

Aufgabenblatt 3

Ziel dieses Blatts

Zur weiteren Vorbereitung auf die Arbeit an Ihrem eigenen Projekt sollen Sie sich mit diesem Arbeitsblatt in eine vorgegebene Applikation ähnlicher Ausrichtung einarbeiten, zu welcher der Quellcode verfügbar ist. Diese Applikation ist in einem **themenspezifischen Begleitblatt** näher beschrieben. Dabei geht es

- um den Einsatz von Rational Rose als Werkzeug zum Re-Engineering,
- um die konzeptionelle Einarbeitung in fremden Quellcode, zu dem wenig weiterführende Dokumentation vorliegt, und
- um die weitere Vertiefung des Verständnisses der Fachkonzepte, die in Ihrem Projekt umzusetzen sind.

Geeignete Teile des studierten Codes können Sie in das eigene Projekt übernehmen.

Reverse Engineering eines Java-Projekts mit Rational Rose (RR)

Um aus den Quellen eines Java-Projekts ein RR-Modell zu erstellen steht die Option **Tools > Java > Reverse Engineer...** zur Verfügung. Starten Sie dazu ein neues Projekt (ohne alle Java-Klassen), ändern die Defaultsprache (Menü **Tools > Model Properties > Edit > Notation**) auf Java (siehe auch den Eintrag in der unteren Statuszeile von RR) und starten den Prozess des Reverse Engineering (**Tools > Java > Reverse Engineer...**). Nun erscheint ein Fenster, in dem zunächst die erforderlichen Parameter eingestellt werden. Mit **> Edit CLASSPATH** wird der Pfad der zu bearbeitenden Dateien festgelegt. Neben den Klassenbibliotheken des J2SDK muss hier das Wurzelverzeichnis der Quellen des zu untersuchenden Projekts eingebunden werden. Dieses erscheint dann zusammen mit den Verzeichnissen des J2SDK im linken Fensterteil. Markieren Sie dieses Verzeichnis und drücken **> Add Recursive**, um alle *.java-Quelldateien zum Reverse Engineering vorzumerken. Diese werden danach im unteren Fensterteil aufgelistet. Nun können Sie **> Reverse** starten. RR erstellt aus den Quellen ein neues Projekt und bindet zugleich die erforderlichen (aus den import-Deklarationen ersichtlichen) Java-Pakete mit ein.

Für das Projekt werden dabei zunächst die Klassendefinitionen, geordnet nach Paketen, als ‚Logical View‘ und die Pakete selbst als ‚Component View‘ ins Modell übernommen und im (linken) Browserfenster angelegt. Geben Sie dem Projekt nun einen Namen und sichern Sie das Ergebnis.

Nun können Sie dazu im Browserfenster verschiedene Diagrammsymbole erzeugen, indem Sie mit der Maus auf einen ‚View‘ gehen, mit der rechten Maustaste ein Menü öffnen und dort unter **> New** ein neues Diagrammsymbol zunächst im Browserfenster (und damit im Modell) anlegen. Zwei Diagrammsymbole jeweils mit dem Namen ‚Main‘ sind bereits angelegt. Sie stehen für eine Darstellung der grundsätzlichen Klassen- bzw. Paketstruktur. Bitte überfrachten Sie diese Darstellung aber nicht mit zu vielen Details bis zur Unübersichtlichkeit, sondern stellen einzelne Details des Designs in weiteren Diagrammen dar.

Ist ein solches Diagrammsymbol im Browserfenster angelegt, öffnen Sie die zugehörige (zunächst leere) grafische Darstellung im (rechten) Diagrammfenster, indem Sie über dem Symbol mit der rechten Maustaste ein Menü öffnen und dort **> Open** wählen. Durch Ziehen mit der Maus können Sie dann einzelne Klassen oder Pakete (Objekte) aus dem Browserfenster in das Diagrammfenster übertragen und ins entsprechenden Diagramm importieren.

Importieren Sie mehrere Objekte, so werden automatisch die zwischen ihnen bekannten Beziehungen mit dargestellt. Sie müssen das Bild aber mglw. zurechtrücken. Einzelne Beziehungen können aus Übersichtsgründen im Diagramm gelöscht werden. Beachten Sie den Unterschied zwischen Löschen im Diagramm und Löschen im Browserfenster (und damit im Modell). Mit der rechten Maustaste über einem Objektsymbol im Browserfenster aktivieren Sie ein Menü, das unter **> Java > Edit Code** integrierten Zugriff auf den jeweiligen Quellcode gibt. Mit diesen beiden Optionen bekommt man schnell einen Überblick über die innere Logik nicht zu umfangreicher Projekte.

Das Zusammenspiel mehrerer Klassen aus unterschiedlichen Paketen kann erforscht werden, indem diese Klassen in ein gemeinsames Diagramm geladen werden.

Beachten Sie, dass CVS Probleme beim Patchen einer von RR angelegten mdl-Datei bereiten kann. Erstellen Sie vor einem cvs-Update deshalb immer eine Sicherungskopie ihrer aktuellen mdl-Datei!

Aufgabe:

Führen Sie ein Reverse Engineering des im Begleitblatt zu Ihrem Thema angegebenen Projekts durch, erforschen Sie die Paket- und Klassenstruktur und fertigen Sie eine Design-Beschreibung an, welche der aus Aufgabenblatt 2 bekannten Gliederung folgt. Erläutern Sie dabei die verschiedenen Aspekte, die im Projektdesign berücksichtigt wurden, durch mit RR erstellte Diagramme (geeigneten Typs), die Sie in die Designbeschreibung importieren. Zur Beschreibung der funktionalen Aspekte (Abschnitt 1 und 2) können Sie sich an den im Beiblatt gegebenen Informationen orientieren bzw. diese weitgehend übernehmen.

Abzugeben sind:

- Die Designbeschreibung des Prototyps (maximal 10 Seiten mit Schwerpunkt auf den Abschnitten 3 und 4) als ps- oder pdf-Datei
- mdl-Datei des durch Reverse Engineering erzeugten Rose-Projekts

Abschluss der Anforderungsanalyse

Aufgabe:

Fertigen Sie auf der Basis von Glossar und Lastenheft sowie den Festlegungen der ersten Inspektion das Pflichtenheft zu Ihrem Projekt an. Halten Sie sich dabei eng an die Gliederungsforderungen in [Balzert].

Das Pflichtenheft soll die Funktionalität beschreiben, die Sie im Rahmen des Projekts mindestens implementieren wollen. Aktualisieren Sie dabei das Glossar.

Bearbeiten Sie das Geometriethema, so können Sie auf die Angabe eines Akteurs verzichten, da dies stets der Nutzer der Einzelplatzapplikation ist. Verwenden Sie in diesem Fall für die Beschreibung der Funktionen im Abschnitt „4. Produktfunktionen“ (Use Cases = Anwendungsfälle sind hier identisch mit „Menüeinträge“) die folgende Gliederung: Vorbedingung, Auslösendes Ereignis, Beschreibung, Erwartetes Verhalten, Nachbedingung, Ausnahmen.

Abzugeben: Glossar und Pflichtenheft, jeweils als ps- oder pdf-Datei

Abgabe der Materialien zu diesem Arbeitsblatt

bis zum 18.5. wie üblich durch Hinterlegen aller geforderten Materialien im Verzeichnis `submit` Ihres Gruppenaccounts auf `pcai003.informatik.uni-leipzig.de`. Beachten Sie, dass die eingereichten Materialien nach dem Ablauf der Bearbeitungsfrist aus diesem Verzeichnis **verschoben** werden.