
[Meeting Protokoll Woche 4]

Thema	Wöchentliches GVS Meeting
Ort	Raum 5.207 (HSR)
Datum	11.10.2017
Uhrzeit	17:15 - 18:15
Teilnehmer	<ul style="list-style-type: none">• Thomas Letsch• Murièle Trentini• Michael Wieland

1 Rückblick

1. Das Projektteam hat GVS 1.0 analysiert und dabei folgende Übersichten erstellt:
 - Sequenzdiagramme** Dienen dem Verständnis, wie die eingehenden XML Files in Java Objekte transformiert und angezeigt werden. Sie zeigen alle relevanten beteiligten Klassen.
 - Domainmodell GVS 1.0** Enthält die wichtigsten Konzepte aus der alten Applikation.
 - Domainmodell GVS 2.0** Enthält die Konzepte, welche in der Vorstellung des Projektteams notwendig sind. Zeigt einen ersten Entwurf einer sauberen Schichtenarchitektur.
 - Migrations-Übersicht GVS 2.0** Stellt die Klassen und Packages von GVS 1.0 den Klassen und Packages von GVS 2.0 gegenüber. In der Übersicht ist ersichtlich, welche Klassen mit welchen neuen Klassen ersetzt werden.
2. Beim Laden einer neuen Session wird in der Klasse "Persistor" XML geparkt. Eine sehr ähnliche Funktionalität bietet die Klasse "ModelBuilder". Das Parsen von XML soll neu einmalig in der Klasse "XMLParser" stattfinden.
3. Die Interface Vererbungshierarchie scheint in GVS 1.0 relativ umfangreich zu sein. Viele Hierarchiestufen verkomplizieren das Design. Beispiele:
 - a) Das Interface "ISessionController" unterteilt sich in die beiden Interfaces "IGraphSessionController" und "ITreeSessionController". Von diesen gibt es dann je eine konkrete Instanz "GraphSessionController" rsp. "TreeSessionController".

- b) Das Interface "IVertex" unterteilt sich in die beiden Interfaces "IDefaultVertex" und "IIconVertex". Von diesen gibt es dann je eine konkrete Instanz "DefaultVertex" resp. "IconVertex".
- 4. Die Begrifflichkeiten bezüglich Session und Model wurden geklärt. Die Namen sollen in GVS 2.0 entweder besser gewählt oder in JavaDoc ausführlicher beschrieben werden.
 - a) GraphModel = Entspricht einer Momentaufnahme des Graphen/Trees.
 - b) Session = Mehrere "GraphModels" gehören einer Session an.

2 Aktuelles

1. Die Physik Engine (Force Field Algorithmus) muss noch genauer analysiert werden
2. Neu soll im Business Layer eine Style-Klasse (DTO) für Nodes und Edges erzeugt werden. Diese repräsentiert das Styling im Business Layer, ohne spezifische Imports aus dem eingesetzten UI Framework. (Swing resp. neu JavaFX)

3 Beschlüsse

1. Korrektur Meeting Protokoll 3, Beschlüsse 1.b: Designentscheide werden nicht im Anforderungsdokument sondern in der Designspezifikation niedergeschrieben. Es wird nur ein Requirement geben. In etwa: "Swing soll durch JavaFX ersetzt werden unter Beibehaltung des Funktionsumfangs."
2. Das Projektteam muss sich für eine der folgenden Varianten entscheiden
 - Variante 1: Swing Komponenten durch JavaFX Komponenten ersetzen und dabei so viel wie nur möglich wiederverwenden.
 - Variante 2: Sauberes Layering einführen. Dies hat den klaren Vorteil, dass zukünftige Erweiterungen an GVS 2.0 leichter durchgeführt werden können. Es besteht aber die Gefahr, dass dieses "Extended Refactoring" den Rahmen der Studienarbeit sprengt.
3. Der beschriebenen Funktionsumfang im Benutzerhandbuch von GVS 1.0 muss beibehalten werden. Anpassungen dürfen nur nach Absprache mit Thomas Letsch durchgeführt werden.
4. Folgende Komponenten können in den GVS 2.0 mit tiefer Priorität migriert werden.
 - Cluster Splitter
 - Icon Support auf den Nodes
 - Draggable Support für Nodes und Edges
5. Das Projektteam erhält von Herrn Letsch alle Übungsaufgaben aus dem Modul AD2, in welchen GVS 1.0 eingesetzt wird.

4 Ausblick

1. GVS 1.0 wird in der nächsten Woche noch einmal genau analysiert. Am Ende der Woche soll klar sein, für welche der beschriebenen Varianten man sich entschieden hat.
2. Es muss noch einmal genau untersucht werden, ob ein Zusammenzug der Graph und Tree Repräsentation möglich ist. (Falls man sich für Variante 2 entscheidet)

5 Nächster Termin

Termin	18.10.2017 - 17:15
Ort	Raum 5.207 (HSR)
Bemerkungen	-
