Light-Weight Synthesis of Ptolemy Diagrams

with KIELER

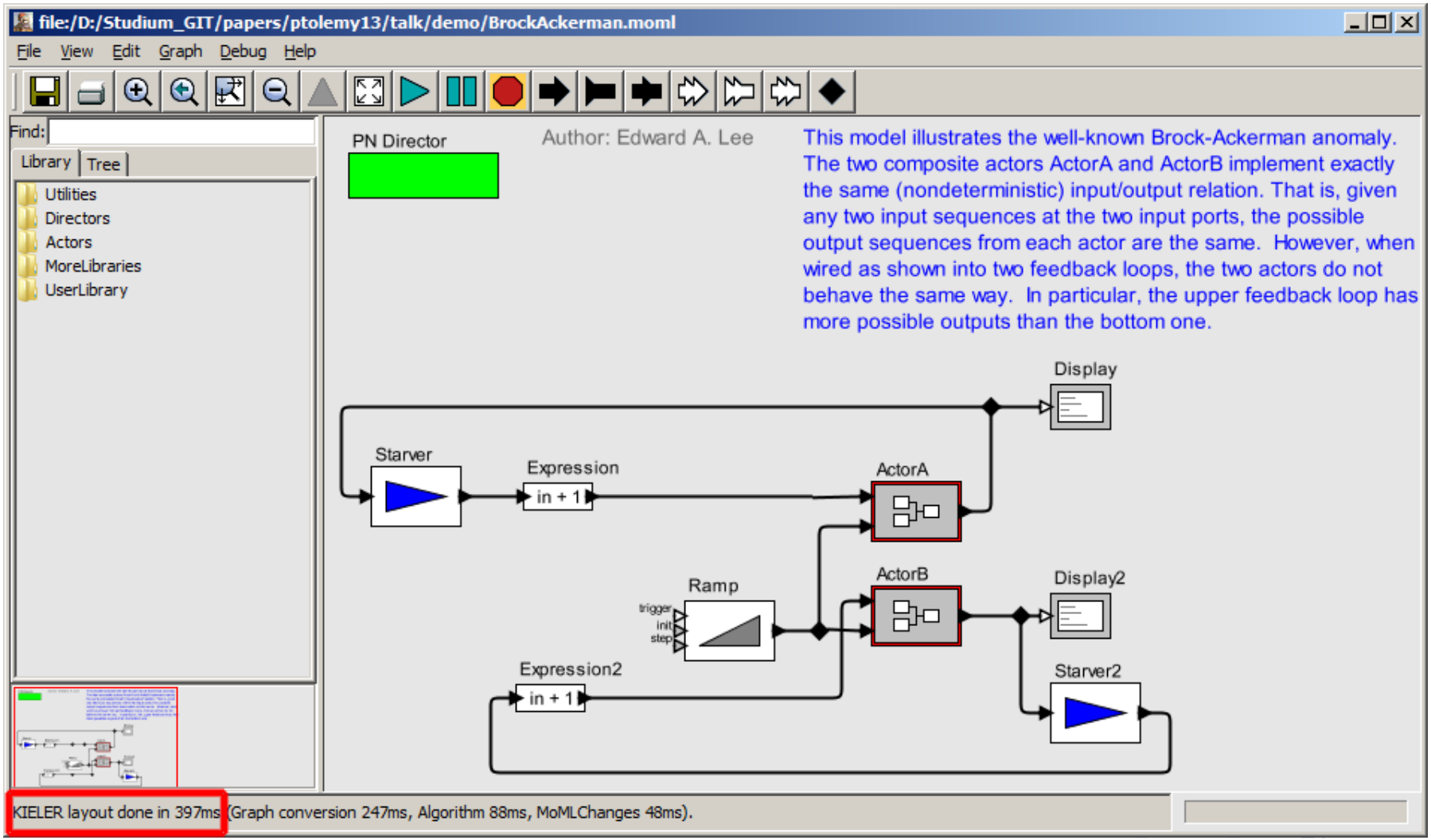
Ulf Rüegg, Christian Schneider, Christoph Daniel Schulze,
Miro Spönemann, Christian Motika, and Reinhard von Hanxleden

Real-Time Systems and Embedded Systems Group
Department of Computer Science
Christian-Albrechts-Universität zu Kiel, Germany

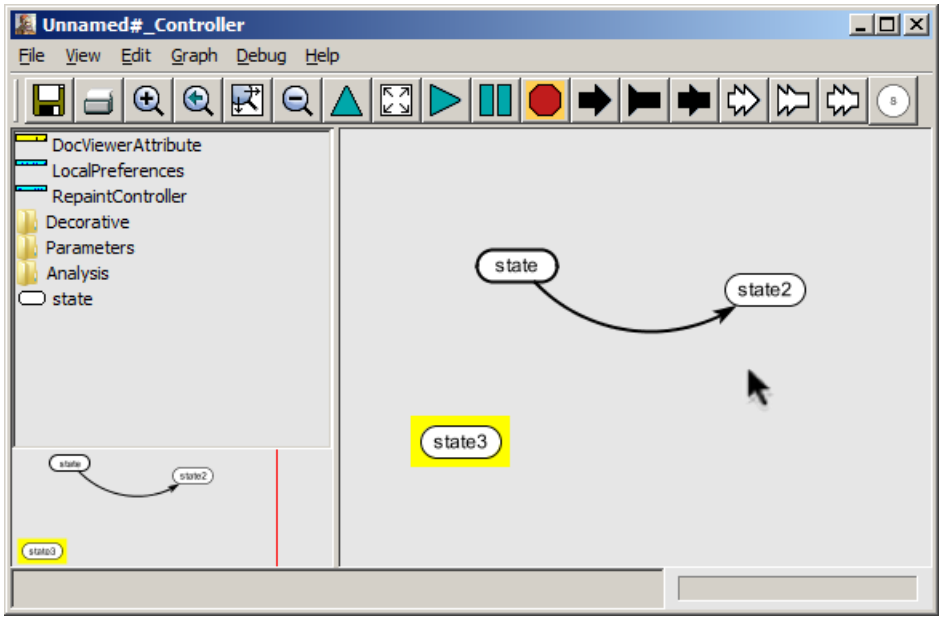
KIELER
Kiel Integrated Environment
for Layout Eclipse RichClient

10th Ptolemy Miniconference
Berkeley, 7 Nov. 2013

Ptolemy & KIELER

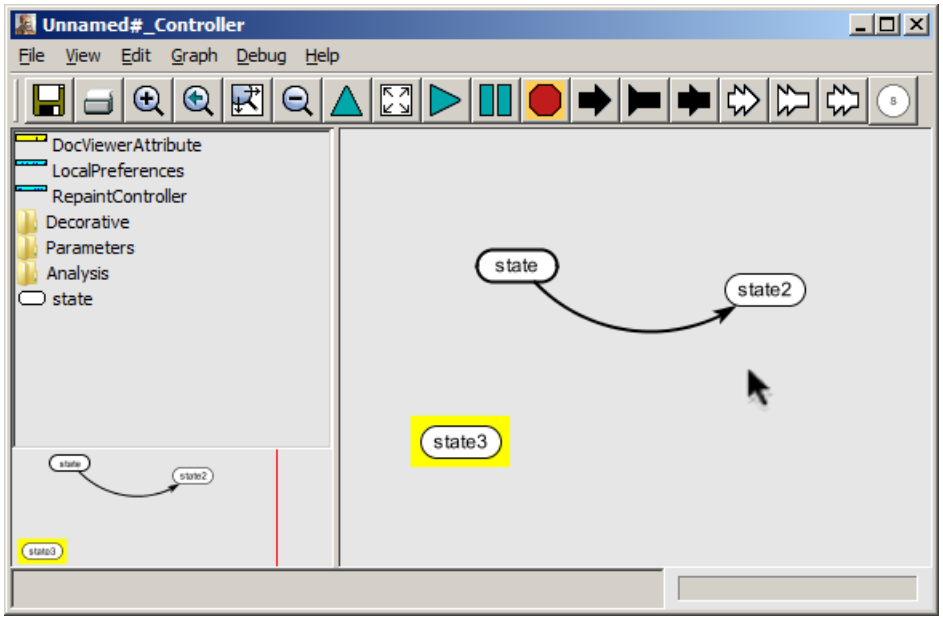


Graphical Modeling



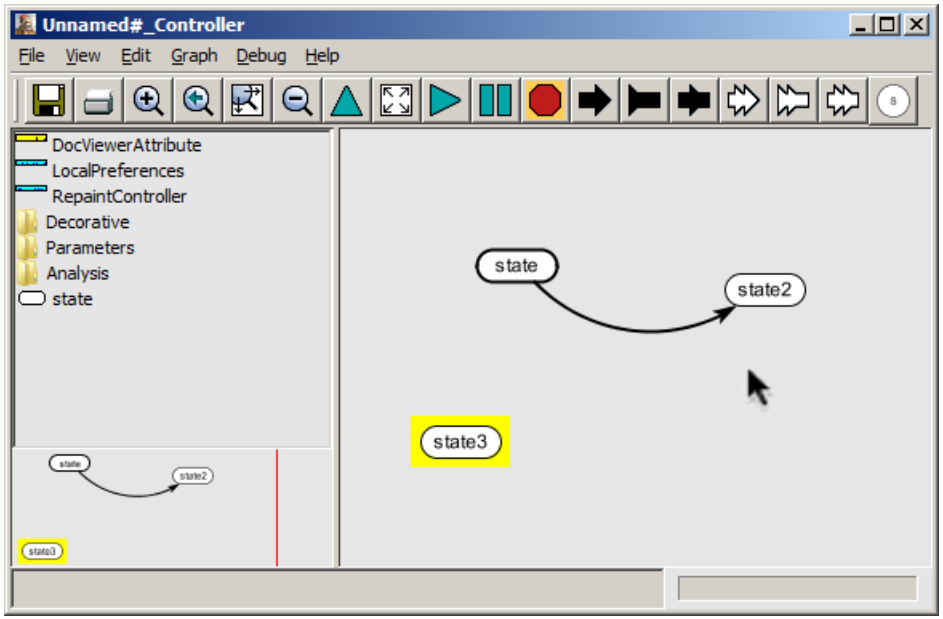
- ☺ Short learning curve (palette, Drag&Drop)

Graphical Modeling



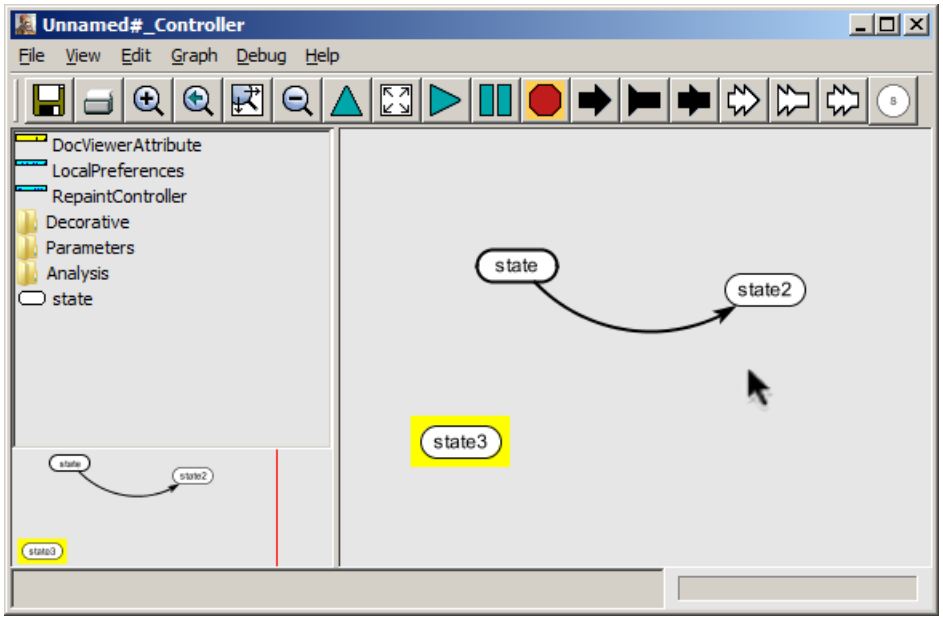
- ☺ Short learning curve (palette, Drag&Drop)
- ☺ Readability (inspecting/understanding, mental map)

Graphical Modeling



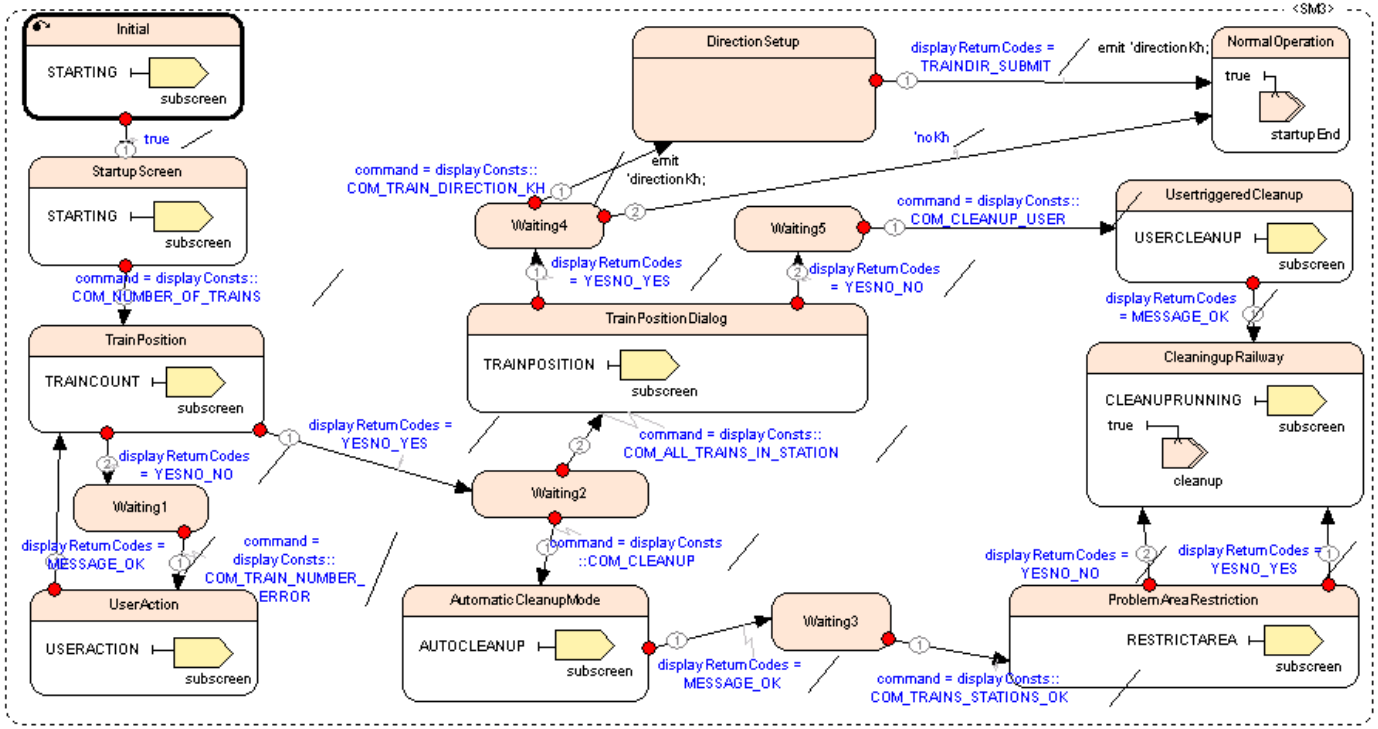
- ☺ Short learning curve (palette, Drag&Drop)
- ☺ Readability (inspecting/understanding, mental map)
- ☺ Visualization of dynamics: Simulation

Graphical Modeling



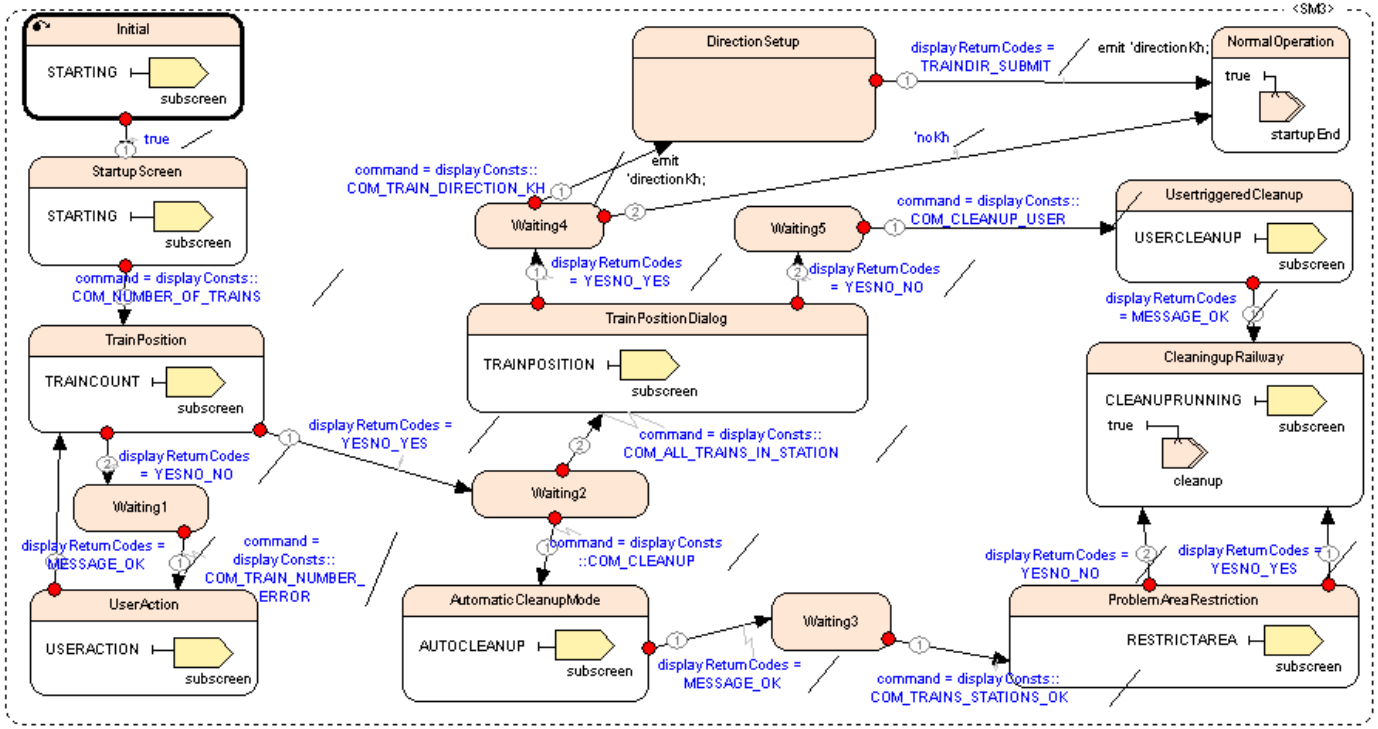
- ☺ Short learning curve (palette, Drag&Drop)
- ☺ Readability (inspecting/understanding, mental map)
- ☺ Visualization of dynamics: Simulation
- ☺ Validation (detect obvious model errors)

Graphical Modeling (cont'd)



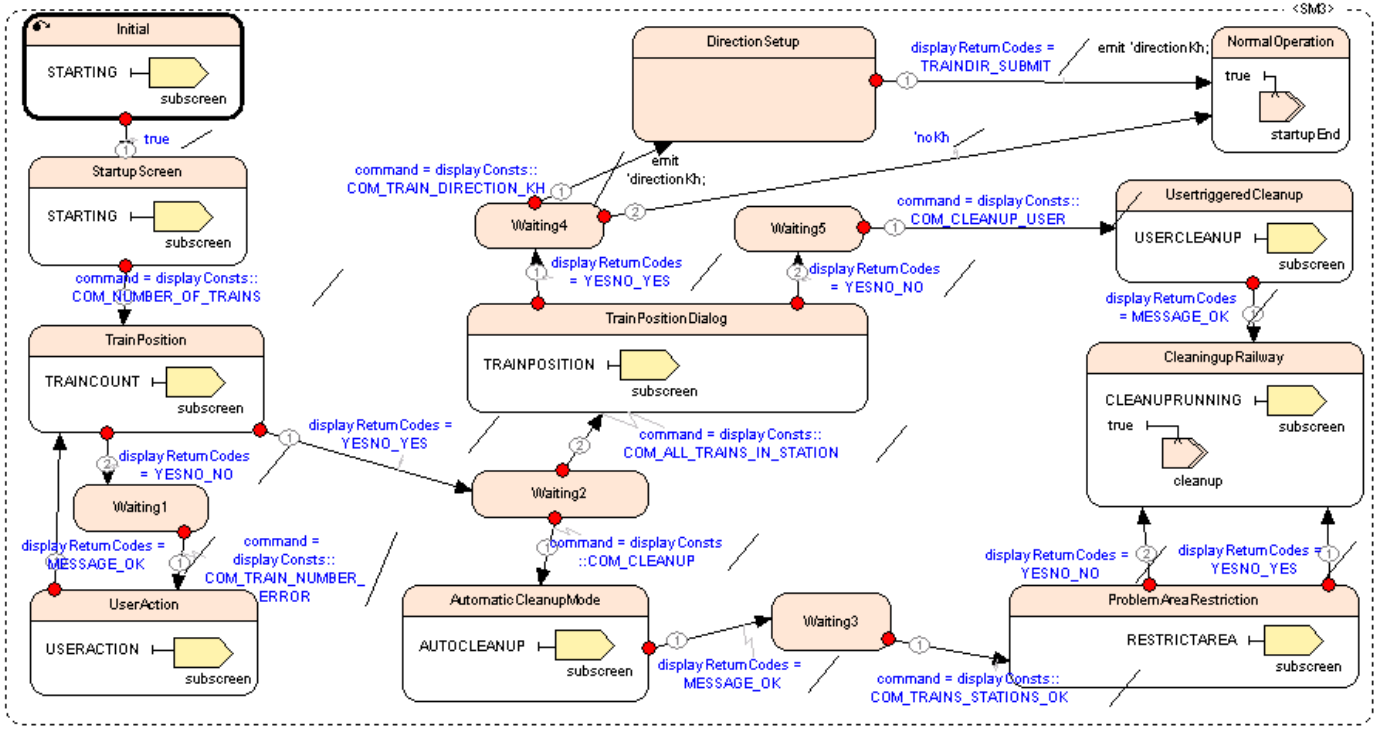
- Widely used in today's industrial tool chains (e.g., SCADE) and academia (e.g., Ptolemy)

Graphical Modeling (cont'd)



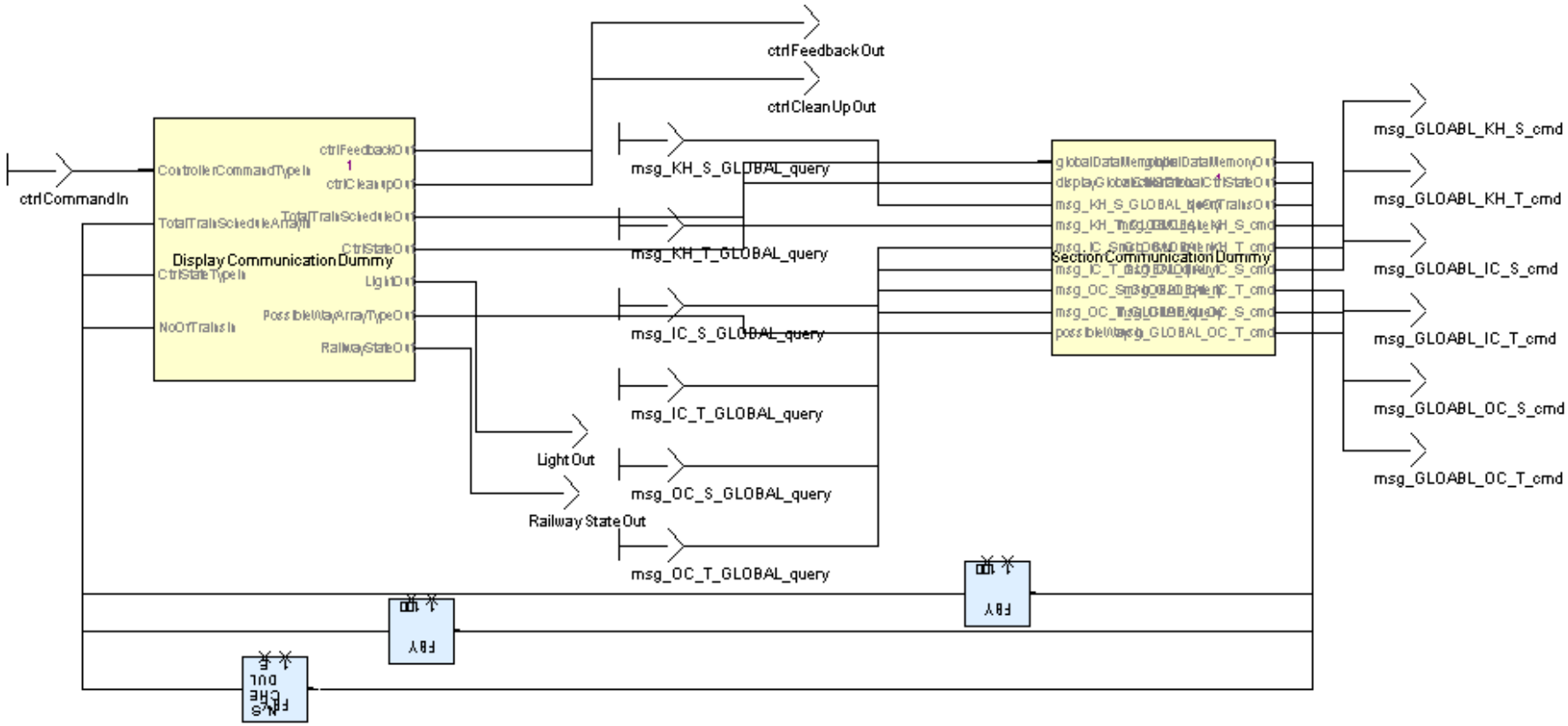
- Widely used in today's industrial tool chains (e.g., SCADE) and academia (e.g., Ptolemy)
- ☹ Readability (overview gets lost quickly)

Graphical Modeling (cont'd)



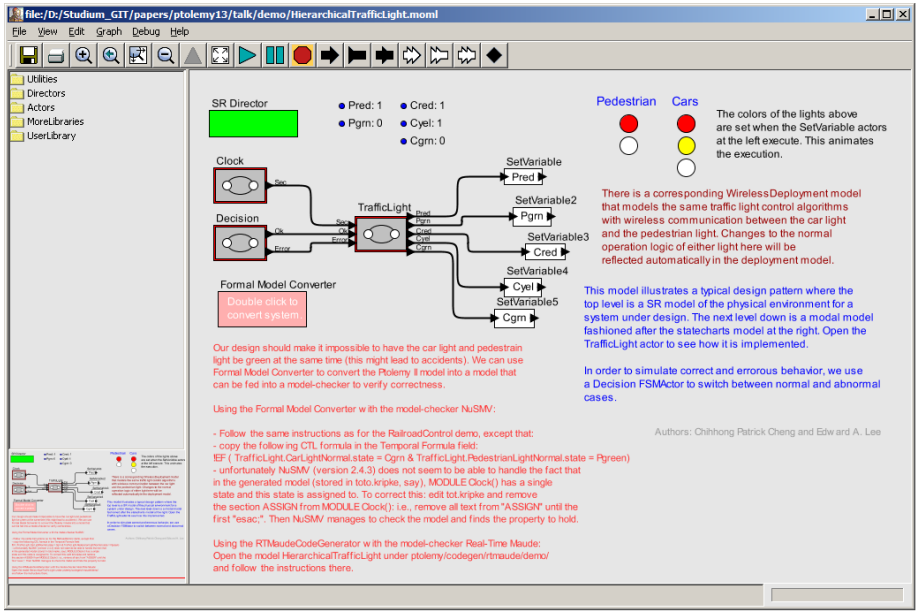
- Widely used in today's industrial tool chains (e.g., SCADE) and academia (e.g., Ptolemy)
- ☹ Readability (overview gets lost quickly)
- ☹ Maintenance (requires lots of manual effort)

Graphical Modeling (cont'd)

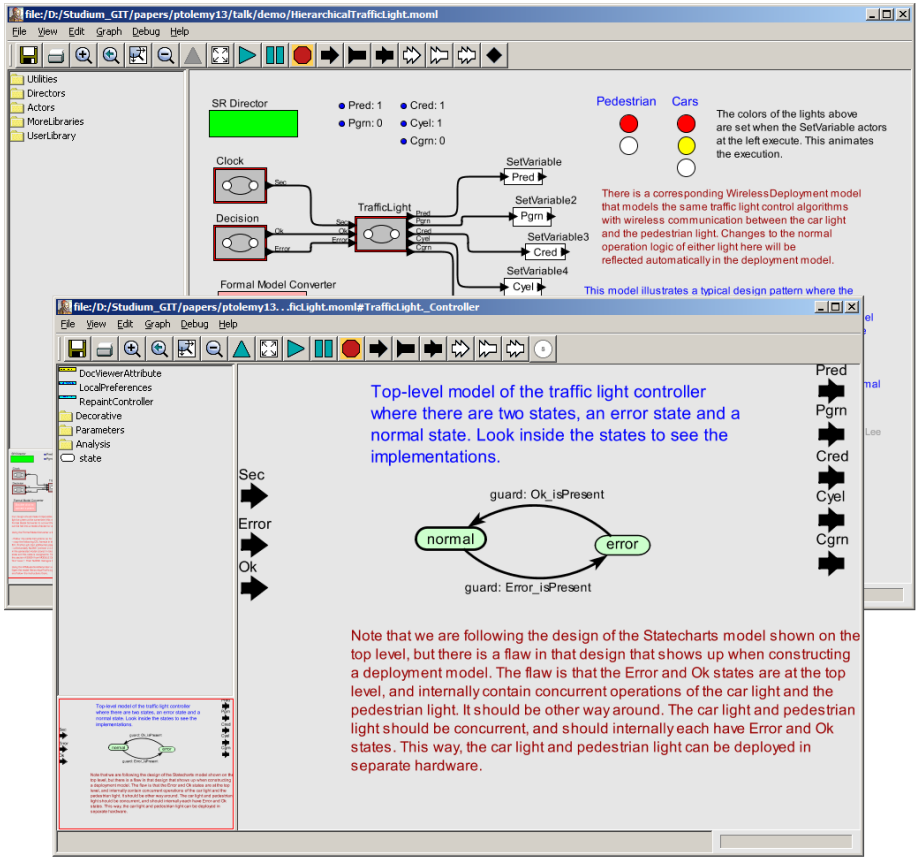


- ☹ Readability (layout critical for understanding semantics)

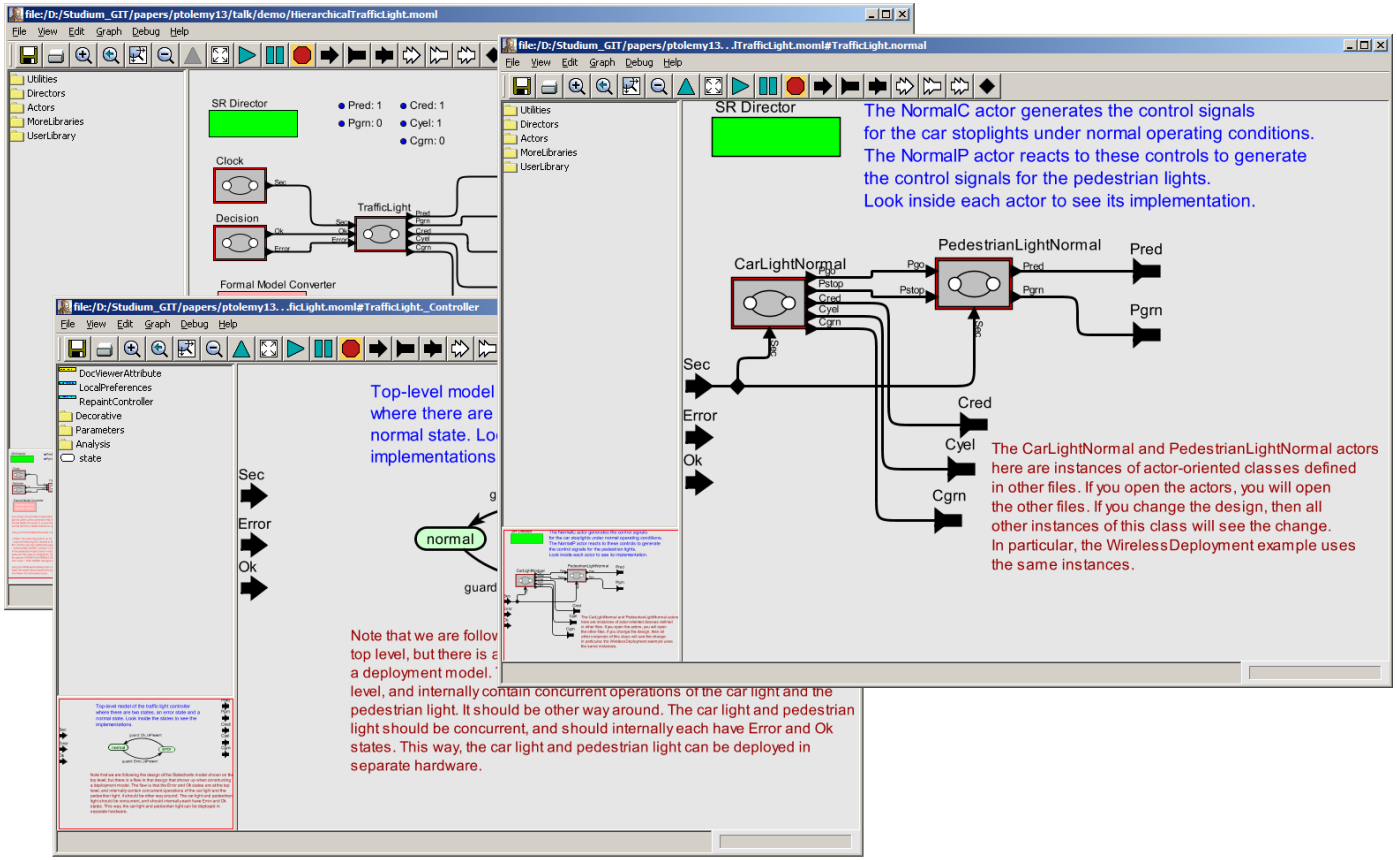
Graphical Modeling (cont'd) - Model Browsing



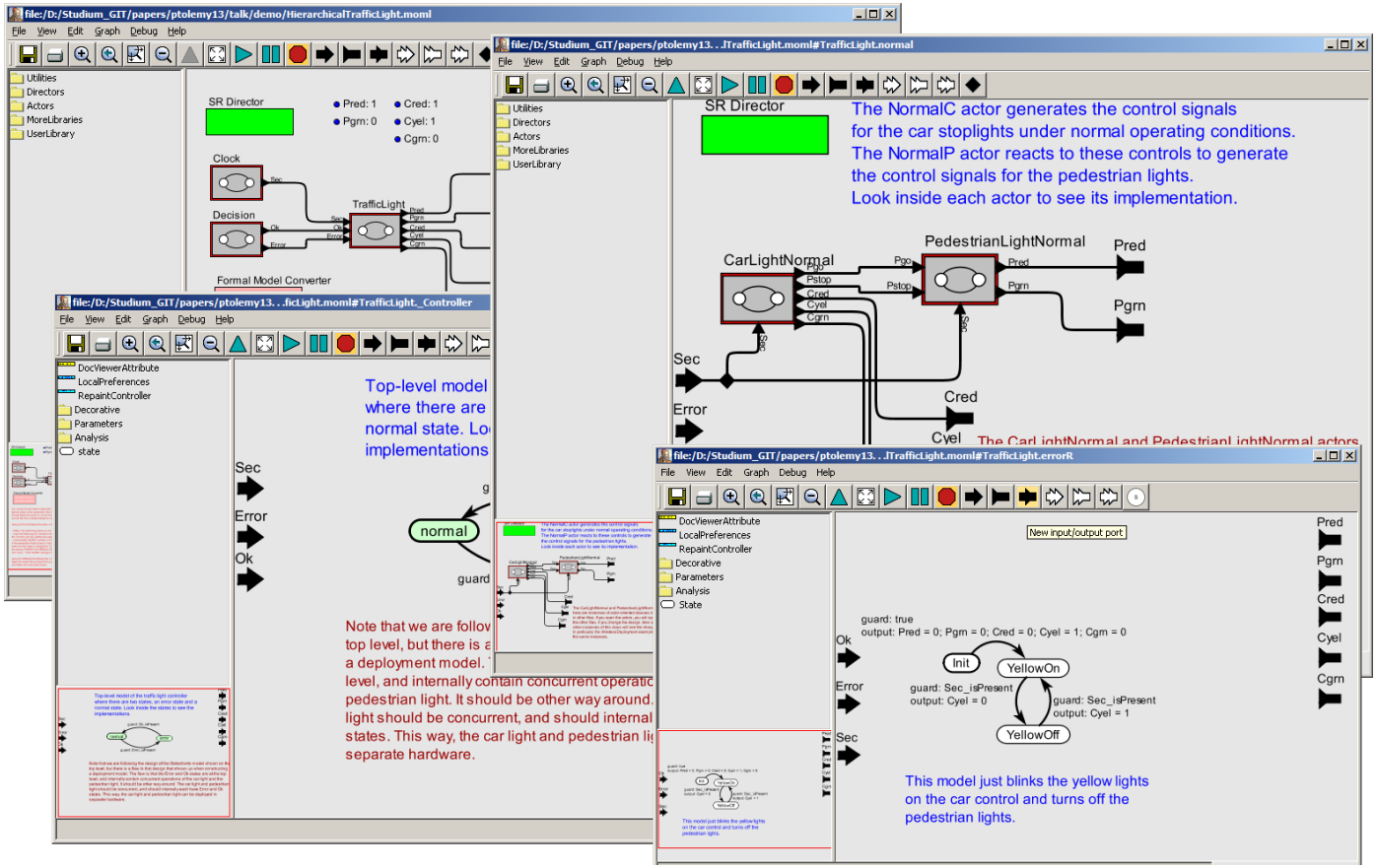
Graphical Modeling (cont'd) - Model Browsing



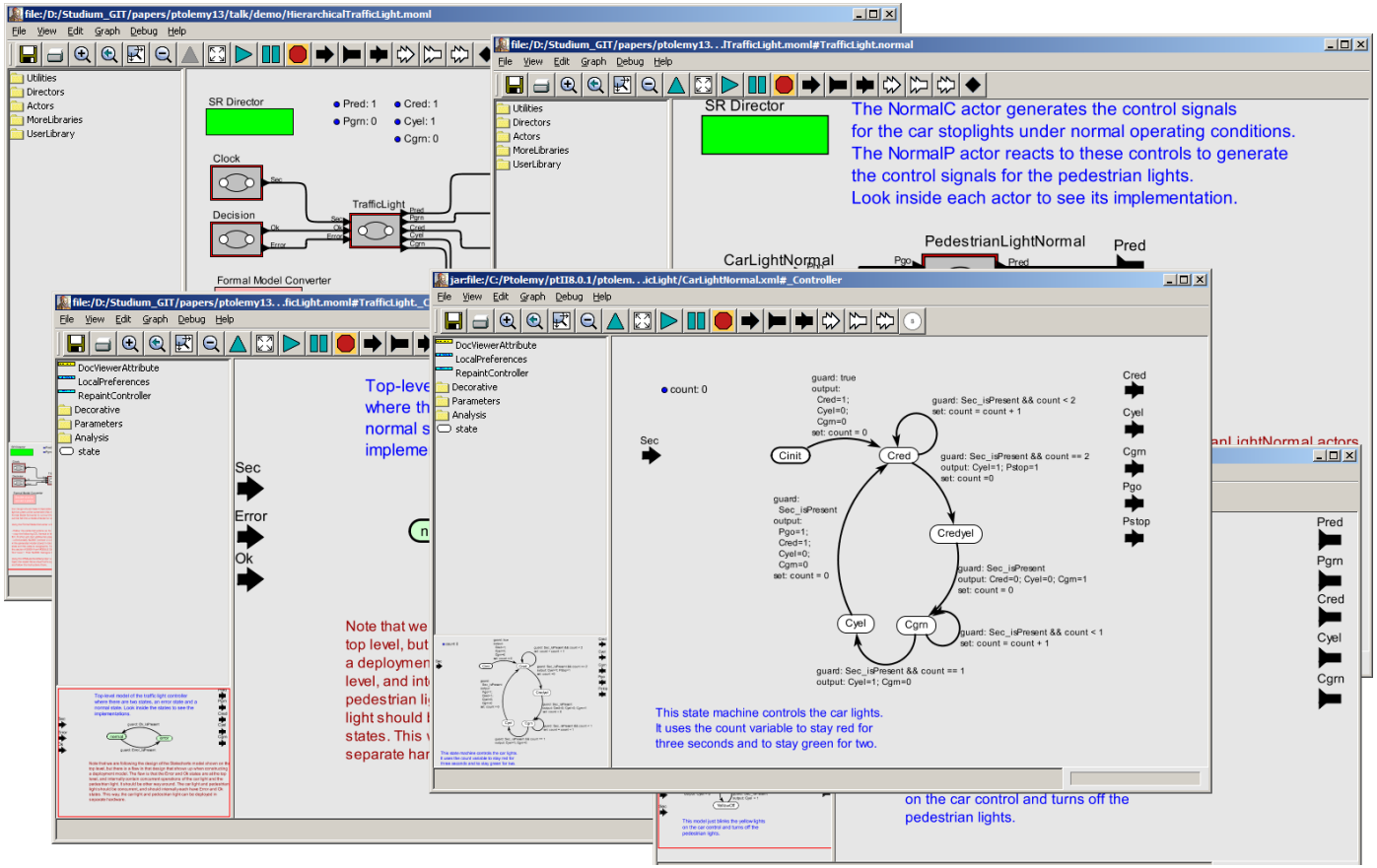
Graphical Modeling (cont'd) - Model Browsing



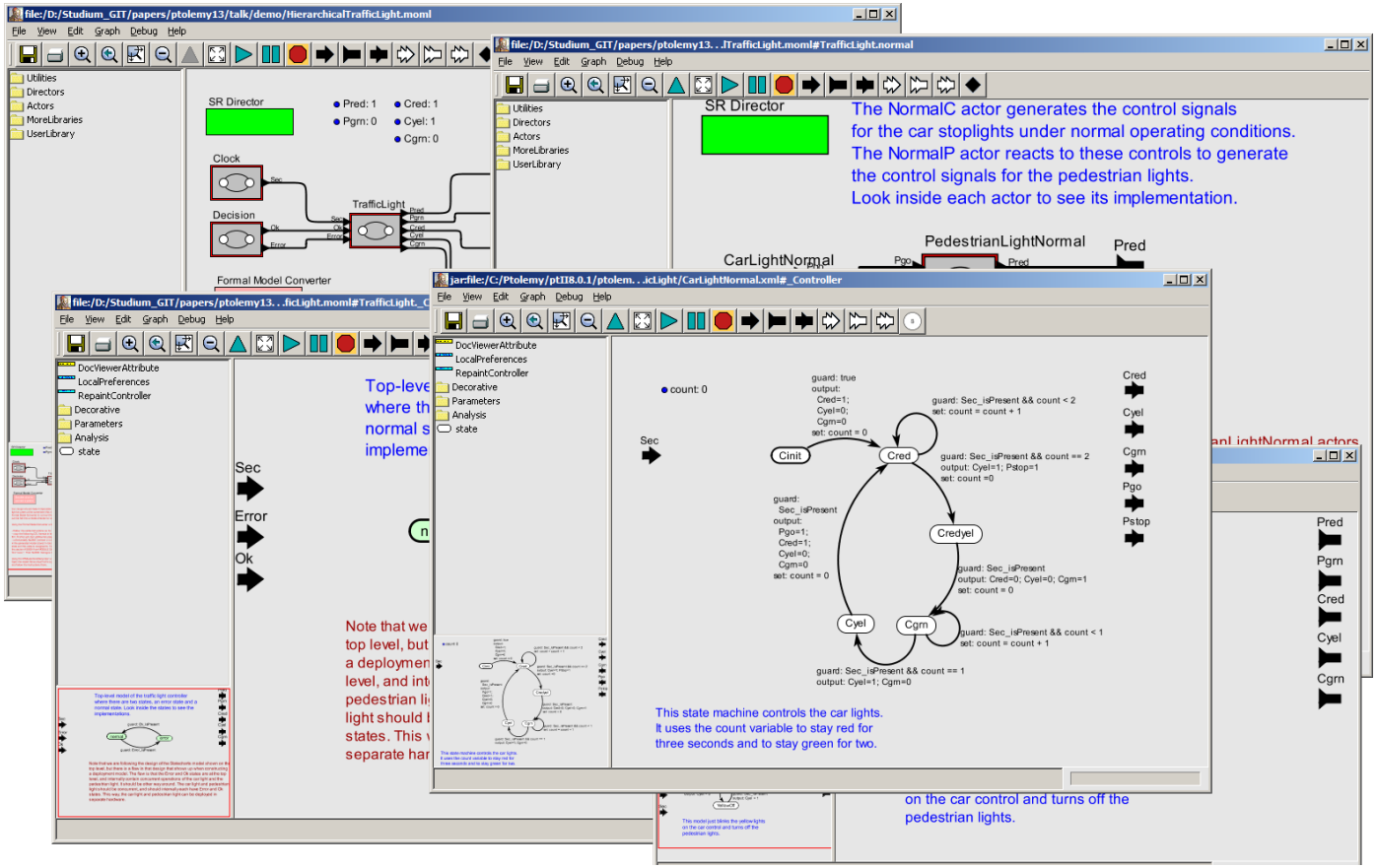
Graphical Modeling (cont'd) - Model Browsing



Graphical Modeling (cont'd) - Model Browsing



Graphical Modeling (cont'd) - Model Browsing



- ☹ Fokus&Context, inner and outer ports vs. performance

Graphical Modeling vs. Textual Modeling

scchart.sct

```
scchart SCChart_Example {
  input signal I;
  signal S;
  output signal O;

  initial state I
  --> M with I;

  state M {

    region:

    initial state C
    --> D immediate with / S;

    final state D;

    region:

    initial state A
    --> B immediate with S;

    final state B;
  }
  >-> E;

  connector state E
  --> F immediate with / O;

  final state F;
}
```

- Emerging trend: *Textual Modeling*
  - Concrete syntax is text

Graphical Modeling vs. Textual Modeling

scchart.sct

```
scchart SCChart_Example {
  input signal I;
  signal S;
  output signal O;

  initial state I
  --> M with I;

  state M {

    region:

    initial state C
    --> D immediate with / S;

    final state D;

    region:

    initial state A
    --> B immediate with S;

    final state B;
  }
  >-> E;

  connector state E
  --> F immediate with / O;

  final state F;
}
```

- Emerging trend: *Textual Modeling*
  - Concrete syntax is text
  - ☺ Rapid Development (good tooling support, e.g., Xtext)

Graphical Modeling vs. Textual Modeling

scchart.sct

```
scchart SCChart_Example {
  input signal I;
  signal S;
  output signal O;

  initial state I
  --> M with I;

  state M {

    region:

    initial state C
    --> D immediate with / S;

    final state D;

    region:

    initial state A
    --> B immediate with S;

    final state B;
  }
  >-> E;

  connector state E
  --> F immediate with / O;

  final state F;
}
```

- Emerging trend: *Textual Modeling*
  - Concrete syntax is text
  - ☺ Rapid Development (good tooling support, e.g., Xtext)
  - ☺ Handling (model diffs, version control, tool interchange)

Graphical Modeling vs. Textual Modeling

scchart.sct

```
scchart SCChart_Example {
  input signal I;
  signal S;
  output signal O;

  initial state I
  --> M with I;

  state M {

    region:

    initial state C
    --> D immediate with / S;

    final state D;

    region:

    initial state A
    --> B immediate with S;

    final state B;
  }
  >-> E;

  connector state E
  --> F immediate with / O;

  final state F;
}
```

- Emerging trend: *Textual Modeling*
  - Concrete syntax is text
  - ☺ Rapid Development (good tooling support, e.g., Xtext)
  - ☺ Handling (model diffs, version control, tool interchange)
  - ☺ Readability, Maintenance (formatter)

Graphical Modeling vs. Textual Modeling

scchart.sct

```
scchart SCChart_Example {
  input signal I;
  signal S;
  output signal O;

  initial state I
  --> M with I;

  state M {

    region:

    initial state C
    --> D immediate with / S;

    final state D;

    region:

    initial state A
    --> B immediate with S;

    final state B;
  }
  >-> E;

  connector state E
  --> F immediate with / O;

  final state F;
}
```

- Emerging trend: *Textual Modeling*
  - Concrete syntax is text
  - ☺ Rapid Development (good tooling support, e.g., Xtext)
  - ☺ Handling (model diffs, version control, tool interchange)
  - ☺ Readability, Maintenance (formatter)
  - ☹ Readability (harder to inspect a larger model, mental map)

Graphical Modeling vs. Textual Modeling

scchart.sct

```
scchart SCChart_Example {
  input signal I;
  signal S;
  output signal O;

  initial state I
  --> M with I;

  state M {

    region:

    initial state C
    --> D immediate with / S;

    final state D;

    region:

    initial state A
    --> B immediate with S;

    final state B;
  }
  >-> E;

  connector state E
  --> F immediate with / O;

  final state F;
}
```

- Emerging trend: *Textual Modeling*
  - Concrete syntax is text
  - ☺ Rapid Development (good tooling support, e.g., Xtext)
  - ☺ Handling (model diffs, version control, tool interchange)
  - ☺ Readability, Maintenance (formatter)
  - ☹ Readability (harder to inspect a larger model, mental map)
  - ☹ Validation (harder to see model errors)

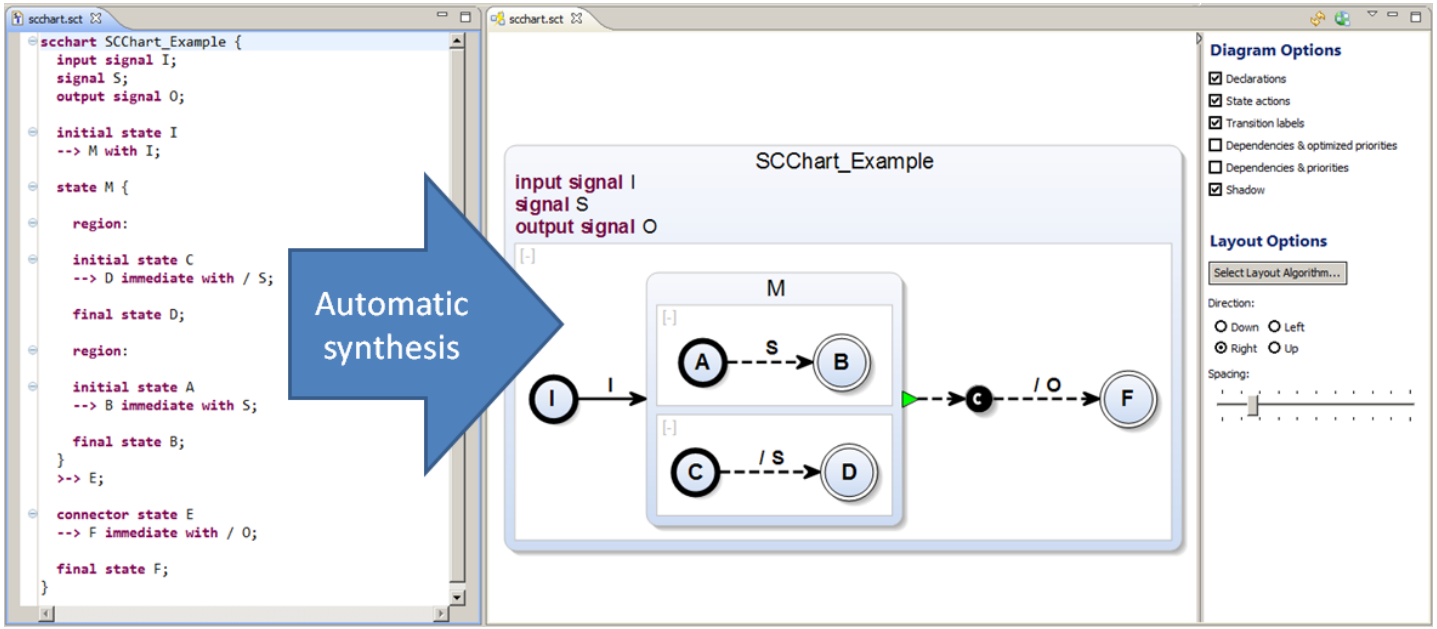
Graphical Modeling vs. Textual Modeling

scchart.sct

```
scchart SCChart_Example {
  input signal I;
  signal S;
  output signal O;

  initial state I
  --> M with I;

  state M {

    region:

    initial state C
    --> D immediate with / S;

    final state D;

    region:

    initial state A
    --> B immediate with S;

    final state B;
  }
  >-> E;

  connector state E
  --> F immediate with / O;

  final state F;
}
```

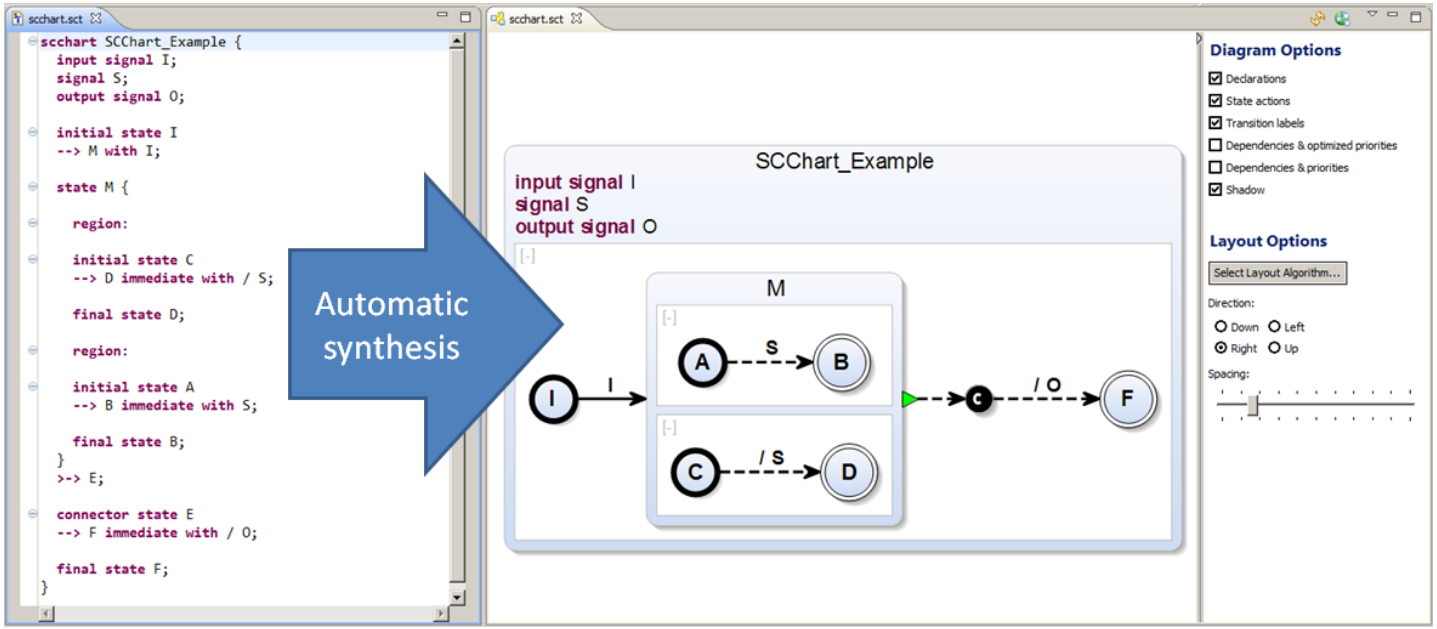
- Emerging trend: *Textual Modeling*
  - Concrete syntax is text
  - ☺ Rapid Development (good tooling support, e.g., Xtext)
  - ☺ Handling (model diffs, version control, tool interchange)
  - ☺ Readability, Maintenance (formatter)
  - ☹ Readability (harder to inspect a larger model, mental map)
  - ☹ Validation (harder to see model errors)
  - ☹ Visualization of dynamics: Simulation

Contribution



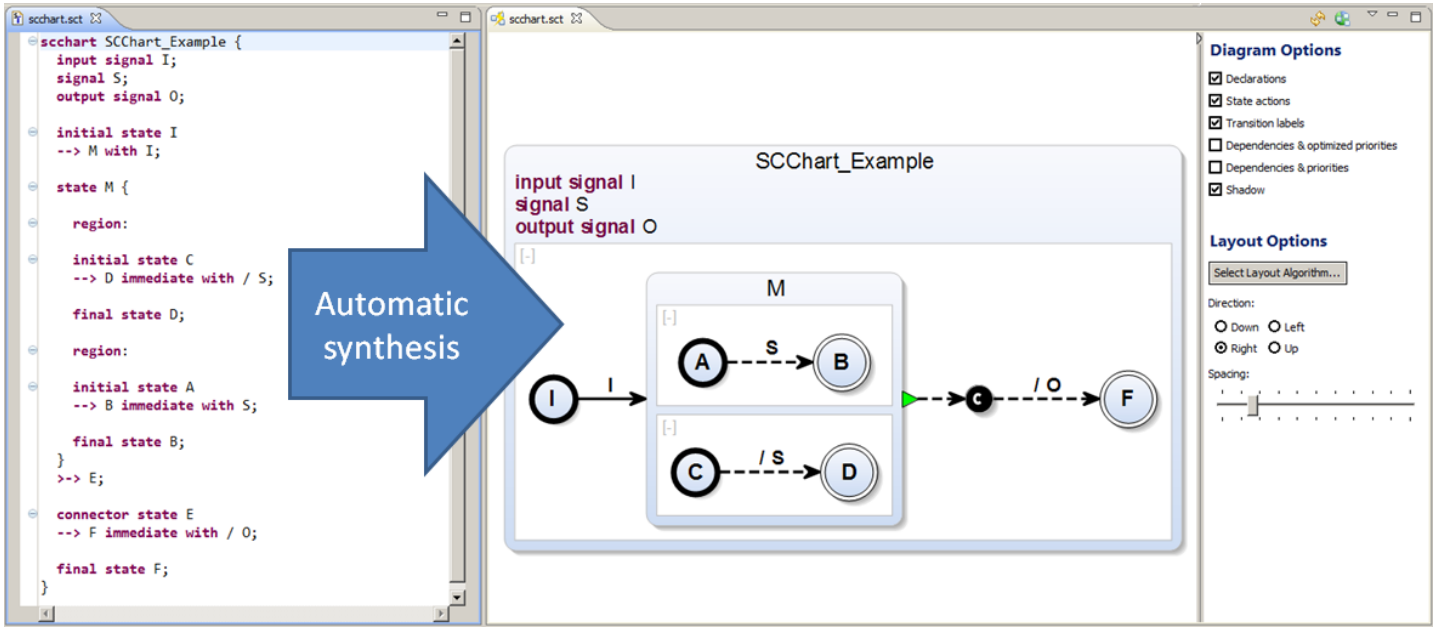
- Get all benefits from graphical modeling

Contribution



- Get all benefits from graphical modeling
- Preserve all the benefits from textual modeling

Contribution



- Get all benefits from graphical modeling
- Preserve all the benefits from textual modeling
- $\Rightarrow$ **Automatic synthesis of diagrams**: **KIELER Light-Weight Diagram (KLighD)**

Overview

- Foundations & Concept
- Demo
- Case Study Results

Synthesis of Diagrams

[C. Schneider et al., VL/HCC'13]

- Input: Model (dsl, xml, moml)

Synthesis of Diagrams

[C. Schneider et al., VL/HCC'13]

- Input: Model (dsl, xml, moml)
- Output: Configurable diagram

Synthesis of Diagrams

[C. Schneider et al., VL/HCC'13]

- Input: Model (dsl, xml, moml)
- Output: Configurable diagram
  - Diagram options: E.g., show transition labels

Synthesis of Diagrams

[C. Schneider et al., VL/HCC'13]

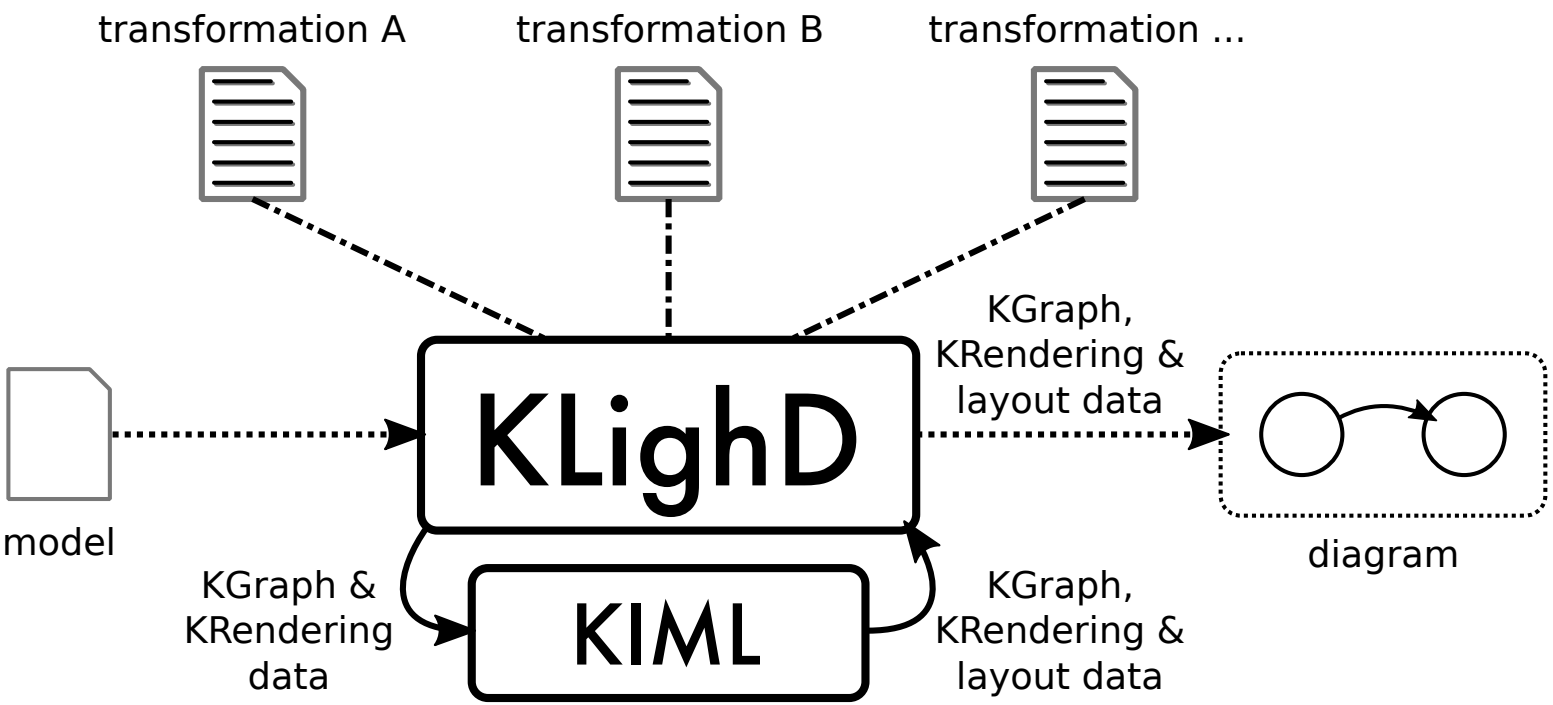
- Input: Model (dsl, xml, moml)
- Output: Configurable diagram
  - Diagram options: E.g., show transition labels
  - Layout options: E.g., direction or spacing

Synthesis of Diagrams

[C. Schneider et al., VL/HCC'13]

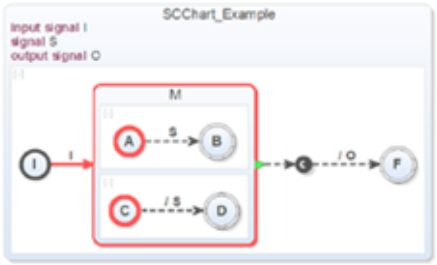
- Input: Model (dsl, xml, moml)
- Output: Configurable diagram
  - Diagram options: E.g., show transition labels
  - Layout options: E.g., direction or spacing
- Requirement: Automatic Layout ($\rightarrow$ KIML)

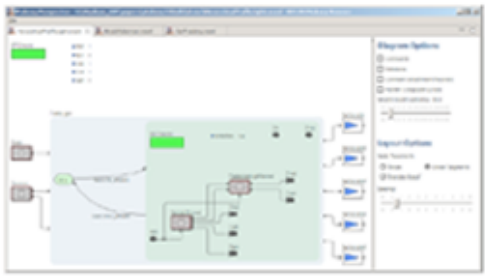
KLighD Architecture



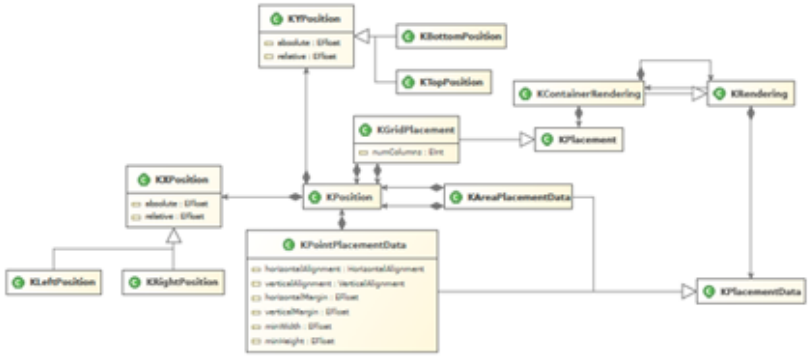
[C. Schneider et al., VL/HCC'13]

KLighD Demo

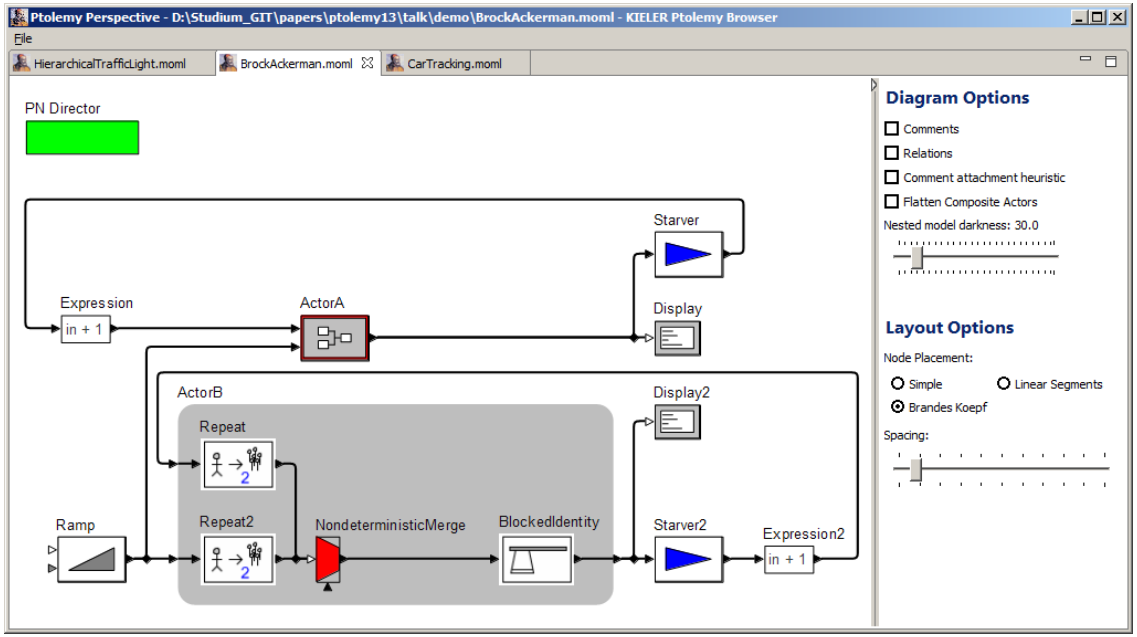




D E M O

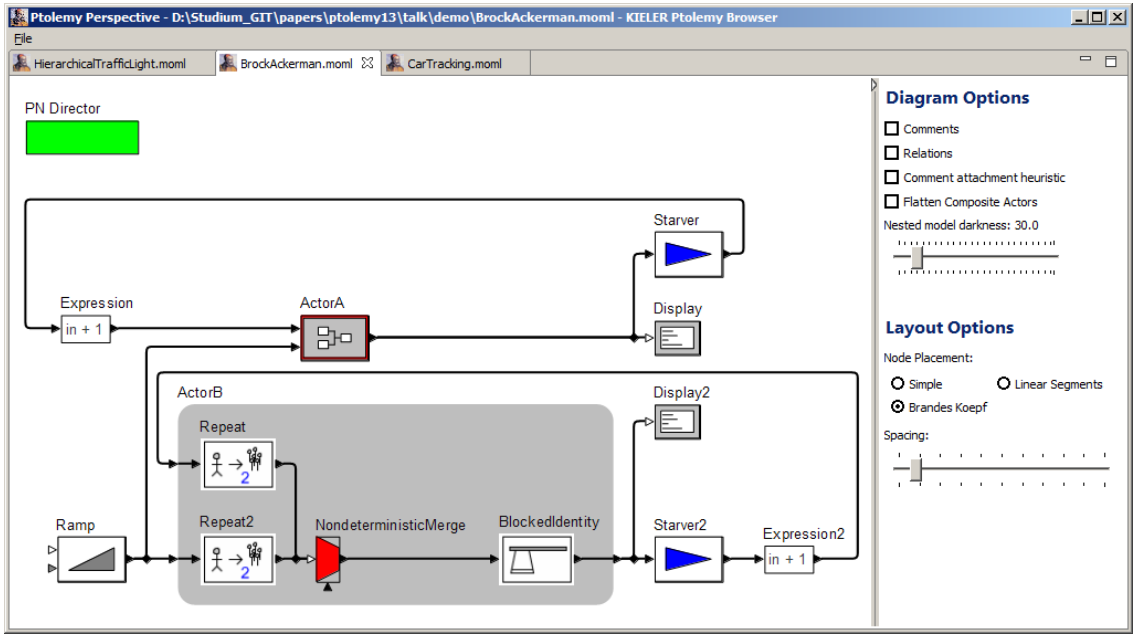


Ptolemy Case Study: Model Browsing



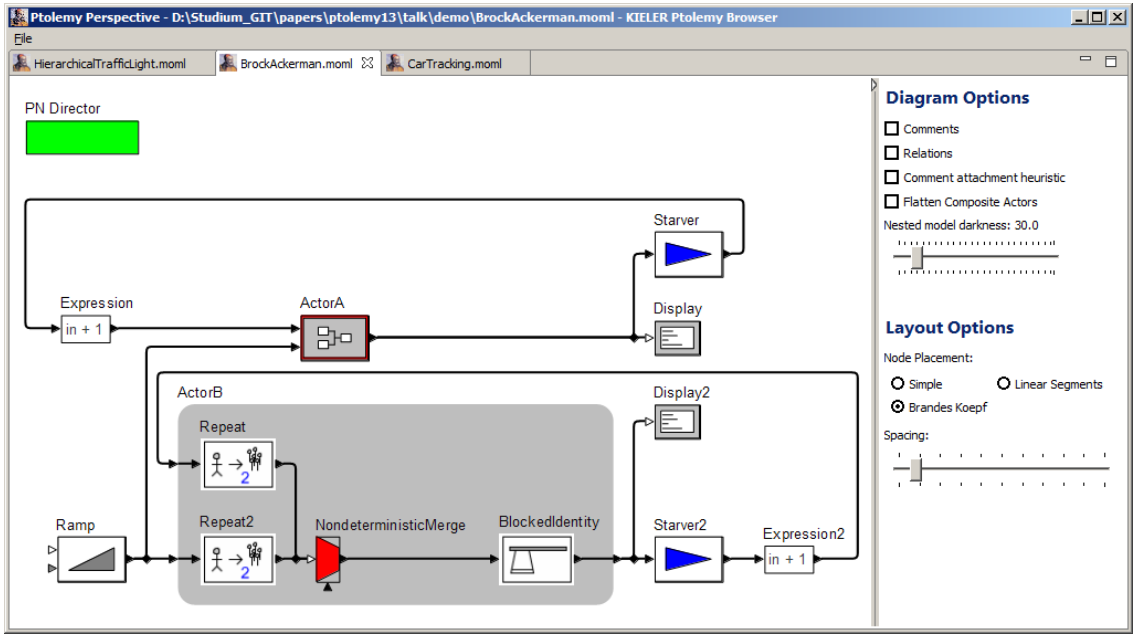
- ☺ Readability (normalized diagrams with fixed layout settings, configurable settings)

Ptolemy Case Study: Model Browsing



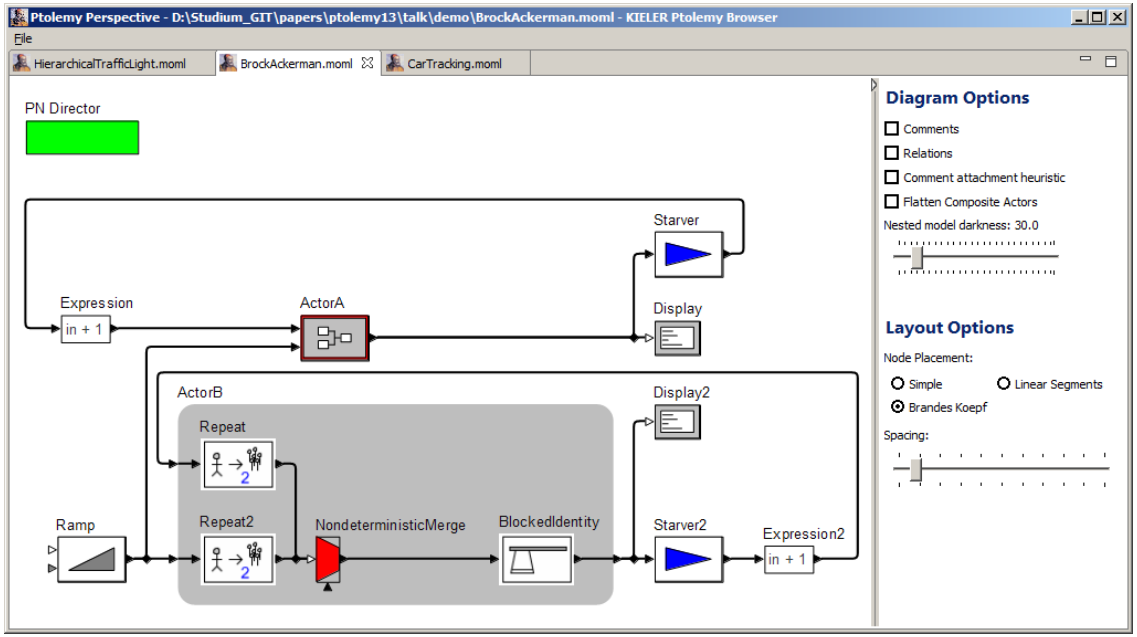
- ☺ Readability (normalized diagrams with fixed layout settings, configurable settings)
  - ☺ Hierarchy (no new windows, inner and outer ports)

Ptolemy Case Study: Model Browsing



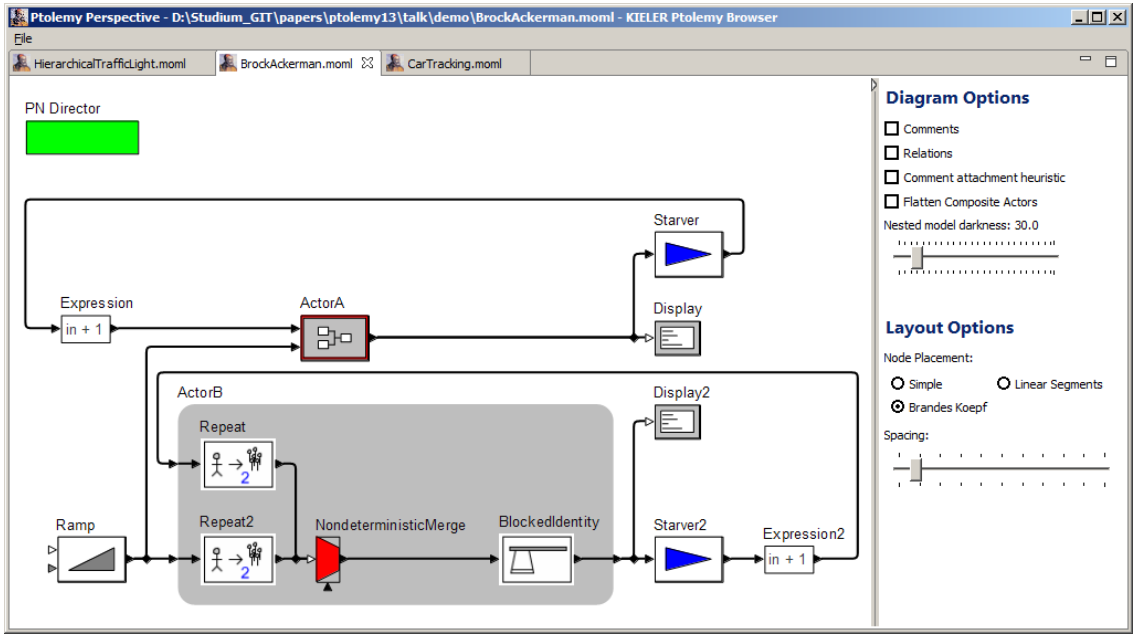
- ☺ Readability (normalized diagrams with fixed layout settings, configurable settings)
  - ☺ Hierarchy (no new windows, inner and outer ports)
  - ☺ Large models (Focus&Context, collapse & expand)

Ptolemy Case Study: Model Browsing



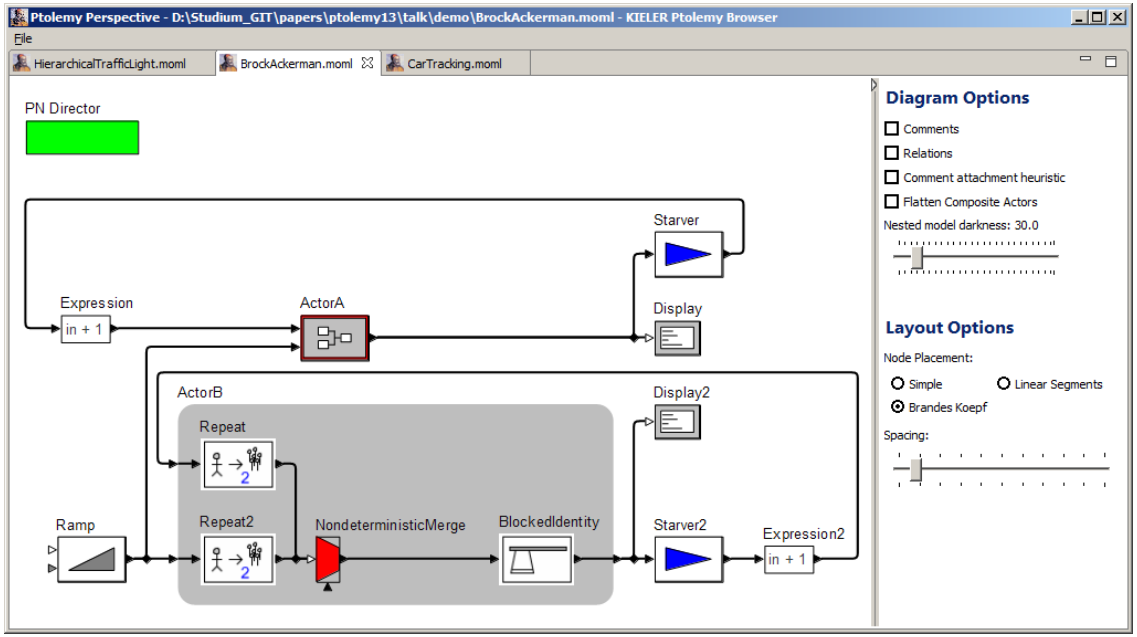
- ☺ Readability (normalized diagrams with fixed layout settings, configurable settings)
  - ☺ Hierarchy (no new windows, inner and outer ports)
  - ☺ Large models (Focus&Context, collapse & expand)
  - ☺ Complex models (filter details, e.g., transition labels)

Ptolemy Case Study: Model Browsing



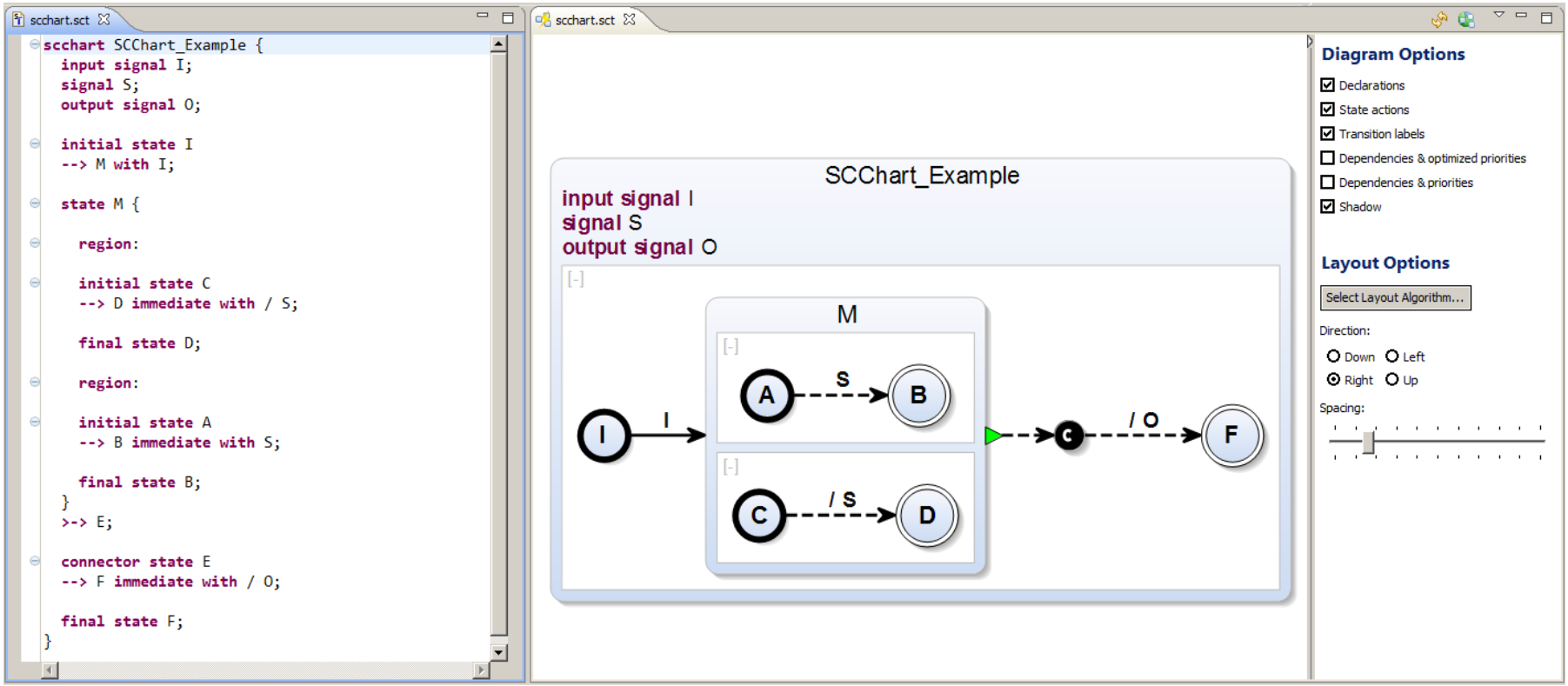
- ☺ Readability (normalized diagrams with fixed layout settings, configurable settings)
  - ☺ Hierarchy (no new windows, inner and outer ports)
  - ☺ Large models (Focus&Context, collapse & expand)
  - ☺ Complex models (filter details, e.g., transition labels)
- ☺ Maintenance / Handling (create/edit the model in Vergil, generate it, use a textual DSL, ...)

Ptolemy Case Study: Model Browsing

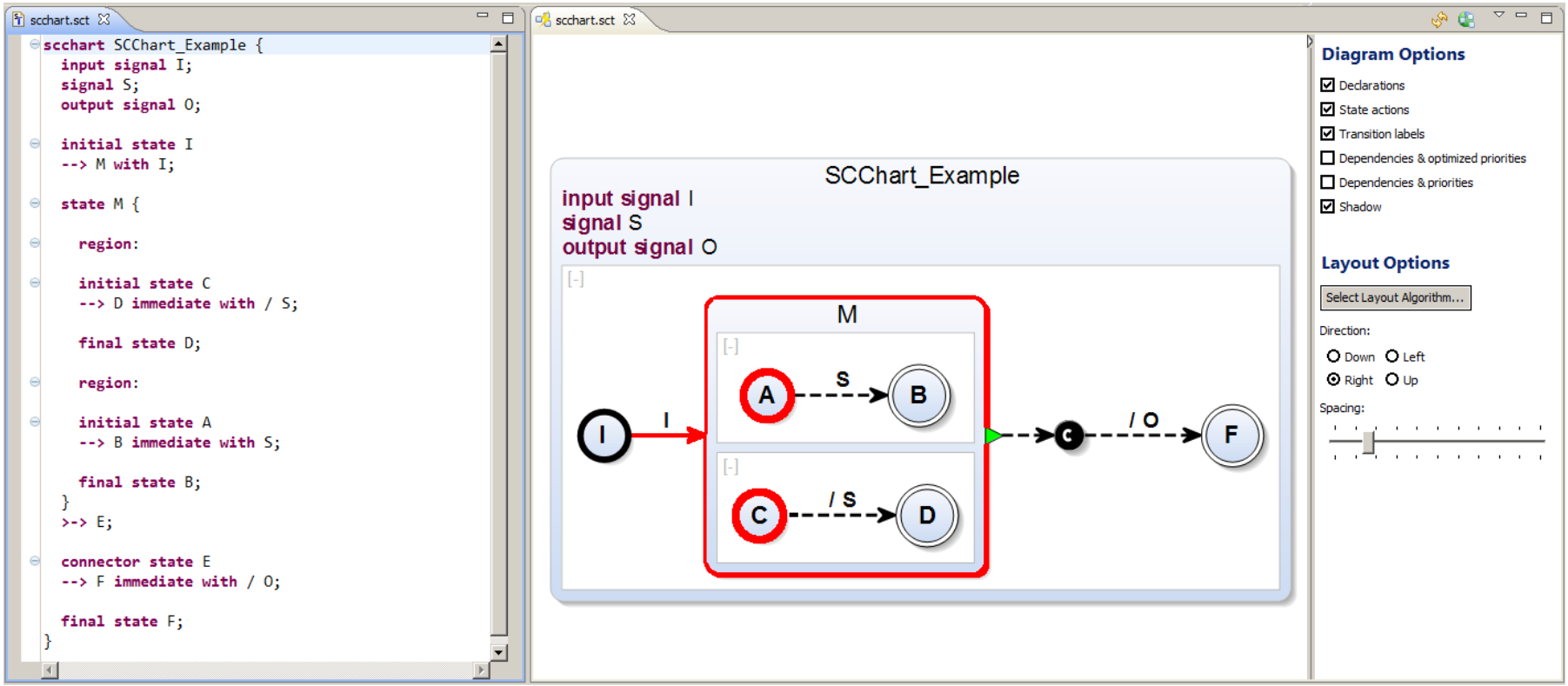


- ☺ Readability (normalized diagrams with fixed layout settings, configurable settings)
  - ☺ Hierarchy (no new windows, inner and outer ports)
  - ☺ Large models (Focus&Context, collapse & expand)
  - ☺ Complex models (filter details, e.g., transition labels)
- ☺ Maintenance / Handling (create/edit the model in Vergil, generate it, use a textual DSL, ...)
- ☺ Light-Weight (no editing, no transactions → just transient views)

Model Editing & Simulation



Model Editing & Simulation



Summary and Outlook

- Models are created once but read many times
- Large and complex, hierarchical models are hard to read and maintain

Summary and Outlook

- Models are created once but read many times
- Large and complex, hierarchical models are hard to read and maintain
- Automatic light-weight diagrams
  - help browsing/reading
  - and maintaining models

Summary and Outlook

- Models are created once but read many times
- Large and complex, hierarchical models are hard to read and maintain
- Automatic light-weight diagrams
  - help browsing/reading
  - and maintaining models
- Models can be textual or graphical

To Go Further

KLAUSKE, L. K., SCHULZE, C. D., SPÖNEMANN, M., AND VON HANXLEDEN, R.

Improved layout for data flow diagrams with port constraints.

In *Proceedings of the 7th International Conference on the Theory and Application of Diagrams (DIAGRAMS'12)* (2012), vol. 7352 of *LNAI*, Springer, pp. 65–79.

SCHNEIDER, C., SPÖNEMANN, M., AND VON HANXLEDEN, R.

Just model! – Putting automatic synthesis of node-link-diagrams into practice.

In *Proceedings of the IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC'13)* (San Jose, CA, USA, 15–19 Sept. 2013).

With accompanying poster.

SUGIYAMA, K., TAGAWA, S., AND TODA, M.

Methods for visual understanding of hierarchical system structures.

IEEE Transactions on Systems, Man and Cybernetics 11, 2 (Feb. 1981), 109–125.

UNI KIEL, REAL-TIME AND EMBEDDED SYSTEMS GROUP.

KIELER webpage.

http://www.informatik.uni-kiel.de/en/rtsys/kieler/.

VON HANXLEDEN, R., LEE, E. A., MOTIKA, C., AND FUHRMANN, H.

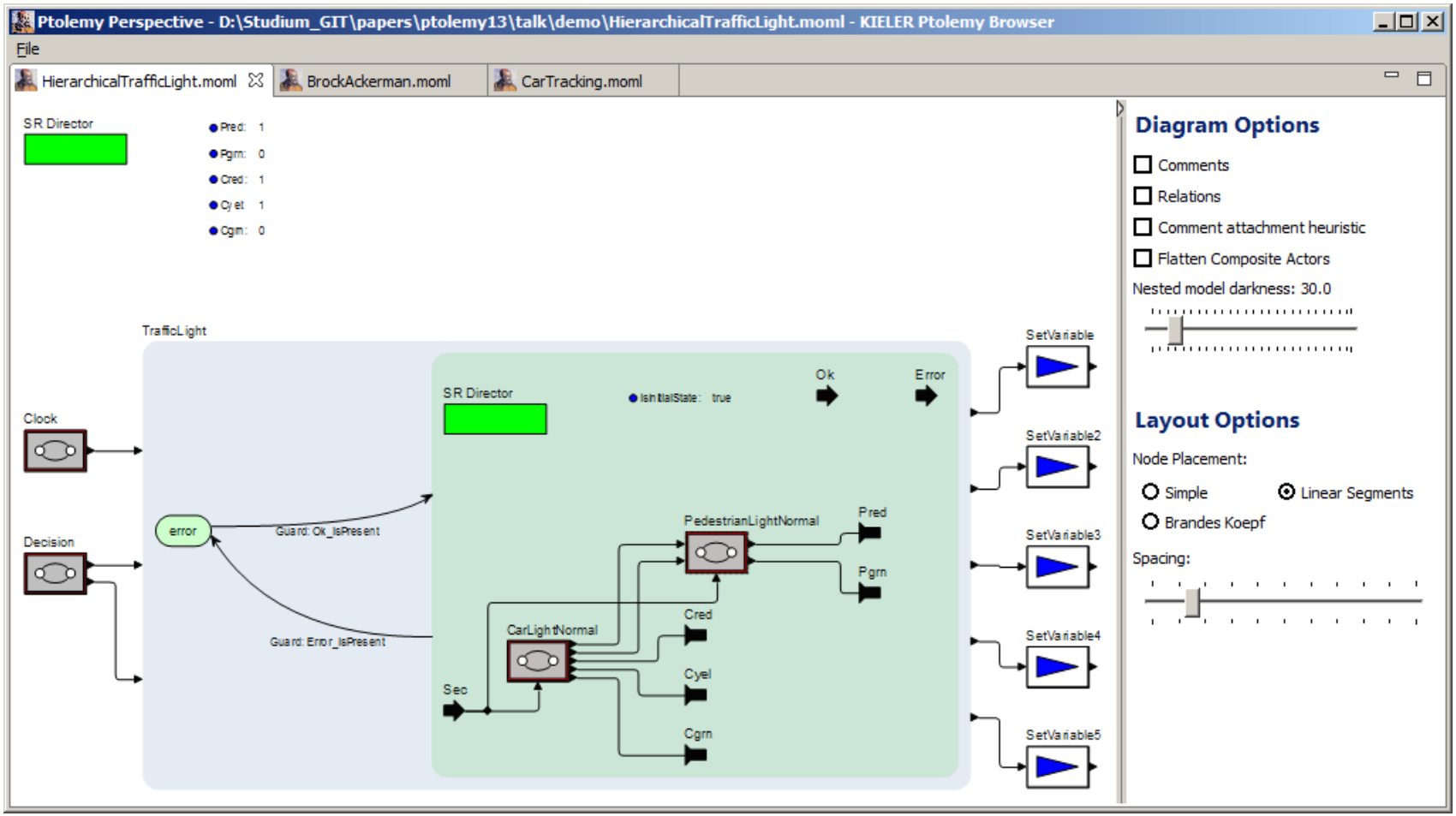
Multi-view modeling and pragmatics in 2020 — position paper on designing complex cyber-physical systems.

In *Proceedings of the 17th International Monterey Workshop on Development, Operation and Management of Large-Scale Complex IT Systems, LNCS* (Oxford, UK, Dec. 2012), vol. 7539.

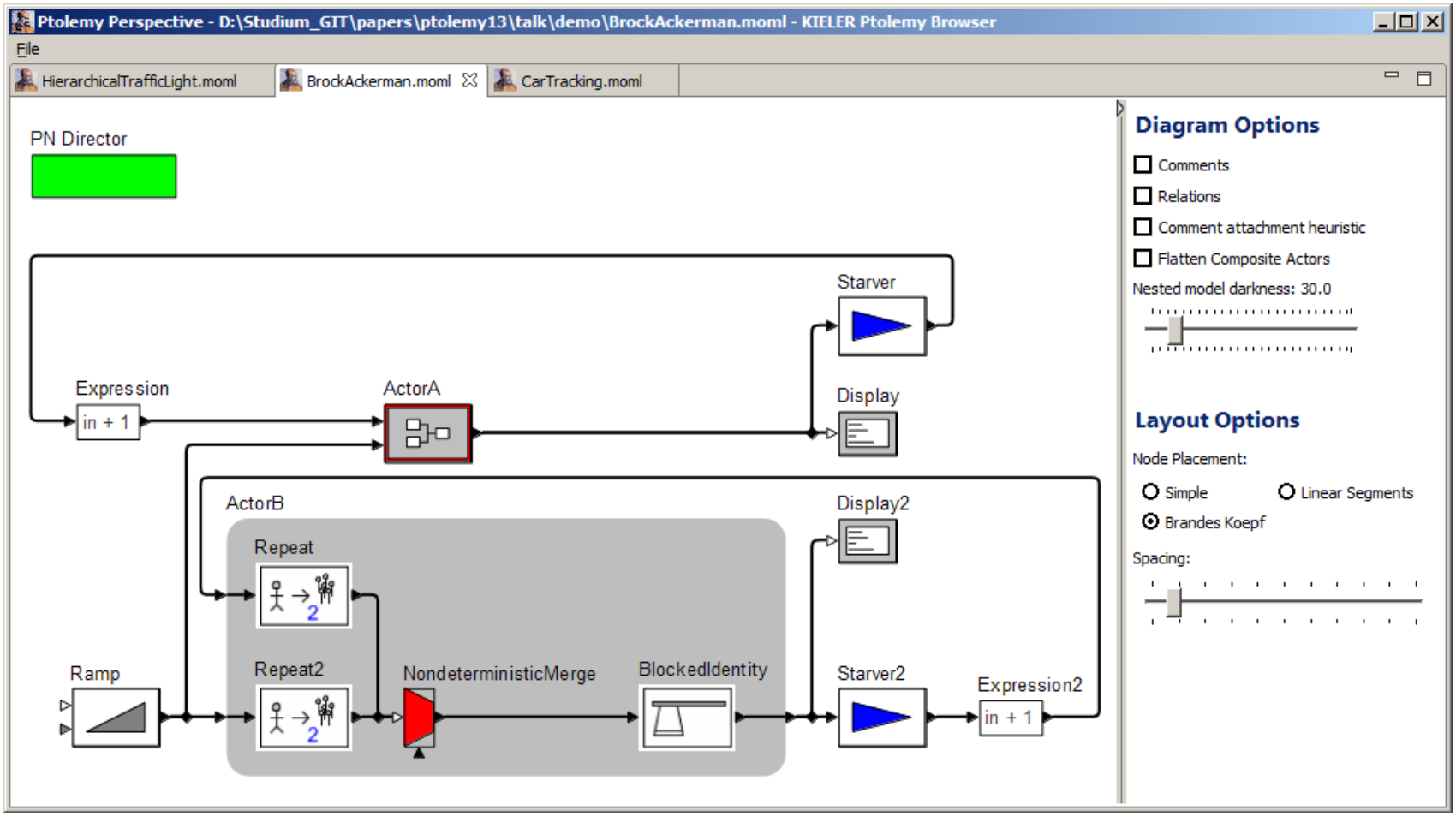
Thank you for your attention and participation!

Any questions or suggestions?

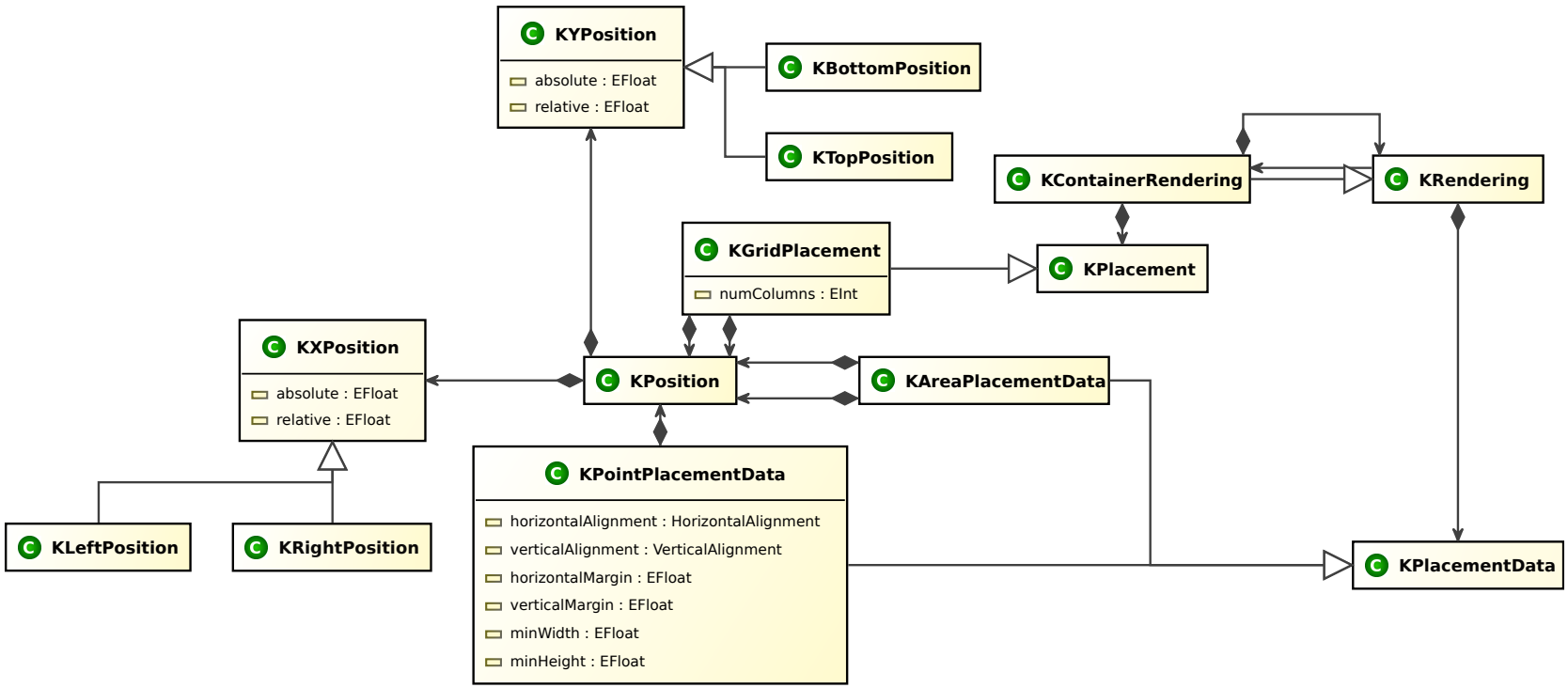
Ptolemy Case Study: Model Browsing



Ptolemy Case Study: Model Browsing (cont'd)



General Data Structure Visualization



KLighDning - Collaborative Browser-Based Viewer