

## Praktikum 2: Problemanalyse – statische Modellierung

Die *Akademie für berufliche Weiterbildung an der Hochschule Aalen* (ABW) führt für die regionale Industrie Weiterbildungsseminare an der Hochschule Aalen durch. Für die Verwaltung der Seminare, Referenten und Kunden der Akademie soll ein Programm entwickelt werden.

### Vorgaben:

Im ersten Schritt der **Anforderungsanalyse** wurden bereits Anwendungsfälle für das System festgelegt. Das *Enterprise-Architect*-Projekt ABW enthält ein Paket *Anwendungsfälle* mit den Anwendungsfällen, verteilt auf zwei Anwendungsfall-Diagramme. Ebenso sind bereits einige Datentypen definiert im Paket *Statische Analyse* (Klassendiagramm *Datentypen*, s.u.)

Die zugehörige EA-Projektdatei ist bei den Kursunterlagen in *moodle* unter Punkt 4,

*Praktikum 2: ABW - Statische Analyse → ABW statische Analyse – Aufgabe*  
abgelegt und kann von dort herunter geladen werden.

### Beschreibung der Anwendungsfälle:

#### AF1: Referentendaten erfassen:

Es werden die Daten sämtlicher Referenten, die für die Akademie Seminare anbieten oder durchführen, registriert. Dabei ist zwischen externen und internen Referenten (das sind FH-Mitarbeiter) zu unterscheiden. Für alle Referenten werden Name, Vorname und eine Email-Adresse gespeichert. Für externe Referenten auch die vollständige Adresse und ein Telefon- und eine Faxnummer, für interne Referenten die Nebenstellenummer und das Dienstzimmer an der Hochschule.

#### AF2: Seminarthema erfassen:

Die Referenten bieten ein oder mehrere Seminarthemen an. Zu einem Seminarthema werden der Titel, eine Kurzbeschreibung, die Anzahl der Unterrichtseinheiten, der Preis sowie die minimale und die maximale Teilnehmerzahl gespeichert. Ein Seminarthema kann auch von mehreren Referenten gleichzeitig angeboten werden, es wird jedoch stets einer der Referenten als verantwortlicher Referent (Seminarleiter) registriert. – Falls der Angebotssachbearbeiter ein Seminarthema erfasst, dessen Referent(en) noch nicht im System gespeichert ist (sind), werden als Erweiterung dieses Anwendungsfalls auch die Referentendaten erfasst.

#### AF3: Seminar erfassen:

Zu einem Seminarthema kann es mehrere Seminare geben (ein Seminar entspricht einer Veranstaltung zu einem Seminarthema) und wird durch eine eindeutige Seminarnummer identifiziert. Ein Seminar kann sich über mehrere Termine erstrecken. Für ein Seminar wird deshalb auch eine Liste von Terminen gespeichert, an denen das Seminar durchgeführt wird. Ein Termin besteht aus einem Datum, und zwei Uhrzeiten (von bis).

#### AF4: Kundendaten erfassen:

In dem zu entwickelnden Verwaltungssystem muss auch eine Kundendatenbank integriert werden. Für jeden registrierten Kunden werden der Name, die Adresse, die Telefon- und Faxnummer sowie eine Email-Adresse registriert.

**AF5: Anmeldung bearbeiten:**

Kunden können sich zu Seminaren anmelden, wobei eine Buchung erstellt wird (AF6). Falls ein Kunde ein Seminar buchen möchte, dessen maximale Teilnehmerzahl bereits erreicht ist, kann er sich in eine Warteliste eintragen lassen.

**AF6: Buchung erstellen:**

Eine Buchung bezieht sich auf ein Seminar (eine Veranstaltung) und einen Kunden. Zu der Buchung werden das Buchungsdatum, ein Rabatt (in Prozent, gegebenenfalls 0) und ein Buchungszustand registriert. Die Buchungszustände sind „offen“ (das heißt gebucht, noch nicht in Rechnung gestellt), „berechnet“ (Rechnung ist gestellt), „bezahlt“ (Rechnung wurde bezahlt) und „storniert“. (Stornierte Buchungen sollen im System registriert bleiben).

**AF7: Stornierung bearbeiten:**

Wird eine Buchung storniert, so wird ein Kunde von der Warteliste (in der Reihenfolge der Anmeldungen) berücksichtigt und für diesen eine neue Buchung erstellt (AF6).

**Aufgabe:**

Erstellen Sie mit dem *Enterprise Architect* ein statisches Analysemodell (Klassendiagramm) für die Seminarverwaltung. Fügen Sie dafür im Package *Statische Analyse* an ein neues Klassendiagramm ein.

Es ist folgendes zu berücksichtigen:

- (1) Die in den Anwendungsfall-Beschreibungen dargestellten Sachverhalte müssen repräsentiert sein.
- (2) Analysemuster: Verwenden Sie Analysemuster, wo dies sinnvoll und möglich ist, und kennzeichnen Sie diese in Ihrem Modell durch entsprechende Notizen.
- (3) Attributtypen: Geben Sie für die Attribute auch Attributtypen an. Verwenden Sie dabei wo sinnvoll die UML-Standard-Datentypen. Achten Sie deshalb darauf, dass die Zielsprache Ihrer Klassen auf *<none>* eingestellt ist: entweder für jede Klasse einzeln über die Klasseneigenschaften oder für das gesamte Projekt über *das Hauptmenü → Configure → Model → Options*. Dort unter *Source Code Engineering: Default Language for Code Generation = <none>* einstellen). Die bereits definierten Datentypen können ebenso verwendet werden (Diagramm *Datentypen*). Falls erforderlich, definieren Sie weitere Datentypen oder primitive Typen.
- (4) Operationen sollen (noch) nicht spezifiziert werden.
- (5) Makroprozess: Gehen Sie nach dem in der Vorlesung vorgestellten „Makroprozess“ vor, d.h. in folgender Reihenfolge: Klassen identifizieren – Assoziationen identifizieren – Attribute identifizieren – Vererbung identifizieren – Assoziationen vervollständigen – Attribute spezifizieren (nur Attributtypen).

Datentypen: Die folgenden Datentypen können als gegeben vorausgesetzt werden:

