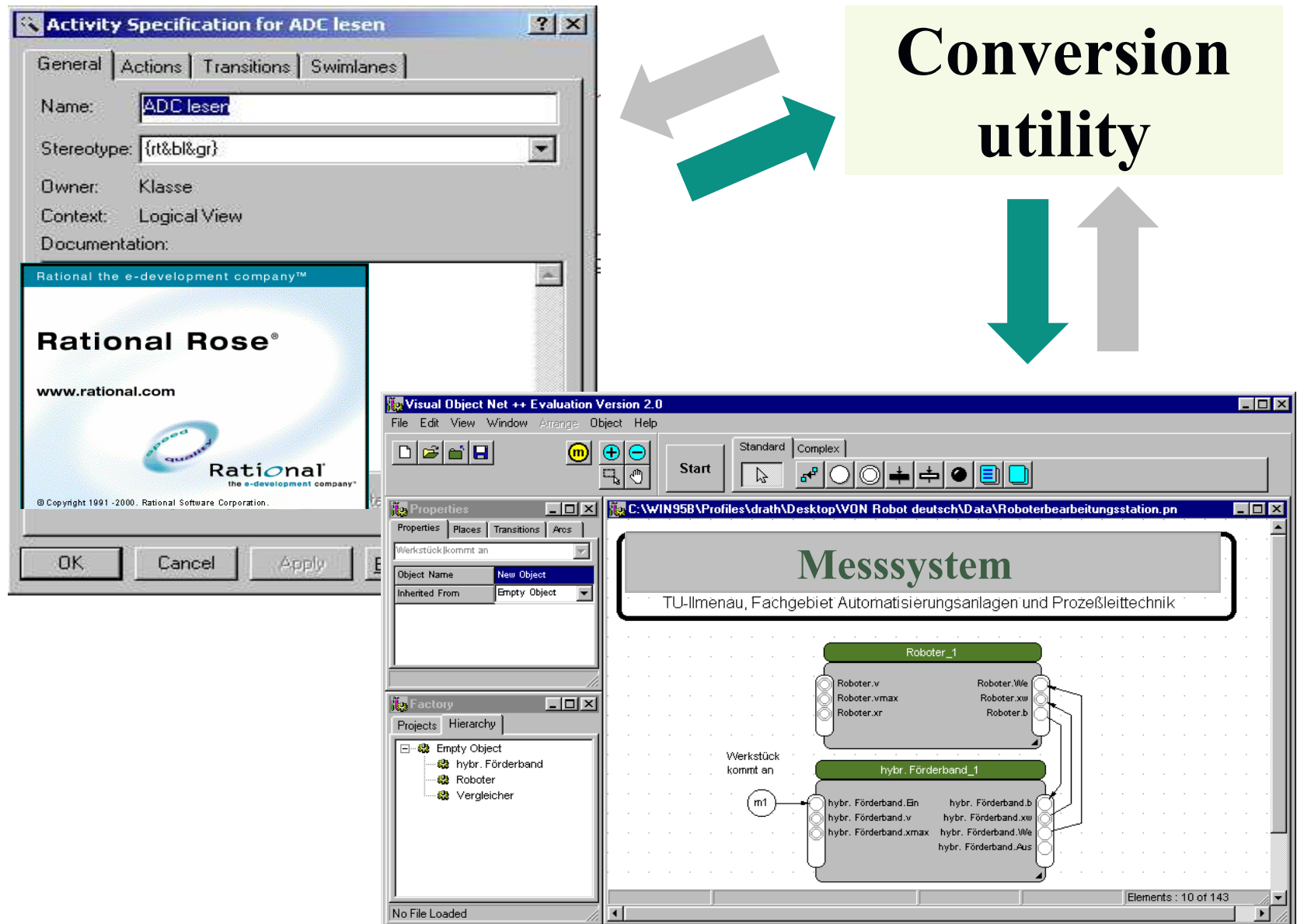
Conversion into High Level Petri Nets

Name Idle	R	G	B
Marking	1	0	0
Capacity	1	1	1
Total capacity	1		

Conversion into High Level Petri Nets

Example Model

clock{axisX && axisY1 && axisY2}

axisMotion

H

forwards
entry/
PositionIncrease

backwards
entry/
PositionDecrease

RA RA RA RA

stop

axis X,
axisY1,
axis Y2

sinusGeneration
entry/
SIN=sin(Pos)

clock

cosinusGeneration
entry/
COS=cos(Pos)

clock

creating disturbance

clock

SinScaling
entry/
mean=
(max + min)/2,
amp=
(max - min)/2
exit/
max = -1, min = 1

clock

CosScaling
entry/
mean=
(max + min)/2,
amp=
(max - min)/2
exit/
max = -1, min = 1

Estimation of the period number

H

{Quadrant0}

{ Quadrant1,
Quadrant2,
Quadrant3 }

system quadrants

QA[W1]

QA

actual quadrant

clock

Update the norm values

clock

calculation of exact position
entry/
Pos = A1

clock

calculation of Y-position and Y-divergence
entry/
Y = A2,
dY= A3

axisY1 &&
axisY2

clock

Output of the position
entry/
Output of actual position

system fault

Ilmenau Technical University • Department of Computer Architectures

TU Ilmenau

13

Tool Development and Adaption

Conversion utility

Conclusion und Perspectives

- ➢ **Compact and clear representation**

- ➢ **Formally defined description**

- ➢ **Available methods for analysis and transformation based on the Petri Net notation**

- ➢ **Available editors for colored diagrams**

- ➢ **Easy-to-use conversion utility**