Software Design, Modelling and Analysis in UML
Lecture 16: Hierarchical State Machines I

2015-01-15

Prof. Dr. Andreas Podelski,
Dr. Bernd Westphal
Albert-Ludwigs-Universität Freiburg, Germany

Contents & Goals

Last Lecture:
• Missing transformers: create and destroy
• Step and run-to-completion (RTC) step, divergence

This Lecture:
• Educational Objectives:
  • What does this State Machine mean? What happens if I inject this event?
  • Can you please model the following behaviour.
  • What does this hierarchical State Machine mean? What may happen if I inject this event?
  • What is: AND-State, OR-State, pseudo-state, entry/exit/do, final state, . . .

Putting It All Together

The Missing Piece: Initial States

Recall: a labelled transition system is
\( (S, \rightarrow, S_0) \).
We have
• \( S \) : system configurations 
  \( (\sigma, \varepsilon) \)
• \( \rightarrow \) : labelled transition relation 
  \( (\sigma, \varepsilon) \rightarrow (\sigma', \varepsilon') \).
Wanted: initial states \( S_0 \).
Proposal: Require a (finite) set of object diagrams \( OD \) as part of a UML model \( (CD, SM, OD) \).
And set \( S_0 = \{ (\sigma, \varepsilon) \mid \sigma \in G^{-1}(OD), OD \in OD, \varepsilon \text{ empty} \} \).

Other Approach: (used by Rhapsody tool) multiplicity of classes.
We can read that as an abbreviation for an object diagram.

Semantics of UML Model — So Far

The semantics of the UML model \( M = (CD, SM, OD) \) where
• some classes in \( CD \) are stereotyped as 'signal' (standard), some signals and attributes are stereotyped as 'external' (non-standard),
• there is a 1-to-1 relation between classes and state machines,
• \( OD \) is a set of object diagrams over \( CD \), is the transition system \( (S, \rightarrow, S_0) \) constructed on the previous slide.
The computations of \( M \) are the computations of \( (S, \rightarrow, S_0) \).
Let $M = (C, D, S, M, O, D)$ be a UML model.

We call $M$ consistent iff, for each OCL constraint $\text{expr} \in \text{Inv}(C, D)$, $\sigma |= \text{expr}$ for each "reasonable point" $(\sigma, \varepsilon)$ of computations of $M$.

(Cf. exercises and tutorial for discussion of "reasonable point").

Note: we could define $\text{Inv}(S, M)$ similar to $\text{Inv}(C, D)$.

**Pragmatics**:

- In UML-as-blueprint mode, if $S, M$ doesn't exist yet, then $M = (C, D, \emptyset, O, D)$ is typically asking the developer to provide $S, M$ such that $M' = (C, D, S, M, O, D)$ is consistent. If the developer makes a mistake, then $M'$ is inconsistent.

- Not common: if $S, M$ is given, then constraints are also considered when choosing transitions in the RTC-algorithm. In other words: even in presence of mistakes, the $S, M$ never move to inconsistent configurations.