Prof. Dr. A. Podelski, Dr. B. Westphal
E. Plaku

Wintersemester 2011/12

Software Design, Modeling, and Analysis in UML

http://swt.informatik.uni-freiburg.de/teaching/winter-term-2011-2012/sdmauml/sdmauml

Exercise Sheet 7

Early submission: Monday, 2012-02-13, 12:00 Regular submission: Tuesday, 2012-02-14, 12:00

Exercise 1 (5/10 Points)

Consider the inheritance hierarchy in Figure 1(a). Assuming the *late binding* approach for choosing the implementation of behavioural features and assuming the state machine of class C_0 given by Figure 1(b), sending an E-event to an instance of class C_3 would use the implementation of $f()$ provided by C_2.

Describe in words how the "right" implementation is selected and formalise this selection principle assuming complete signatures including the inheritance relation "$\lhd$" and the set-inclusion semantics, i.e.,

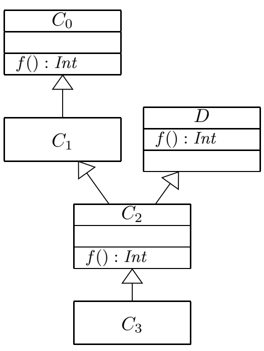
$$C \lhd C' \iff \mathscr{D}(C') \subsetneq \mathscr{D}(C).$$

Exercise 2 (5/10 Points)

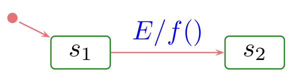
Consider the State Machine from Figure 1(b).

(i) Explain how the State Machine can be seen as an instance of the UML meta-model as given in Section 15 of [OMG, 2007]. (3)

(ii) Choose one of the constraints applying to pseudostates (cf. Section 15.3) and prove that the State Machine satisfies it (and thus, that the State Machine is well-formed regarding that condition). (2)



(a)



(b)

Figure 1: Inheritance Hierarchy and State Machine of C_3.

Exercise 3 (10 Bonus)

We could formalise the substitution principle as follows: Class C_2 is a behavioural subtype of class C_1 in UML model $\mathcal{M}$ if and only if for each system configuration (σ, ε) and for all $id_1 \in \mathscr{D}(C_1) \cap \mathrm{dom}(()\sigma)$, for all $id_2 \in \mathscr{D}(C_2) \setminus \mathscr{D}(\sigma)$ such that $\sigma(id_2)|_{atr(C_1)} = \sigma(id_1)$, we have that if

$$(\sigma, \varepsilon)[id_1/id_2] = (\sigma_0, \varepsilon_0) \xrightarrow[u_0]{(cons_0, Snd_0)} \cdots \in [\![\mathcal{M}]\!]$$

then there exists

$$(\sigma, \varepsilon) = (\sigma'_0, \varepsilon'_0) \xrightarrow[u'_0]{(cons'_0, Snd'_0)} \cdots \in [\![\mathcal{M}]\!]$$

such that

$$\forall\, i \in \mathbb{N} \bullet \sigma_i(id_2)|_{atr(C_1)} = \sigma_i(id_1)$$

where $(\sigma, \varepsilon)[id_1/id_2]$ denotes consistent replacement of id_1 by id_2 in σ and ε, e.g., replace id_1 by id_2 in all values of links and event destination id_1 to id_2 etc.
Page 818 of [Telelogic, 2008] states the following on inherited State Machines:

> *" You cannot make the following changes to items in the statechart of a subclass:*
>
> - *Change the source of a transition.*
> - *Change the triggers (events or triggered operations).*
> - *Delete or rename a state.*
> - *Draw a state around an existing state.*
>
> *You can make the following changes to items in the statechart of a subclass:*
>
> - *Change anything that does not affect the model, such as moving things in the diagram without actually editing.*
> - *Add objects to a state.*
> - *Add more states, but not re-parent states.*
> - *Attach a transition to a different target.*
>
> *An inherited statechart consists of all the items inherited from the superclass, as well as modified and added elements."*

(i) Prove that these rules *do not ensure* that $C \lhd C'$ implies behavioural sub-typing as defined above. (9)

(ii) Can you propose rules that do? (1)

Hint: That an implication does not hold can be proven by a counter-example. For instance a modification of the C-and-D example from Lecture 15 together with witness computation paths in form of recorded sequence diagrams.

References

[OMG, 2007] OMG (2007). Unified modeling language: Superstructure, version 2.1.2. Technical Report formal/07-11-02.

[Telelogic, 2008] Telelogic (2008). *Rhapsody User Guide.*