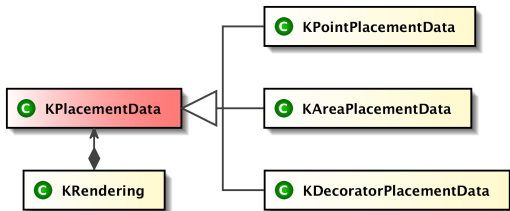


# Transient Node-Link-Diagrams in Practice



Christian Schneider

Institut für Informatik  
Christian-Albrechts-Universität zu Kiel

27. August 2014

# My Vision

No humans but **tools draw diagrams.**

The diagram creation **effort is reduced**  
to the **formulation of**  
model  $\leftrightarrow$  diagram **mappings.**

**Tools care on** all concrete **positionings.**

## What we did achieve 1/2

- ▶ an own view model syntax: *KRendering*
  - ▶ provides descriptive micro layout statements
  - ▶ increases rendering performance (efficient application of automatic layout)
  - ▶ supports advanced stuff, e.g. hierarchy, splines, “host” figures



Ch. Schneider, M. Spönemann & R. v. Hanxleden

Transient View Generation in Eclipse

1st Workshop on Academics Modeling with Eclipse (ACME) at ECMFA '12

## What we did achieve 2/2

- ▶ a rendering implementation of KRendering view models
  - ▶ based on Piccolo2D (SWT)
  - ▶ places edge labels properly
  - ▶ resolves declarative micro layout statements
  - ▶ incorporates incremental update strategies

## What (we're | will be) achieving

- ▶ **currently under development:**
  - ▶ automatic figure size estimation based on text labels
  - ▶ cubic splines
  - ▶ deployment in MENGES
- ▶ **next steps:**
  - ▶ support for diagram options
  - ▶ proper selection support for edges, esp. splines
  - ▶ implementation of a proper incremental update strategy
  - ▶ SyncCharts diagram synthesis
  - ▶ (x|K)Diagram description language
  - ▶ Draw2d-based rendering implementation

## What (we're | will be) achieving

### ▶ currently under development:

- ▶ automatic figure size estimation based on text labels
- ▶ cubic splines
- ▶ deployment in MENGES



### ▶ next steps:

- ▶ support for diagram options
- ▶ proper selection support for edges, esp. splines
- ▶ implementation of a proper incremental update strategy
- ▶ SyncCharts diagram synthesis
- ▶ (x|K)Diagram description language
- ▶ Draw2d-based rendering implementation



## What we (are | will be | should be) achieving

- ▶ enable structure-based editing
  - ▶ prototype available
- ▶ hovering text label widget
  - ▶ copy & paste text labels
  - ▶ change names  $\rightsquigarrow$  structure-based editing
- ▶ improve event handling, esp. selection handling
- ▶ improve selection highlighting
- ▶ fix grid placement strategy
- ▶ realize incremental update with new EMF Compare
- ▶ KDiagram description language

## What we (are | will be | should be) achieving

- ▶ enable structure-based editing
  - ▶ prototype available
- ▶ hovering text label widget
  - ▶ copy & paste text labels
  - ▶ change names  $\rightsquigarrow$  structure-based editing
- ▶ improve event handling, esp. selection handling
- ▶ improve selection highlighting
- ▶ fix grid placement strategy
- ▶ realize incremental update with new EMF Compare
- ▶ **KDiagram description language**





## What we (are | will be | should be) achieving

- ▶ enable structure-based editing
  - ▶ prototype available
- ▶ hovering text label widget
  - ▶ copy & paste text labels
  - ▶ change names  $\rightsquigarrow$  structure-based editing
- ▶ improve event handling, esp. selection handling
- ▶ improve selection highlighting
- ▶ fix grid placement strategy
- ▶ realize incremental update with new EMF Compare
- ▶ KDiagram description language



## What we (are | will be | should be) achieving

- ▶ enable structure-based editing
  - ▶ prototype available
- ▶ hovering text label widget
  - ▶ copy & paste text labels
  - ▶ change names  $\rightsquigarrow$  structure-based editing
- ▶ improve event handling, esp. selection handling
- ▶ improve selection highlighting
- ▶ fix grid placement strategy
- ▶ realize incremental update with new EMF Compare
- ▶ KDiagram description language



## What we (are | will be | should be) achieving

- ▶ enable structure-based editing
  - ▶ prototype available
- ▶ hovering text label widget
  - ▶ copy & paste text labels
  - ▶ change names  $\rightsquigarrow$  structure-based editing
- ▶ improve event handling, esp. selection handling
- ▶ improve selection highlighting
- ▶ fix grid placement strategy
- ▶ realize incremental update with new EMF Compare
- ▶ KDiagram description language



## What we (are | will be | should be) achieving

- ▶ enable structure-based editing
  - ▶ prototype available
- ▶ hovering text label widget
  - ▶ copy & paste text labels
  - ▶ change names  $\rightsquigarrow$  structure-based editing
- ▶ improve event handling, esp. selection handling
- ▶ improve selection highlighting
- ▶ fix grid placement strategy
- ▶ realize incremental update with new EMF Compare
- ▶ KDiagram description language



## What we (are | will be | should be) achieving

- ▶ enable structure-based editing
  - ▶ prototype available
- ▶ hovering text label widget
  - ▶ copy & paste text labels
  - ▶ change names  $\rightsquigarrow$  structure-based editing
- ▶ improve event handling, esp. selection handling
- ▶ improve selection highlighting
- ▶ fix grid placement strategy
- ▶ realize incremental update with new EMF Compare
- ▶ KDiagram description language



## What we (are | will be | should be) achieving

- ▶ enable structure-based editing
  - ▶ prototype available
- ▶ hovering text label widget
  - ▶ copy & paste text labels
  - ▶ change names  $\rightsquigarrow$  structure-based editing
- ▶ improve event handling, esp. selection handling
- ▶ improve selection highlighting
- ▶ fix grid placement strategy
- ▶ realize incremental update with new EMF Compare
- ▶ KDiagram description language



# ETAS EHANDBOOK

## Interactive ECU Documentation







# Before EHandbook

- ▶ Model-based Embedded software engineering of ECUs
- ▶ Software development by *modelers*
- ▶ Deployment by *calibration engineers*
- ▶ Fixed textual documentation containing raster images

Block Diagram Editor for: ControlAlgorithm [ Main ] Project: ControlAlgorithm\_DEFAULT [PC/Physical]

File Edit View Insert Build Extras Tools Window Help

Offline (PC) <Show all> 100%

Tree Pane

Outline Navigation

- self::ControlAlgorithm
  - actual\_position::mesg[cont]
  - of\_Degree2Rad::cont
  - Limiter::Limiter\_1
  - MaxPosition::cont
  - new\_position::mesg[cont]
  - Offset\_pos::cont
  - P\_Gain::cont
  - P\_max::cont
  - PID1::PIDT1
    - T\_i::cont
    - T\_d::cont
    - T\_t::cont
  - target\_pos::mesg[cont]
  - T\_i::cont
- Main
  - normal []
  - out []

This parameter block allows to tune the behavior from extern of the PID controller.

This implementation mainly serves the purpose of showing many features of ASCET - we are well aware that compacting this algorithm is possible on the functional specification level.

Block Diagram Editor for: Limiter\_1 [ Main ] Project: ControlAlgorithm\_DEFAULT [PC/Physical]

File Edit View Insert Build Extras Tools Window Help

Offline (PC) <Show all> 100%

Tree Pane

Outline Navigation

- self::Limiter\_1
  - Main
    - out (mn::cont;pc::cont;mc::cont)

478 @ 53

# Showtime

## Further customers

- ▶ The logo for Sigasi, featuring the word "Sigasi." in a black, sans-serif font. A yellow brushstroke underline is positioned beneath the "i" and extends to the left.

- ▶



Register Now

Home >

## Slick graphical views in Eclipse with the Kieler Lightweight Diagrams Toolkit

This talk presents the [Kieler Lightweight Diagrams \(KlighD\) toolkit](#). This toolkit helps in creating visual representations of your models. In order to create graphical diagrams, you just have to implement a model-to-model transformation that maps your (EMF) models to a uniform and self contained view model. This in turn frees you, as modeling tool builder, from layouting and diagram drawing issues.

In addition to visualization, it supports a lot of high level diagram modification operations like: collapsion, expansion and scaling of single diagram elements, diagram clipping, cropping, zooming, and more. These are important operations for graphical inspection and exploration of models.

### Session details

**Speaker(s):**

Christian Schneider

Hendrik Eeckhaut

[Sigasi]

**Session Type:**

Standard

**Experience level:**

Beginner

**Track:**

Eclipse Technology

# What's next?

- ▶ Write my thesis
  - ▶ Continue writing
  - ▶ Continue writing
  - ▶ ...
- ▶ Flexible overlay labels
- ▶ Reliable diagram export
  - ▶ images
  - ▶ printouts
  - ▶ overlays
  - ▶ tiles
- ▶ KGraph revision??

# What's next?

- ▶ Write my thesis
- ▶ Continue writing
- ▶ Continue writing
- ▶ ...
  
- ▶ Flexible overlay labels
- ▶ Reliable diagram export
  - ▶ images
  - ▶ printouts
  - ▶ overlays
  - ▶ tiles
  
- ▶ KGraph revision??

# What's next?

- ▶ Write my thesis
- ▶ Continue writing
- ▶ Continue writing
- ▶ ...
  
- ▶ Flexible overlay labels
- ▶ Reliable diagram export
  - ▶ images
  - ▶ printouts
  - ▶ overlays
  - ▶ tiles
  
- ▶ KGraph revision??



# What's next?

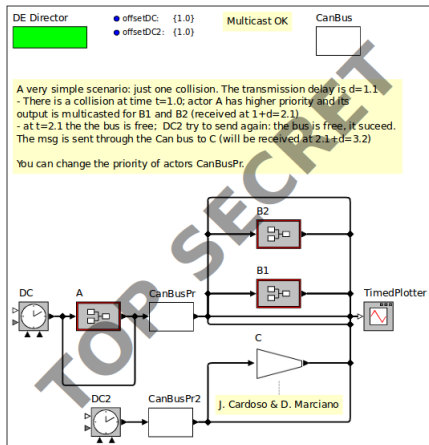
- ▶ Write my thesis
- ▶ Continue writing
- ▶ Continue writing
- ▶ ...
  
- ▶ Flexible overlay labels
- ▶ Reliable diagram export
  - ▶ images
  - ▶ printouts
  - ▶ overlays
  - ▶ tiles
  
- ▶ KGraph revision??

# What's next?

- ▶ Write my thesis
- ▶ Continue writing
- ▶ Continue writing
- ▶ ...
  
- ▶ Flexible overlay labels
- ▶ Reliable diagram export
  - ▶ images
  - ▶ printouts
  - ▶ overlays
  - ▶ tiles
  
- ▶ KGraph revision??

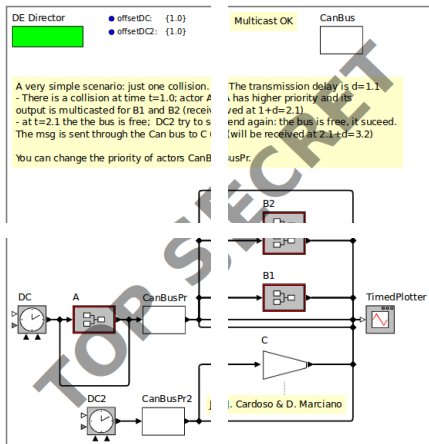
# What's next?

- ▶ Write my thesis
- ▶ Continue writing
- ▶ Continue writing
- ▶ ...
  
- ▶ Flexible overlay labels
- ▶ Reliable diagram export
  - ▶ images
  - ▶ printouts
  - ▶ overlays
  - ▶ tiles
  
- ▶ KGraph revision??



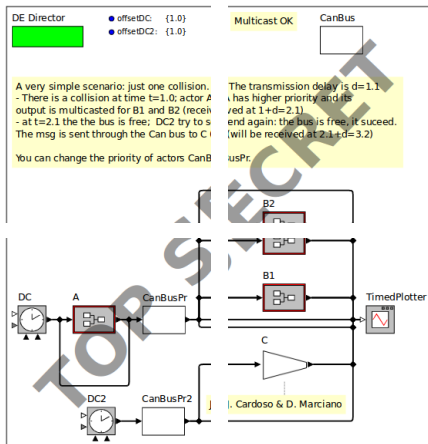
# What's next?

- ▶ Write my thesis
- ▶ Continue writing
- ▶ Continue writing
- ▶ ...
  
- ▶ Flexible overlay labels
- ▶ Reliable diagram export
  - ▶ images
  - ▶ printouts
  - ▶ overlays
  - ▶ tiles
  
- ▶ KGraph revision??



# What's next?

- ▶ Write my thesis
- ▶ Continue writing
- ▶ Continue writing
- ▶ ...
  
- ▶ Flexible overlay labels
- ▶ Reliable diagram export
  - ▶ images
  - ▶ printouts
  - ▶ overlays
  - ▶ tiles
  
- ▶ KGraph revision??



That's it!