Interactive Esterel to SyncCharts Transformation for executing Esterel with Ptolemy

Christian Motika

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

SYNCHRON ’11, Dammarie-les-Lys
01. DEC 2011
Motivation

- Synchronous model of computation (MoC):
  - Esterel, SyncCharts, SC (control flow)
  - Lustre, Signal, SCADE (data flow)
  - Ptolemy (SR domain)

- SyncCharts a synchronous statechart dialect
  - Primary example for KIELER framework

- KlePto: Executing SyncCharts w/ Ptolemy

- KIES: Esterel to SyncCharts transformation
  - Execute Esterel w/ Ptolemy
Overview

- Esterel to SyncCharts transformation (KIES)
- SyncCharts Execution (KlePto)
  - Demo
- Summary
Esterel

module ABRO:

input A, B, R;
output O;

loop
  [ await A || await B ];
  emit O;
  each R

end module

- Synchronous, imperative, control flow language
  - Developed by J.-P. Marmorat and J.-P. Rigault
  - G. Berry developed a formal semantics for Esterel in 1983

- Synchrony hypothesis
  - Discrete ticks
  - Computations take no time

- Signal coherence rule
SyncCharts

- Invented by Charles André
- Statechart dialect
- Mealy machine with
  - Parallelism, hierarchy, compound events, broadcast
- Built on Esterel semantics

Interface: A, B, Reset, arm, disarm, AB,
Example: ABRO

```python
module ABRO:
    input A, B, R;
    output O;
    loop
        [ await A || await B ];
        emit O;
        each R
    end module
```

![ABRO diagram](image-url)
Model transformations

▶ Applications
  ▶ Synthesize multiple (graphical/textual) views from one model
  ▶ Edit a model (refactoring, optimization)
  ▶ Code generation
  ▶ Simulation desires

▶ Drawbacks
  ▶ Large and inflexible
  ▶ Hard to visualize
  ▶ Hard to debug
  ▶ Not interactive

▶ Goal of KIES: Address the above drawbacks
  → Use case: KIELER Esterel to SyncCharts transformation
Esterel to SyncCharts

Motivation and Concept
Transformation Rules
SyncCharts Optimization

Esterel to SyncCharts

Transformation Optimization

Christian Motika - CAU Kiel
Transformation Rule

“A transformation rule is a description of how one or more constructs in the source language can be transformed into one or more constructs in the target language” (Mens and Gorp)

- Esterel to SyncCharts
  - One rule for each Esterel statement
  - Rules presented by Lars Kühl (also formal proofs for Esterel to SyncCharts)
  - [Synthesizing Safe State Machines from Esterel, LCTES 2006]

- SyncCharts Optimization
  - One rule for a SyncCharts meeting certain criteria
Implementation

[Interactive Transformations for Visual Models, MEMWe 2011]
Esterel to SyncCharts - *emit*, *loop-each*

**emit**

```
emit sig1
```

**loop-each**

```
loop s each e
```
SyncCharts Optimization

rule 7

any

any

1

2

any

any

1

2

Christian Motika - CAU Kiel
Overview

- Esterel to SyncCharts transformation (KIES)
- SyncCharts Execution (KlePto)
  - Demo
- Summary
What is KIELER?

- Kiel Integrated Environment for Layout Eclipse Rich Client
- Modeling platform and test bed
  - Improve pragmatics
- Open source and Eclipse based (plug-ins)
- General concepts:
  - Generic approaches
  - Symbiosis w/ Eclipse technologies (e.g., EMF, GMF, TMF, Xpand, Xtend)
  - Interfaces to other tools (Ptolemy, Papyrus)
SyncCharts Execution in KIELER
Ptolemy

▶ „The Ptolemy project studies heterogeneous modeling, simulation, and design of concurrent systems."

Introduction to Ptolemy II, UC Berkeley

▶ Executable Models to describe behavior of reactive systems

▶ Ptolemy models are a set of interacting components → Actor-Oriented Design
SyncCharts in Ptolemy

- Mapping SyncCharts to Ptolemy:
  - Mealy machine ↔ ModalModel
  - Orthogonality ↔ Concurrent Actors (inherent)
  - Hierarchy ↔ Compound Actors, state refinements
  - Compound events ↔ Expression language

- Interesting:
  - Implicit broadcast vs. explicit signal representation
  - Signal coherence (must/cannot analysis)
SyncCharts in Ptolemy - Example
KIELER Demo

LIVE DEMO
Summary

▶ Research goals (long term)
  ▶ Investigate on synchronous languages
  ▶ Bringing together graphical and textual syntax
  ▶ Integrate Esterel in KIELER
    ▶ Improve pragmatics
    ▶ Validation purposes (SC and KlePto)
    ▶ Current work in progress: Simulation/Debugging with CEC

▶ Research goals (short term)
  ▶ Modular and interactive transformations
    ▶ Understand
    ▶ Debug
  ▶ Teaching

▶ Acknowledgements: Ulf Rüegg
To Go Further

**CHARLES ANDRÉ.**

**GÉRARD BERRY.**
The Esterel v5 Language Primer, 2000.

**RÜEGG, U., MOTIKA, C., AND VON HANXLEDEN, R.**
Interactive transformations for visual models.

**UC BERKELEY, EECS DEPT.**
Ptolemy webpage.
http://ptolemy.eecs.berkeley.edu/.

**UNI KIEL, REAL-TIME AND EMBEDDED SYSTEMS GROUP.**
KIELER webpage.
http://www.informatik.uni-kiel.de/en/rtsys/kieler/.
Thank you for your attention and participation!

Any questions or suggestions?