

Übung Softwaretechnik I

Blatt 10 — Ausgabe am 21.12.2018, Abgabe der Lösungen bis **17.01.2019**

Aufgabe 10.1: Zustandsautomaten

[sportlich / Klausur-Bearbeitungszeit ca. 90 Min.; tatsächlich: _____; vorbereitende Lektüre: _____]

Schreiben Sie einen Interpreter für Zustandsübergangsdiagramme bzw. einfache UML-Zustandsautomaten.

Wesentliche Teile dieses Interpreters, insb. die Bedienschnittstelle, bekommen Sie als Vorgabe, Sie müssen diese Teile nur noch ergänzen. Diese Vorgaben und weitere Materialien finden Sie auf der WWW-Seite für die Übungsaufgaben.

Der Interpreter hat einen Zustandsautomaten als Eingabeparameter (Details hierzu s.u.). Nach Einlesen des Zustandsautomaten erscheint ein GUI mit folgenden Merkmalen:

- für jeden Trigger, der in dem Zustandsautomaten vorkommt, ist eine Schaltfläche vorhanden
- In einem Ausgabefeld werden die eingetretenen Trigger, die daraufhin eingenommenen Zustände und die auszuführenden Aktionen protokolliert.

Bild 1 zeigt ein Beispiel.

Am Anfang befindet sich das simulierte System im Startzustand gemäß Zustandsautomat. Beim Anklicken einer der Schaltflächen gilt das entsprechende Ereignis als eingetreten, und das System geht in den Folgezustand gemäß Zustandsautomat über. Protokolliert werden das eingetretene Ereignis, der neue Zustand und die auszuführenden Aktionen.

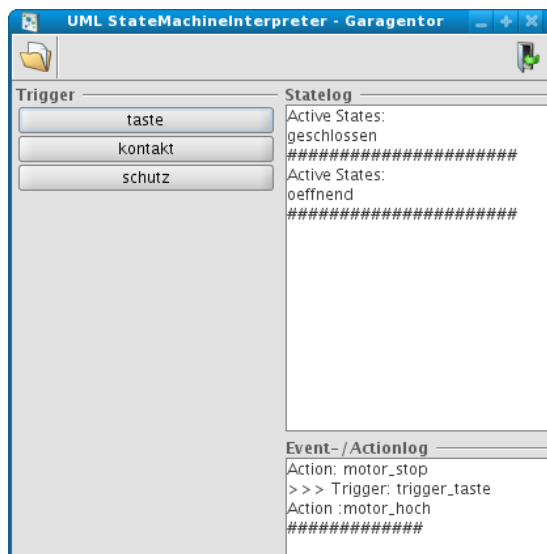
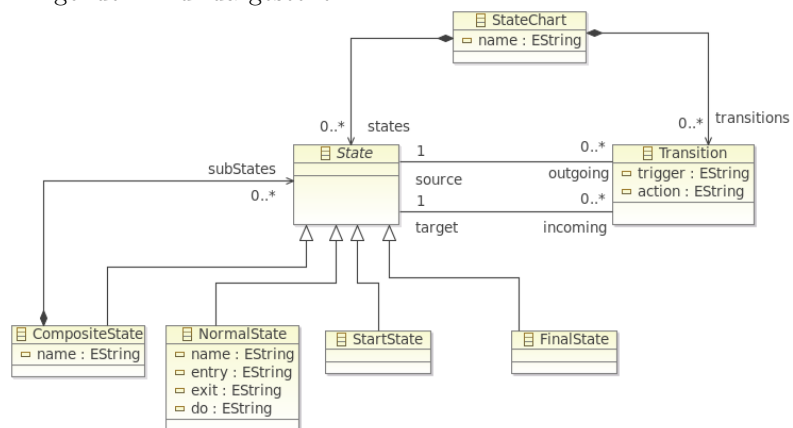


Abbildung 1: GUI

Technische Vorgaben. Das System muß in Eclipse realisiert werden. Vorgegeben werden diejenigen Teile des Simulators, die einen Zustandsautomaten einlesen und das GUI mit den o.g. Schaltflächen aufbauen. Entsprechende Dateien finden Sie auf der WWW-Seite für die Übungen.

Struktur der Eingabedaten. Vorausgesetzt wird i.f., daß Sie schon aus den Übungen zu EMF damit vertraut sind, wie Modelle in Form von Ecore-Laufzeitobjekten repräsentiert werden. Bei der ersten EMF-Übung waren die Modelle ER-Modelle; stattdessen werden es hier Zustandsautomaten sein. Zustandsautomaten werden ebenfalls als Graph dargestellt, allerdings mit anderen Typen von Knoten und Kanten als ER-Modelle.

Die Typen der Knoten und Kanten eines Graphen, der eine *state machine*¹ darstellt, sind im folgenden Bild dargestellt.



Testdaten. Als Testdaten werden Ihnen drei Zustandsautomaten zur Verfügung gestellt, die folgende Systeme modellieren:

1. ein Garagantor
2. einen Fernseher
3. eine Mikrowelle

Geben Sie Ihre Lösung als zip-Datei des Eclipse-Projekts ab.

¹Um Verwechslungen vorzubeugen: dies ist ein vereinfachtes Datenmodell für *state machines*; die UML-Definition einer *state machine* ist wesentlich komplexer.