Describing a reactive system as static input-output transformation is hard

- A reactive system changes under inputs
  - Inputs trigger internal state changes
- Data transformation description inadequate
  - Different transformations depending on the state
- Well-known examples
  - Many embedded systems
  - Interactive Software (Desktop-Applications)
  - Even hardware can be seen this way
Three interleaved areas of state-based specification and verification

We will focus on Statecharts and recent interleavings

Outline

- Introduction
- State-based technologies
- Maturation of Statecharts
- Inspiration between fields
- Summary

Statechart maturation
Statecharts more or less follow Redwine-Riddle

1. Basic research
2. Concept formulation
3. Development & extension
4. Internal exploration
5. External exploration
6. Popularization
7. Inspiration

Inspiration phase
Exchange of ideas
- Inspiration of
- Inspiration by other fields
- Other to advance research

STATEMATE provided the first graphical specification environment

- STATEMATE: A Working Environment for the Development of Complex Reactive Systems
  - D. Harel, H. Lachover, A. Naamad, A. Pnueli, M. Politi, R. Sherman and A. Shull-Trauring
  - ICSE-10 1988 [ICSE n-10]
- Enhanced method
- Question: How can we specify reactive systems?
- Method of development
- Result: We use three interrelated graphical notations
  - Technique
- Validation: Used in practice, looks good
  - Experience (qualitative model)
  - Assertion

Statecharts are expressive and scalable

OR-states
Hierarchy
AND-states

STATEMATE: A Working Environment for the Development of
Complex Reactive Systems
‰
D. Harel, H. Lachover, A. Naamad, A. Pnueli, M. Politi, R. Sherman and A. Shull-Trauring
‰
ICSE-10 1988 [ICSE n-10]
‰
Enhanced method
‰
Question: How can we specify reactive systems?
‰
Method of development
‰
Result: We use three interrelated graphical notations
‰
Technique
‰
Validation: Used in practice, looks good
‰
Experience (qualitative model)
‰
Assertion
RSML tailors Statecharts for requirements engineering

- Graphical language to formalize requirements
  - Based on Statecharts
  - Explicit focus on ease of communication with domain experts
    - Logical tables to capture state transition conditions
- Ignores design aspects of the system
  - No module or even activity structure
- Successfully adopted by practitioners in the field
  - Validated in major specification effort for TCAS II
  - Looked into as case study for research paradigm

Outline

- Introduction
- State-based technologies
- Maturation of Statecharts
- Inspiration between fields
- Summary

Statecharts entered the inspiration phase around 2000

- Statecharts and model checking
  - Model Checking of Hierarchical State Machines
    - R. Alur and M. Yannakakis
  - Overview of ConFoRT: A Model Checking Reasoning Framework
    - J. Ivers & N. Sharygina
    - SEI technical report 2004
- Statechart generation from scenario diagrams
  - A Workbench for Synthesising Behaviour Models from Scenarios
    - S. Uchitel and J. Kramer
    - ICSE-23, 188-197, 2001
  - Revisiting Statechart Synthesis with an Algebraic Approach
    - T. Zabl, L. Helouet, J.-M. Jouvelot
    - ICSE-26, 2004