

SOFTWARETECHNIK SS01

ASSIGNMENT 3

The goal of this assignment is to complete the product definition — the next milestone along the way to the ECO system. The artifacts to be developed for this milestone are a glossary, an SRS (requirements specifications, Pflichtenheft), and a product model. In the previous assignment, you already finished the glossary and the SRS.

Since our project follows an object-oriented approach, the product model of our system should be constructed by an object-oriented analysis. So far, you started modeling the static properties of the model. In this assignment you will complete this last artifact by also modeling the dynamic properties and by revising the static model.

Exercise 4: (15 points) Use an object-oriented analysis approach to model the dynamic properties of the ECO-system. To express the relevant properties, use sequence diagrams, collaboration diagrams, state chart diagrams, and/or activity diagrams. Roughly Balzert’s guideline for an iterative modeling of the dynamics [Balzert, p. 390] is:

1. Refine each identified use case into different scenarios. Document the scenarios using sequence and/or collaboration diagrams. While doing that you should identify further needed operations and classes (cf. [Balzert, p. 413]).
2. An object’s life cycle is determined by the scenarios it is involved. Check for each class whether the life cycle of its instances is nontrivial. If so, express the life cycle with state chart diagrams (cf. [Balzert, p. 416]).
3. Check the operations in your class diagram. If the semantics of an operation is not obviously derivable from its name, describe it using state chart and/or activity diagrams (cf. [Balzert, p. 419]).

In Exercise 2 of the previous assignment, some additional non-functional requirements were stated. Note that your model must guarantee those requirements, too. To exemplify where and how these requirement are enforced, add notes and constraints to your UML diagrams.

Exercise 5: (5 points) While modeling the dynamic properties, revise your static model—that is, your class diagram(s)—according to the insights you gained by further working on the model.

Note that in UML class diagrams, you can annotate operations with attributes (*sequential*, *guarded*, *concurrent*) specifying the semantics of concurrent calls to instances of the operation (cf. Section 2.5.2 Abstract Syntax – Operation, and Section 3.26.2 Operation – Notation in the *OMG UML Specification, Version 1.3*.) If your CASE tool (Together) does not allow to add those, use notes instead.

Please check in all diagrams either as postscript or pdf documents by June 6, 2001.