

## Designbericht für WebAssign

### 1. Allgemein

Es wird eine Designstudie an Hand des OpenSource Produktes WebAssign erarbeitet. Hierbei handelt es sich um eine Plattform, die die Durchführung von Übungsveranstaltungen im Internet ermöglicht und erleichtert. Es können und sollen Aufgaben an die Studenten gestellt werden, welche diese über lösen und ihre Lösungen über das Internet einsenden. Die Korrektur und Bewertung erfolgt ebenfalls auf diesem Wege. Das Produkt unterstützt somit nicht nur die Studenten, sondern auch die Betreuer und Kontrolleure.

WebAssign wird z.B. von der Fernuni Hagen benutzt. Mit diesem Dokument soll ein kurzer Überblick über die Gestaltung und die Implementierung von WebAssign gegeben werden. Zur Erstellung der Diagramme mit Hilfe von Rational Rose wurde die im Praktikum bereitgestellte WebAssign.mdl benutzt.

### 2. Produktübersicht

Bei WebAssign handelt sich um ein Anwendungssystem das ausschließlich über Internet verwendet wird. Die Nutzer, die sich in drei Gruppen aufteilen, greifen über eine Web-Schnittstelle auf das zentrale System zu. Bei den drei Gruppen handelt es sich um Studenten, Betreuer und Korrekteure. Dem Betreuer obliegt die Administration des Kurses. Er schreibt die Studenten und Korrekteure ein und er ist für die Erstellung der Aufgaben und Musterlösungen zuständig. Der Student kann auf die Aufgabenseite zugreifen, welche es ihm ermöglicht Aufgaben zur Vertiefung des Lehrstoffes zu bearbeiten und an den Korrektor zu schicken. Außerdem kann er nach einer festgelegten Zeit die Ergebnis seiner Bearbeitungen abfragen und sich seine korrigierten Aufgaben und die Musterlösungen ansehen. Der Korrektor prüft die vom Studenten eingesandten Lösungen und versieht sie mit Anmerkungen. Danach werden sie dem Studenten wieder zugänglich gemacht. Nach der Korrektur gibt der Korrektor noch die zu jeder Teilaufgabe und insgesamt erhaltenen Punkte an. Durch das Einstellen immer neuer und die Erhaltung der bearbeiteten und korrigierten Aufgaben, sowie die ständig anwachsenden Punkteliste wird eine große Menge an Daten erzeugt. Um diese persistent sichern zu können benutzt WebAssign eine Datenbank.

Das WebAssign-Programm ist Java implementiert und verwendet Servlets. Es ist als notwendig das der Anwender über einen Servlet-unterstützenden Webserver verfügt. Wie bereits erwähnt werden die Daten in einer Datenbank gespeichert. Es wird ein relationales Datenbankmanagementsystem mit JDBC-Schnittstelle genutzt. Der Hersteller bezieht sich hierbei auf die Open Source Software „MySQL“.

### 3. Layout und logischer Aufbau

Im folgenden soll der Aufbau und das Design wie es sich den drei Benutzergruppen darstellt erläutert werden. Zu Beginn befindet sich jeder Nutzer auf der allgemeinen Kursstartseite die dem jeweiligen Kurs angepasst ist oder werden kann. Für deren Zugang ist allerdings keine gesonderte Benutzererkennung notwendig. Ausgehend von der Einstiegsseite können nun alle anderen erreicht werden. An dieser Stelle wird die Zugangsberechtigung geprüft.

Die weitere Gliederung ist der Beschreibung der Fernuni-Hagen entnommen:

- Auf der Studentenstartseite sieht man eine nach Aufgabenheften geordnete Übersicht aller schon bearbeiteten bzw. noch zu bearbeitenden Aufgaben. Für Aufgabenhefte, deren Bearbeitungszeitraum noch nicht begonnen hat, sind die einzelnen Aufgaben noch nicht zugänglich. Über die jeweiligen Aufgaben-Links gelangt man zum eigentlichen Aufgabentext. Die Studentenstartseite kann den Erfordernissen des jeweiligen Kurses angepasst werden.
- Auf der Aufgabenseite befindet sich der eigentliche Aufgabentext. Durch das Eintragen von Lösungen in die dafür vorgesehenen Felder wird die Aufgabe bearbeitet und durch Drücken des Einsendeknopfes zum WebAssign-Server geschickt. Endgültig abgeschlossen ist die Bearbeitung der Aufgabe jedoch erst, wenn das gesamte Heft geschlossen oder das

Bearbeitungsende überschritten ist. Bis dahin kann die Aufgabe beliebig oft eingesandt werden.

- Zur Überprüfung seiner persönlichen Ergebnisse erhält der Studierende eine Auflistung seiner Korrekturen mit deren Status und ggf. den erreichten Punkten sowie Links zu den einzelnen Musterlösungen. Diese Seite ist nur ihm zugänglich.
- Die Betreuerstartseite ist die zentrale Anlaufstelle für Kursbetreuer. Von hier aus können alle administrativen Funktionen erreicht werden. Beispielsweise findet sich hier eine Aufgabenübersicht, die zu Konfigurationsseiten für die einzelnen Aufgaben führt. Eine Einsendungs- und Korrekturübersicht gibt einen nach Aufgabenheften und Korrektoren geordneten Überblick, wieviele Aufgaben welchen Status haben. Die Betreuerstartseite und deren Folgeseiten sind passwortgeschützt und nur für Betreuer zugänglich.
- Auf der Korrektorstartseite sieht ein Korrektor die ihm zugeordneten Aufgaben im Überblick. Nach Aufgabenheften geordnet wird angezeigt, wieviele Aufgaben welchen Status haben. Sie ist ebenfalls passwortgeschützt und nur für Korrektoren zugänglich.

Der Betreuer ist für das Einstellen der Aufgaben zuständig. Hierfür bietet das WebAssign eine gute Unterstützung:

- Der Betreuer kann die Aufgabenseiten mit Hilfe eines beliebigen HTML-Editors gestaltet und, unter Berücksichtigung einiger Vorschriften, beliebige Medientypen, Rahmenstrukturen oder zusätzliche Seiten benutzt.
- Über eine einfache Funktion wird das Aufgabematerial ins System geladen. Dabei wird automatisch die Aufgabenstruktur überprüft und die Datenbank konfiguriert.
- Über vorgefertigte Auswahlflächen und Fenster kann nun die Bearbeitungszeit und die Korrekturart angegeben werden. Außerdem legt der Betreuer hier die Empfangsbestätigung und die Rückmeldeseiten fest.

Es stehen verschiedene Optionen für Korrekturen zur Verfügung: Vorkorrektur, abschließende Korrektur, automatische und manuelle Korrektur.

- die Vorkorrektur wird vom System bereits bei Einsenden der Lösung vorgenommen. WebAssign unterstützt verschieden automatische Korrekturen.
- die abschließende Korrektur erfolgt nach Beendigung der Bearbeitungszeit. Sie kann entweder manuell oder automatisch erfolgen. Für einige Aufgabenarten ist auch eine Kombination dieser beiden Möglichkeiten sinnvoll: Die Lösung wird zunächst automatisch überprüft; danach wird sie manuell mit Anmerkungen versehen und bewertet.
- automatische Korrekturtypen, die von WebAssign unterstützt werden sind: Ja/Nein-Fragen, Multiple-Choice-Aufgaben und Aufgaben mit numerischen Antworten. Es kann als sowohl als Vorkorrektur als auch als abschließende genutzt werden. Für kompliziertere automatische Bewertungen können individuelle Korrekturmodule (in der Programmiersprache Java) implementiert und dann ins System geladen werden. Insbesondere bei Programmieraufgaben werden solche individuellen Korrekturmodule eingesetzt. Studenten senden (Teile von) Computerprogrammen ein, die vom System kompiliert und getestet werden.
- die manuelle Korrektur erfolgt am Ende. der Korrektor schreibt seine Anmerkungen direkt in die Einsendungen des Studenten. Dies geschieht über html-Dokumente.

Der Ablauf einer Aufgabebearbeitung geht wie folgt vor sich:

- der Betreuer stellt mit Hilfe von WebAssign seine Aufgaben und die entsprechenden Konfigurationen ins Internet.
- der Student gelangt über die Startseite auf die Seite der aufgelisteten Aufgabenhefte. Hier wählt er eines aus, dessen Bearbeitungszeitraum bereits begonnen hat (und natürlich noch nicht beendet ist)
- Während dieses Bearbeitungszeitraumes kann der Student die Aufgabe bearbeiten und an das WebAssign-System senden. Die Einsendung wird gespeichert und direkt vom System bestätigt. Wenn für die Aufgabe eine Vorkorrektur eingestellt wurde, testet das System die Lösungen und teilt die Testergebnisse zusammen mit der Bestätigung mit. Wenn nötig kann

der Student nun mehrmals seine Lösungen verbessern und diese dann wieder ans System übergeben, solange bis die Bearbeitungsphase des Aufgabenheftes beendet ist.

- Ist die Bearbeitungsfrist abgelaufen, erstellt das System eine sogenannte Rückmeldungsseite, die die Lösungsbestandteile eines Studenten enthält und ordnet diese einem Korrektor zu.
- Der Korrektor bewertet die studentischen Einsendungen online oder offline. Im online-Modus kann er direkt auf die Rückmeldungsseiten zugreifen, sie mit Anmerkungen versehen und Punkte vergeben. Im offline-Modus muss er sich die Rückmeldungsseiten zunächst herunterladen, sie bewerten und anschließend zum WebAssign-System hochladen.
- Anschließend überprüft der Betreuer die Arbeit des Korrektors und gibt die Korrektur frei.
- Der Student erhält eine automatisch generierte E-Mail, dass er auf seine Korrekturen zugreifen kann. Er kann nun seine Punkte und korrigierten Lösungen ansehen.

#### 4. Paket- und Klassenstruktur

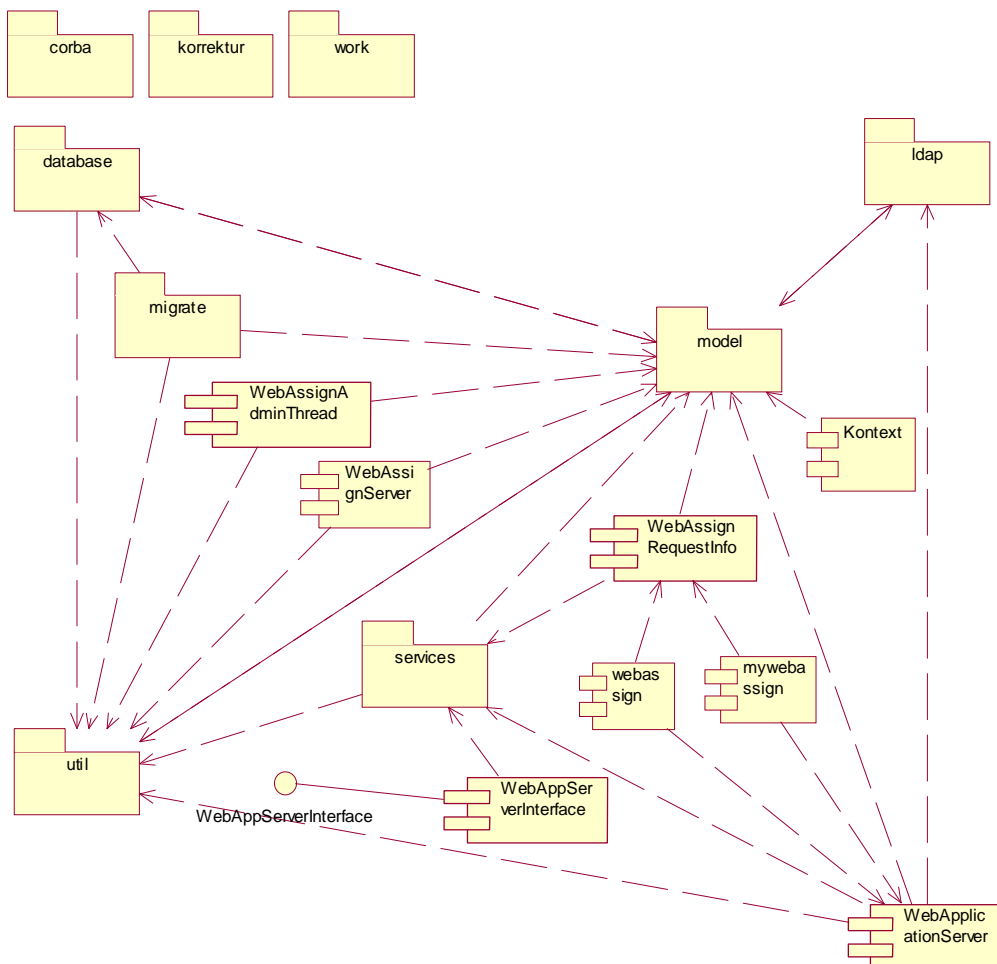
Hier befinden sich die Diagramme zur Verdeutlichung der Zusammenhänge der implementierten Klassen. Bei den Paketstrukturen wurde nur auf WebAssign eingegangen. Im Main-Diagramm sind nur die Hauptstrukturen mit ihren Verbindungen dargestellt. Sollte eine Beziehung zwischen einer Unterstruktur und einer anderen Haupt- oder Unterstruktur bestehen, so wird dies in den einzelnen Paketdiagrammen berücksichtigt.

Die Klassendiagramme beginnen mit dem Main-Diagramm von WebAssign. Die Weiteren werden in der logischer Reihenfolge der Pakete angezeigt.

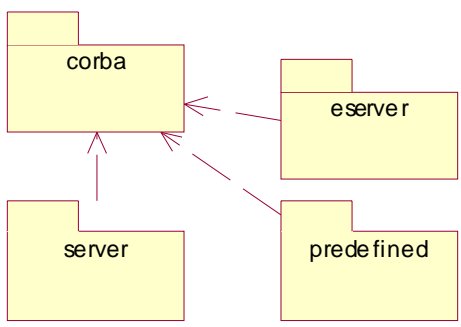
Aus Platzgründen werden in den Diagrammen nur die Klassen oder Pakete aufgeführt, die zu irgendeiner/einem anderen in Verbindung stehen. Die vollständigen Diagramme befinden sich in der WebAssign.mdl.

##### a) Paketstrukturen

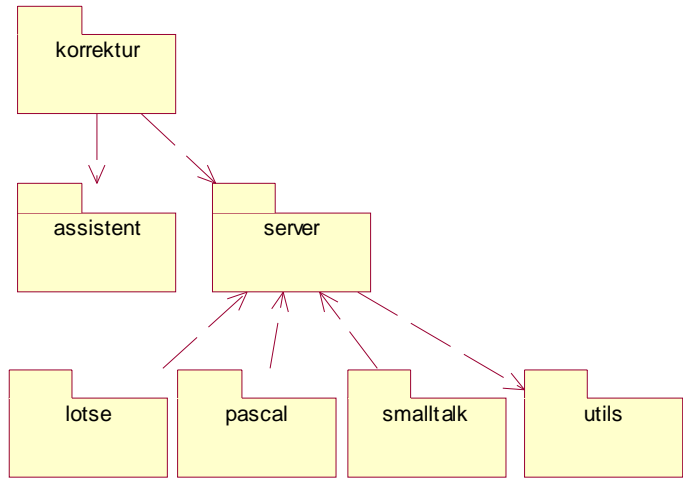
Main:



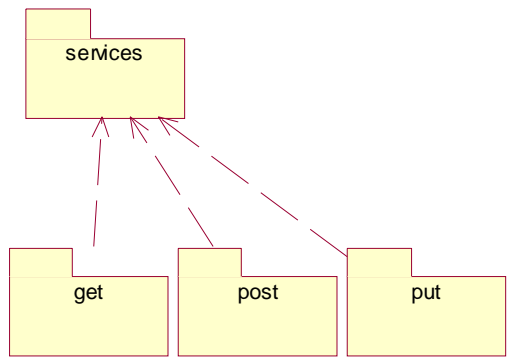
I. corba



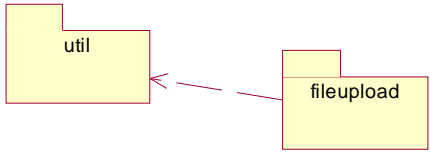
II. korrektur und server



III. services

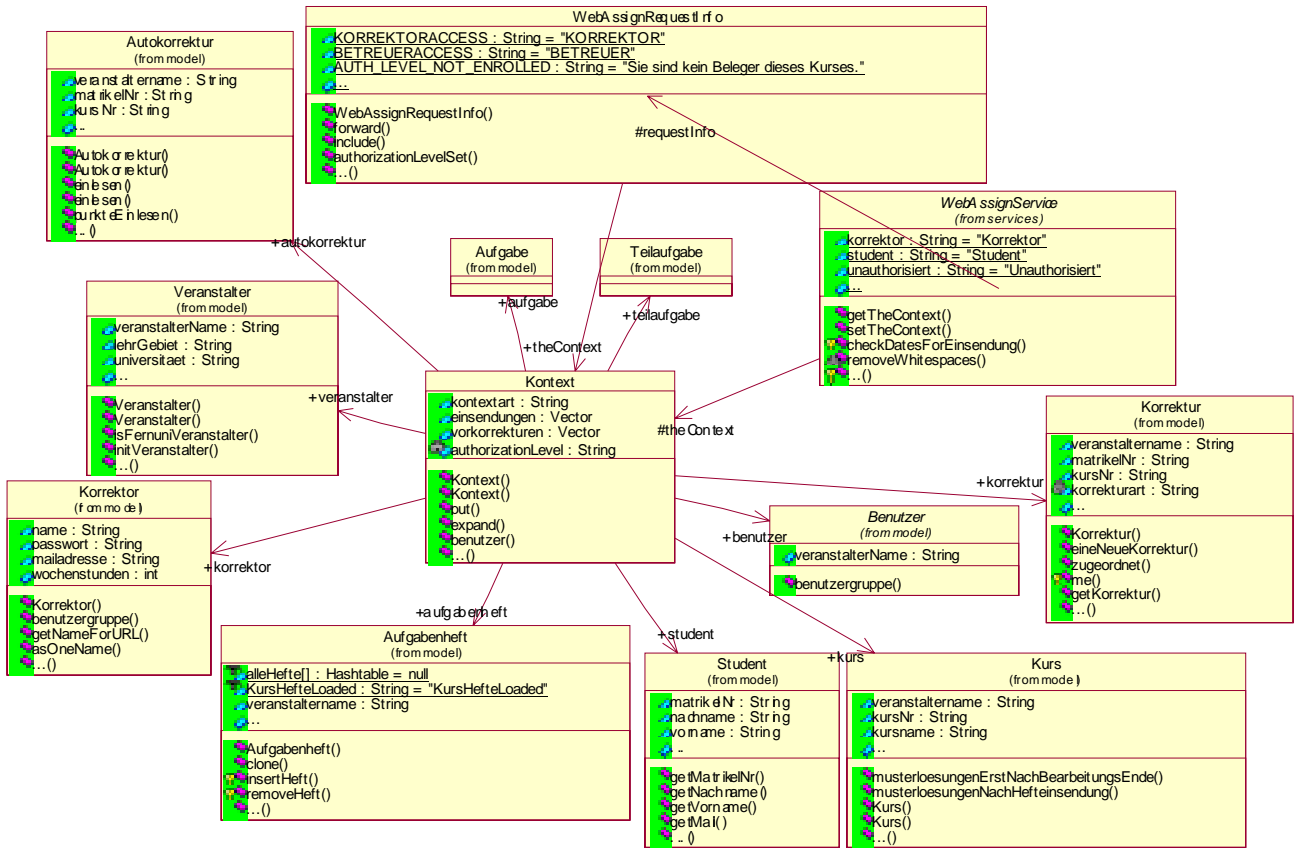


IV. util



b) Klassenstrukturen

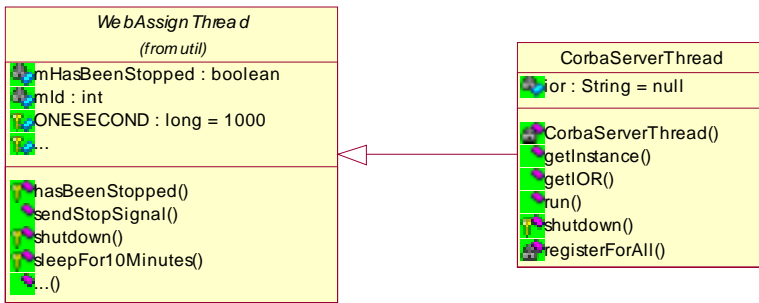
WebAssign-Class-Diagramm



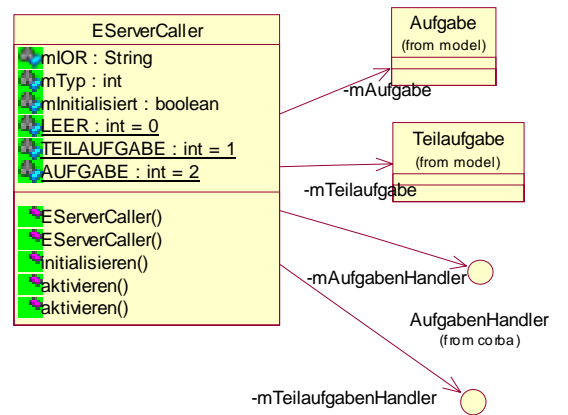
I.corba



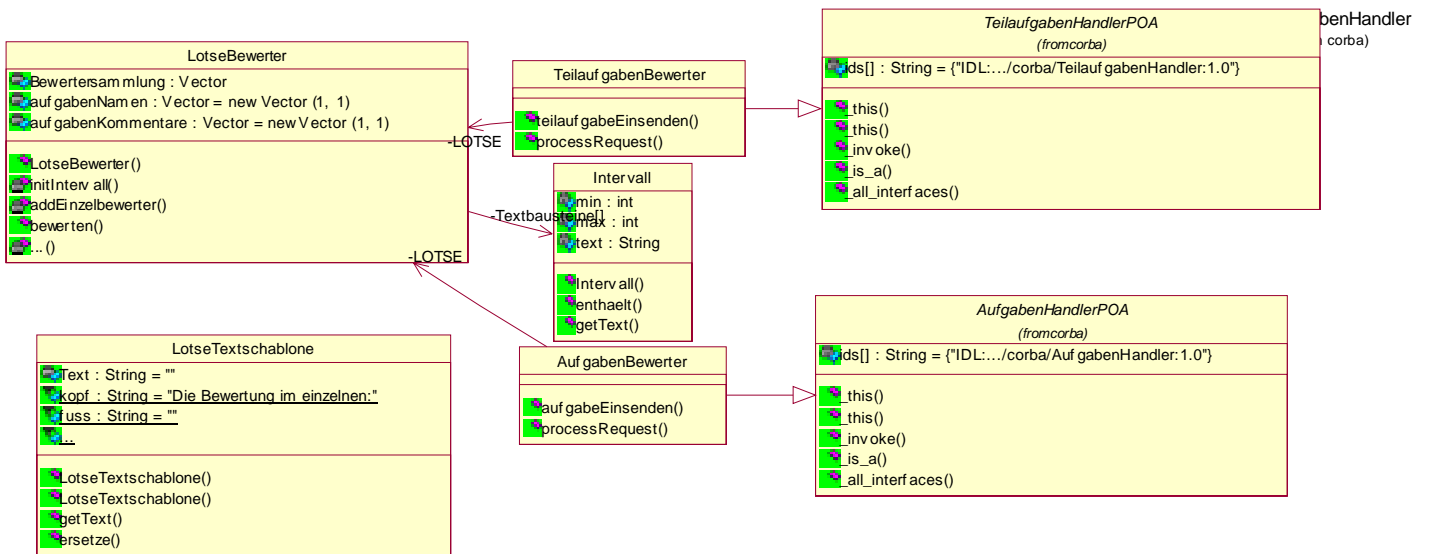
1) server



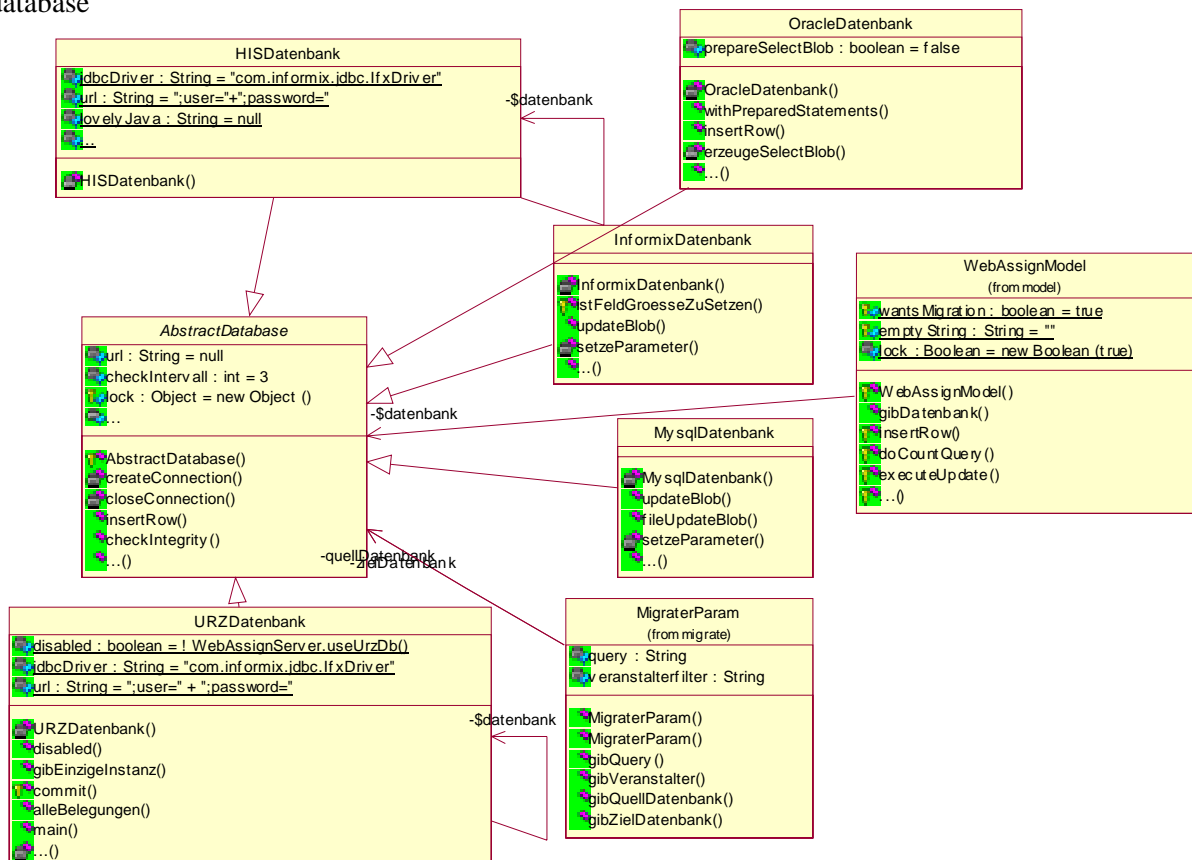
2) eserver



3) predefined

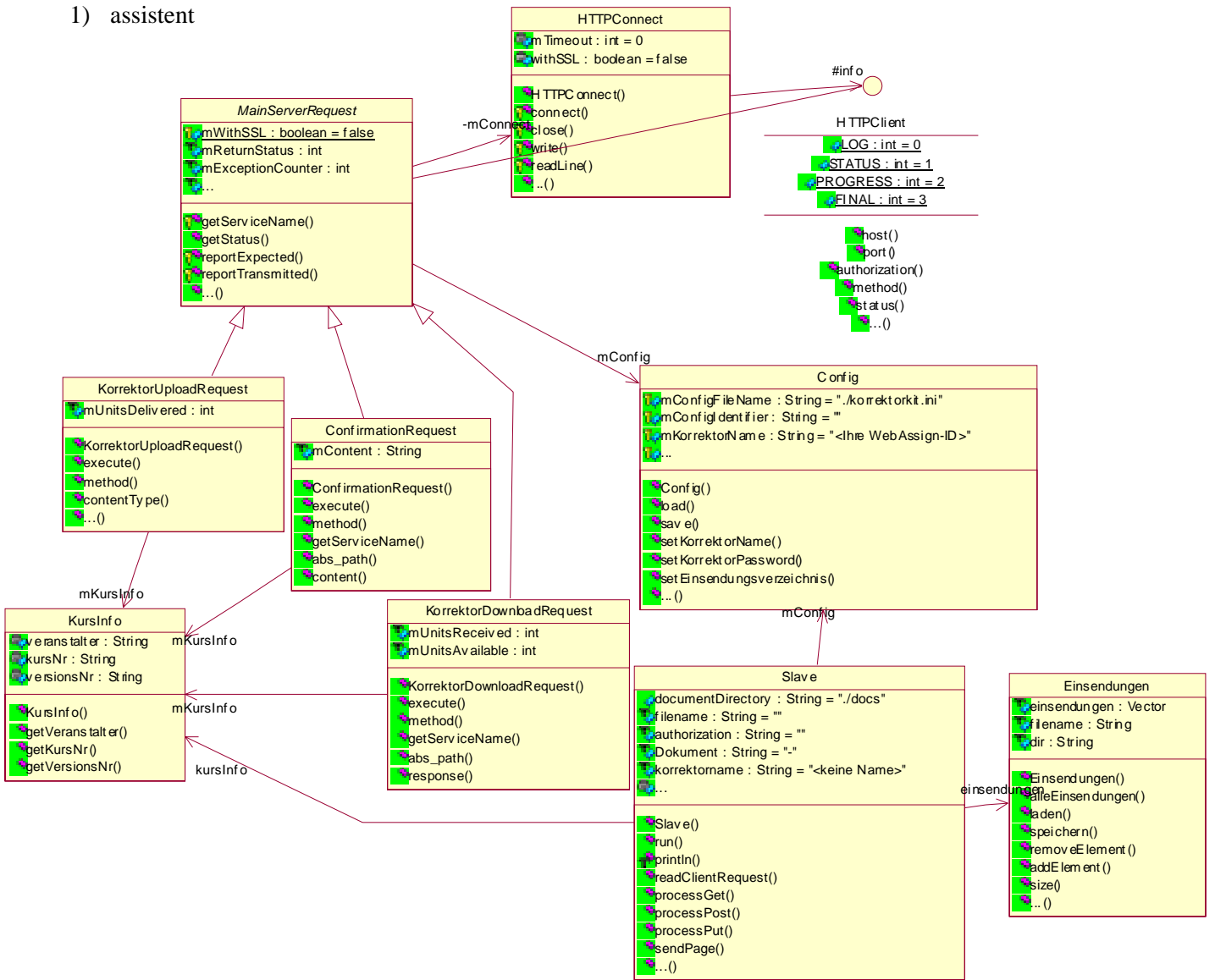


II. database

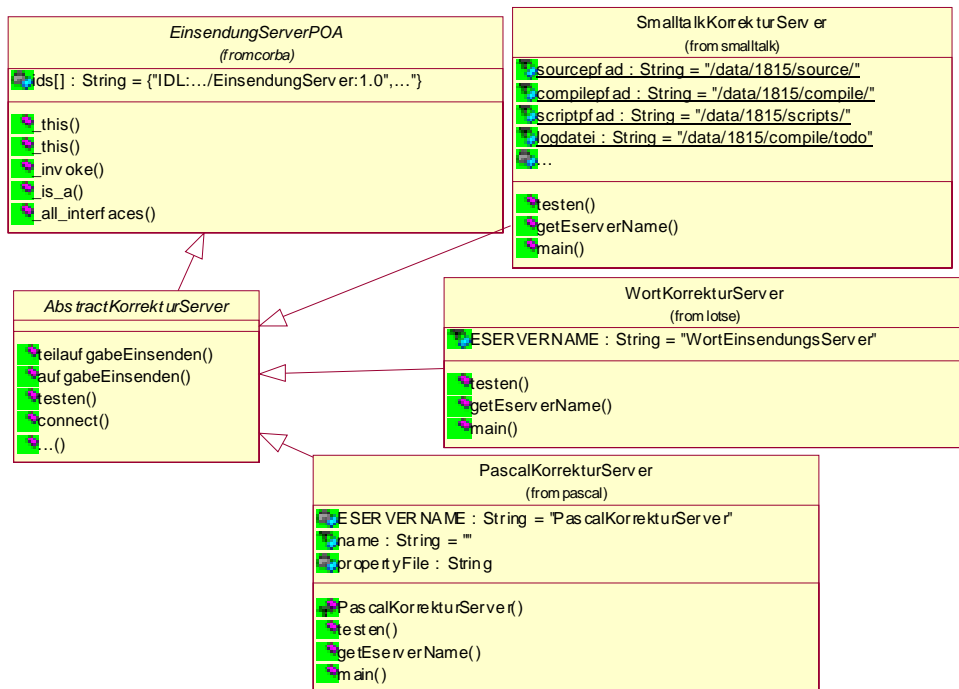


III. korrektur

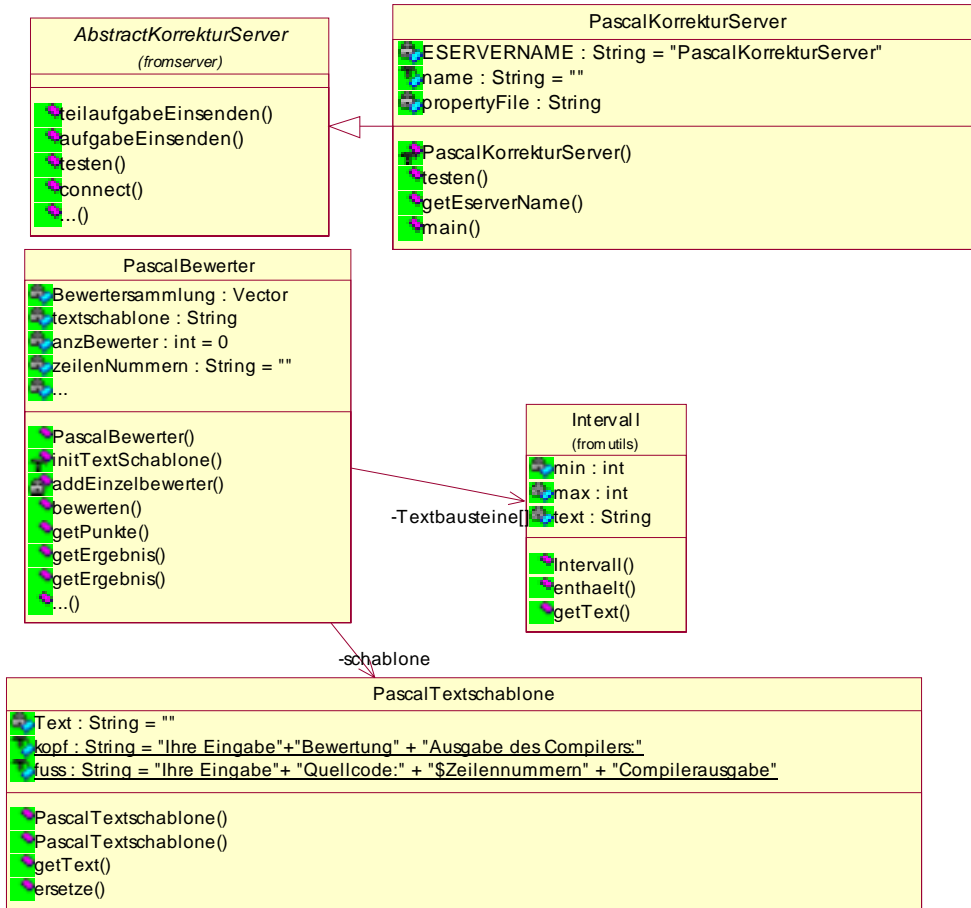
1) assistant



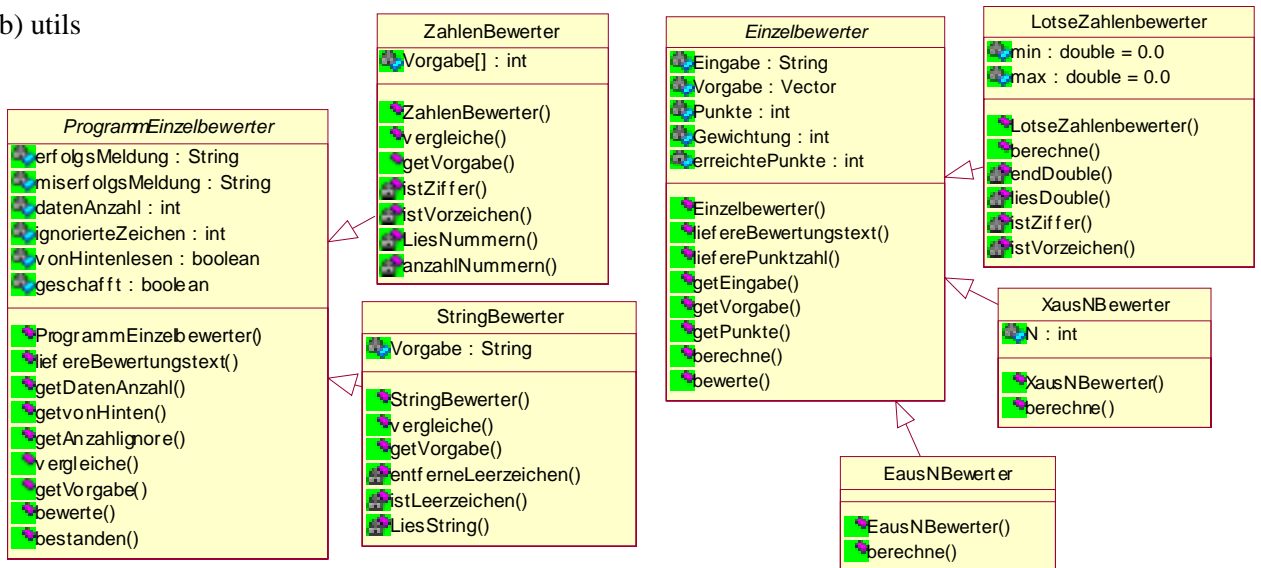
2) server



2a) pascal

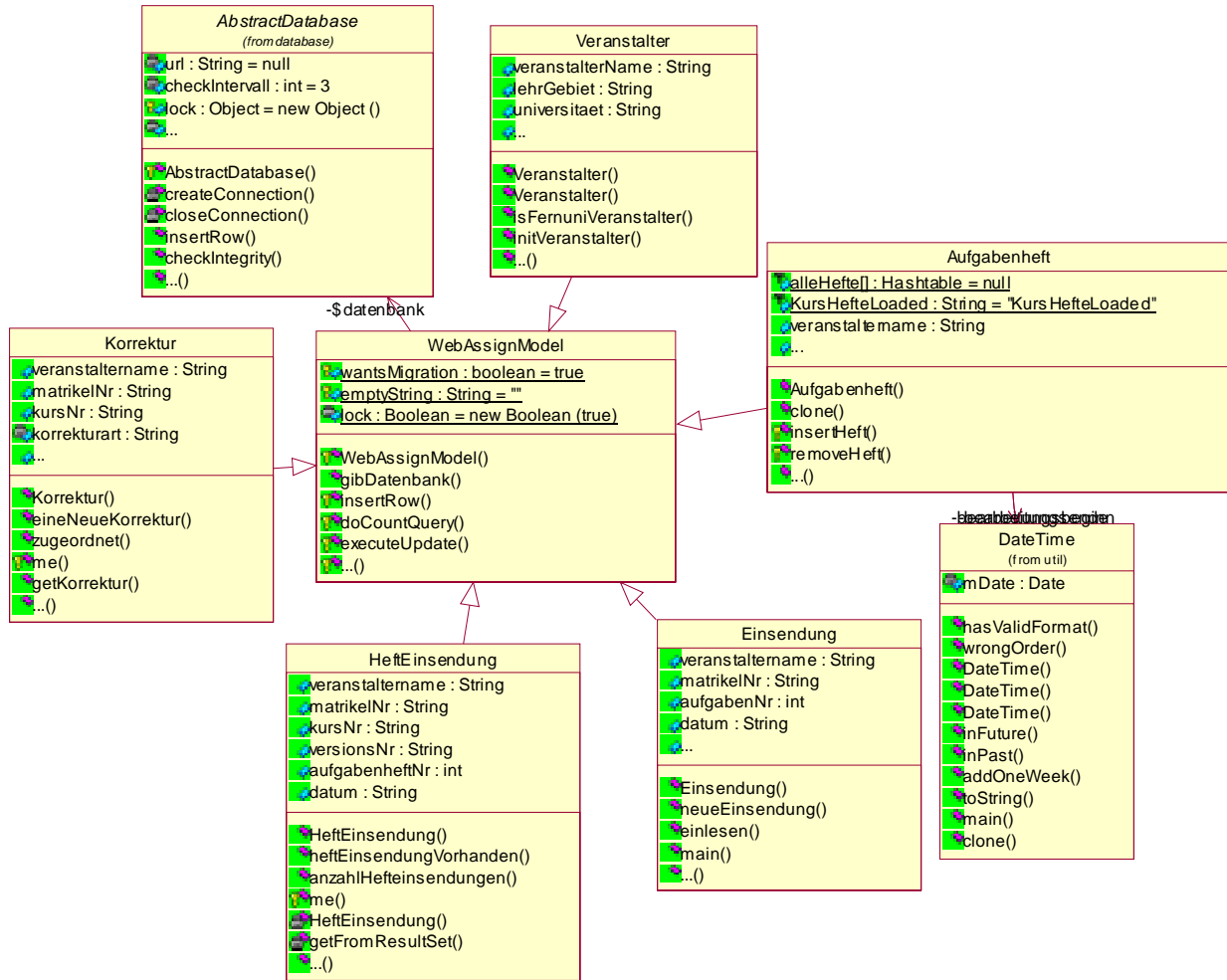


2b) utils

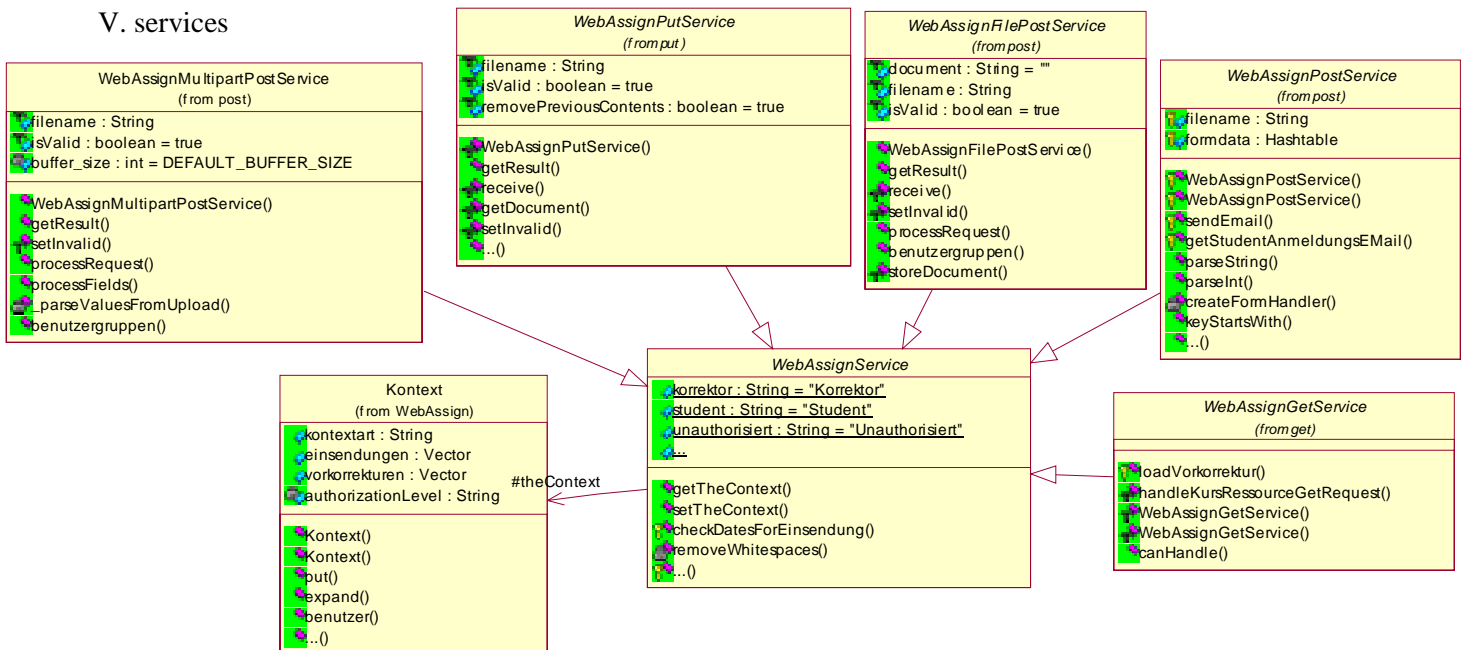




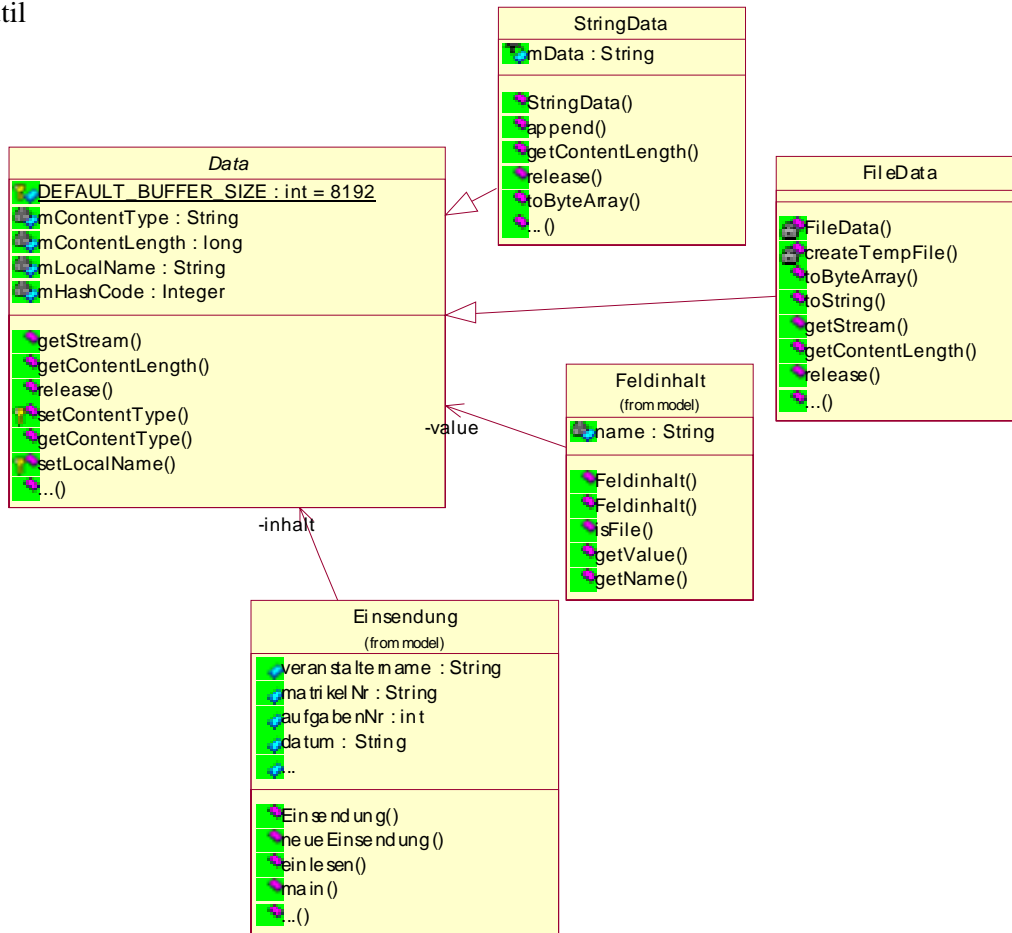
IV. model



V. services



VI. util



VII. work

