Component-Interaction Automata

for Component-Based System Specification and Verification

Barbora Zimmerova

Masaryk University in Brno
Czech Republic

Laser 2005
Outline

• Introduction
  – Specification languages

• Component-Interaction automata language
  – CI automaton
  – Composition of CI automata

• Future work
Introduction

- Component-based systems (CBSs)
- Component interactions in CBSs
- Specification and verification process

• **Specification languages:**
  - Architecture description languages (ADLs) 
    (*Wright, Darwin/Tracta, SOFA*)
  - Automata-based languages
    (*Interface automata, I/O automata, Team automata*)
Component-Interaction automata (CI automata) language

- Automata-based language
- Three types of actions (input, output and internal)
- Synchronization of one input and one output action which becomes internal later on
- Freedom of choosing the transition set
- Preservation of important interaction information
CI automaton

- A component-interaction automaton is a 5-tuple $C = (Q, \text{Act}, \delta, I, S)$
  - $Q$ is a finite set of states
  - $\text{Act}$ is a finite set of actions involved in structured transition labels (triplets)
  - $\delta$ is a finite set of labelled transitions
  - $I$ is a nonempty set of initial states
  - $S$ represents a hierarchy of components in $C$
Example of CI automata

$C_1$: $0 \rightarrow 1$ 
$C_1$: $0 \rightarrow 2$ 

$C_2$: $0 \rightarrow 1$ 
$C_2$: $0 \rightarrow 3$ 

$C_3$: $0 \rightarrow 1$ 
$C_3$: $0 \rightarrow 3$ 

Hierarchy: (1) 
Hierarchy: (2) 
Hierarchy: (3)
Composition of CI automata

- Set of CI automata
- CI automaton over the set
  - Complete transition space $\Delta$
  - Transition set $\delta$ is a subset of $\Delta$
    (determined by the architecture description and other characteristics)
Example of composition

Hierarchy: \(((1), (2), (3))\)
Future work

• Practical
  – Automatic transformation from ADL specification (SOFA, …) to CI automata
  – Automatic transformation from CI automata specification to input languages of model checking tools (DiVinE, …)

• Theoretical
  – Theory of CI automata (behavioral equivalences, expressibility, …)