

# Entwurfsbeschreibung

Universität Leipzig – Softwarepraktikum 2014

## Semantic Chess

Jonas Herrig

Hanno Krümpelmann

Nathalie Bargenda

Duc Hieu Nguyen

Stefan Süßmaier

Lisa Höncke

## Inhaltsverzeichnis

1. Allgemeines.....	3
2. Produktübersicht.....	3
3. Grundsätzliche Struktur- und Entwurfsprinzipien.....	3
4. Struktur- und Entwurfsprinzipien.....	5
5. Datenmodell.....	6
6. Testkonzept.....	7
7. Glossar.....	7
8. Quellen.....	9

## 1. Allgemeines

In diesem Dokument soll das System zum Verarbeiten von Anfragen an eine RDF-Datenbank vorgestellt werden, welches als Prototyp im Rahmen unseres Vorprojekts zum Thema Semantic Chess erstellt wurde. Es war bis auf wenige Details dabei möglich, die gesetzte Planung der Klassenstruktur umzusetzen.

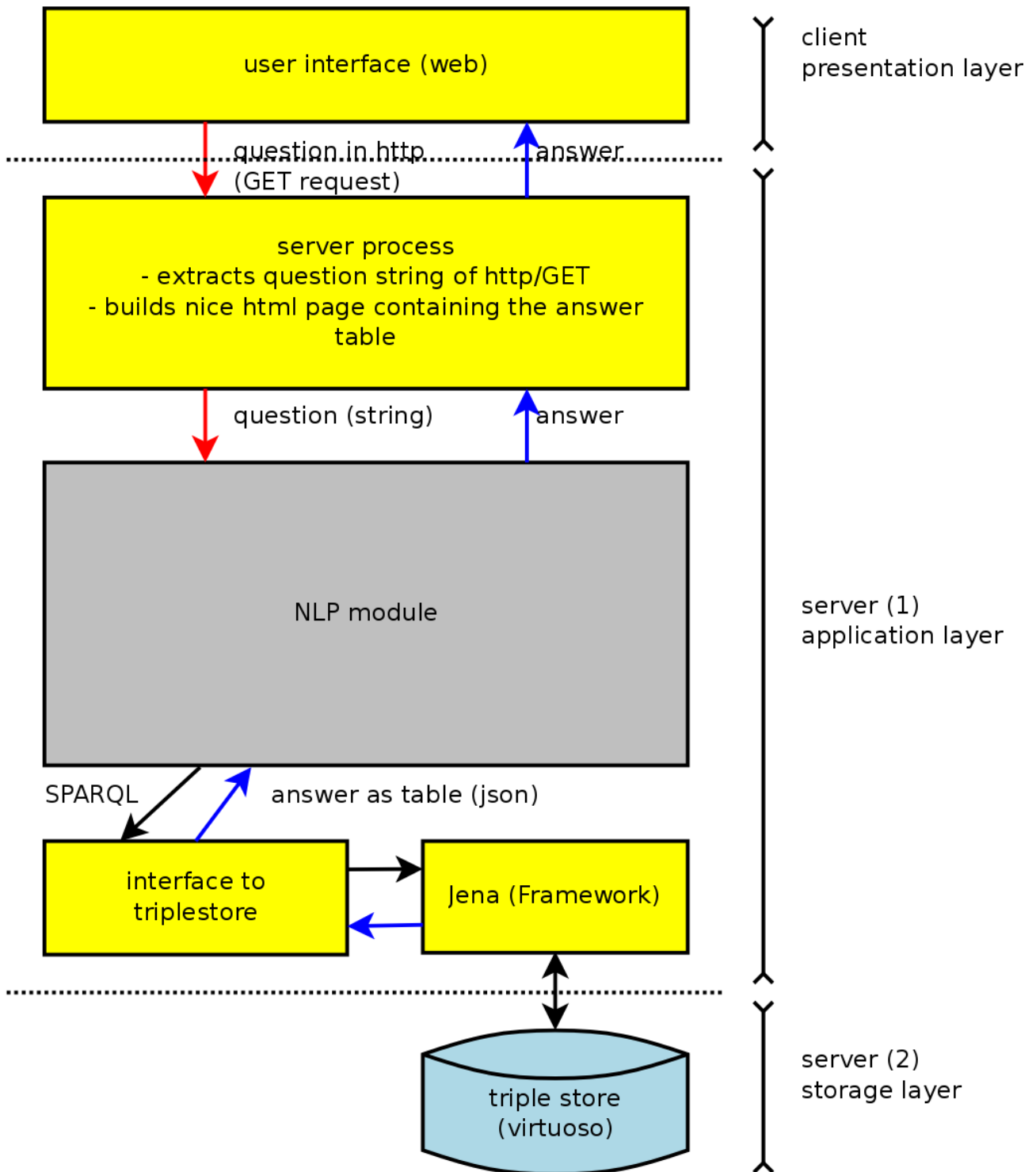
## 2. Produktübersicht

Die Software setzt im wesentlichen eine Funktion um: über eine simple Weboberfläche kann der User aus einer Liste vorgefertigter Anfragen an eine Schachdatenbank wählen. Diese Anfrage wird an einen Virtuoso-Triplestore weitergeleitet und das Ergebnis wird in Form einer einfachen Tabelle zurückgegeben. Die Antwort erfolgt ohne nennenswerte Verzögerung. Derzeit hat der User keine weiteren Einstellmöglichkeiten außer einem Dropdown-Menü mit den zur Auswahl stehenden Anfragen.

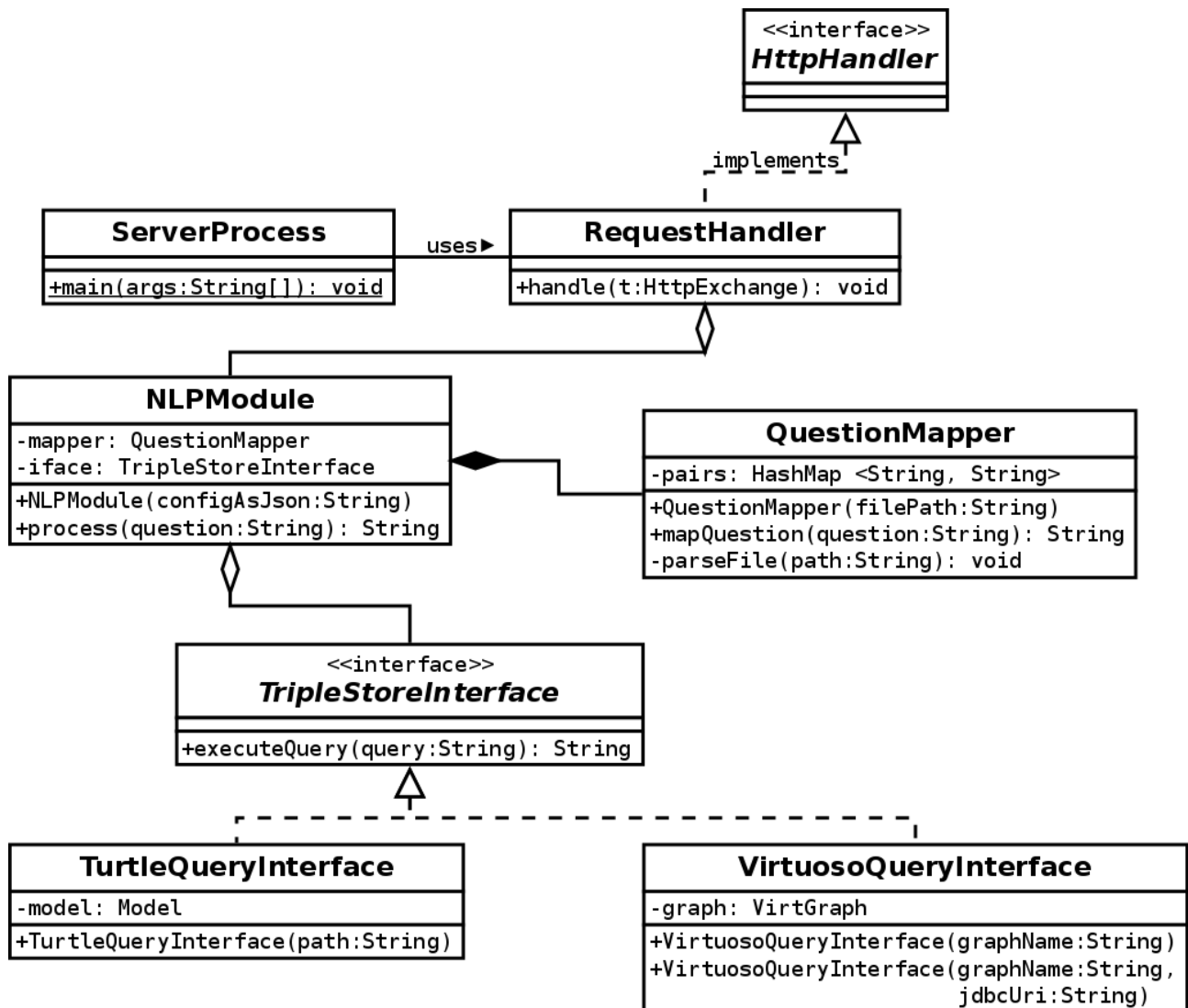
## 3. Grundsätzliche Struktur- und Entwurfsprinzipien

Da es sich bei der Software um eine Webanwendung handelt, die eine textuelle Benutzereingabe in sequenziellen Teilschritten in eine SPARQL-query umwandeln soll, die schließlich mittels einer query engine beantwortet werden kann, wurde eine Schichtenarchitektur gewählt. Diese untergliedert sich grob in 3 Schichten: Eine Clientschicht für die graphische Benutzerschnittstelle und 2 Serverschichten für die Anwendung selbst und den triple store. Diese 3 Schichten können auf 3 verschiedenen Rechnern ausgeführt werden, da die Kommunikation zwischen diesen Schichten über Netzwerkprotokolle erfolgt. Die Anwendungsschicht untergliedert sich ihrerseits wieder in 3 Schichten: Den Serverprozess, das NLP-modul und die Schnittstelle zum triple store. Der Serverprozess agiert als Webserver, der auf Anfragen (http/GET) wartet und eine entsprechende Antwort (als HTML) generiert. Das NLP-modul wandelt eine natürliche Frage in eine SPARQL-query um und führt diese über die Schnittstelle zum triple store aus.

Der Vorteil dieser Architektur ist, dass diese die nötigen Schnittstellen für das Hauptprojekt bereits implementiert, sodass lediglich das NLP-modul neu entwickelt werden muss.



## 4. Struktur- und Entwurfsprinzipien



Die Klasse `ServerProcess` verfügt über die `main`-Methode, in welcher der `http`-Server aus der Java-Standardbibliothek unter Verwendung der Klasse `RequestHandler` gestartet wird. Der `RequestHandler` extrahiert in der Methode `handle` aus dem übergebenen Objekt vom Typ `HttpExchange` die angefragte URI und aus dieser anschließend die gestellte Frage als einfache Zeichenkette.

Dieser `String` wird an die Klasse `NLPModule` übermittelt durch Aufruf der Methode `process`. In `NLPModule` wird ein `QuestionMapper` instanziiert. Dieser dient dazu, die Anfrage im `String`-Format in ein `SPARQL-Query` umzuwandeln, welches von `NLPModule` an das `TripleStoreInterface` gestellt wird.

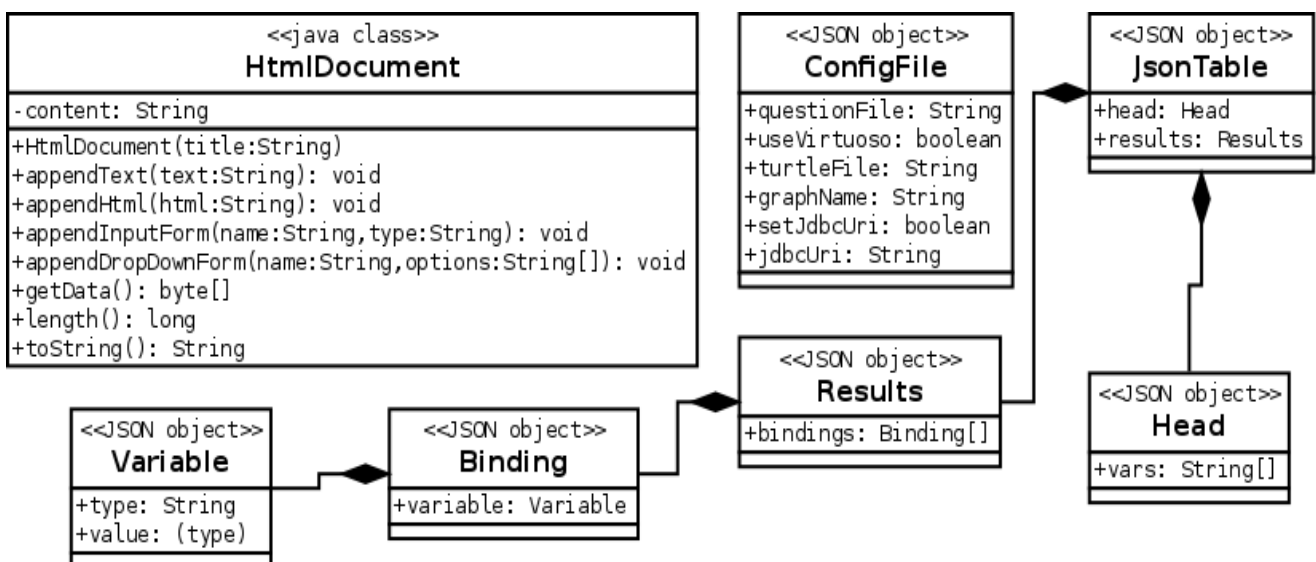
Derzeit ist der QuestionMapper noch relativ primitiv. Durch Einlesen einer Datei im csv-Format in der Methode parseFile wird hier eine HashMap erstellt, welche jedem String statisch genau eine SPARQL-Anfrage zuweist.

Diese Anfrage wird dann über das NLPModule an das TripleStoreInterface übergeben. Dieses setzt die Kommunikation mit dem Triplestore über die zur Verfügung stehende API Jena um. Das Ergebnis der Anfrage wird als JSON-Objekt zurück gegeben.

Die eben einzeln beschriebenen Klassen lassen sich zu Modulen im Sinne der Schichtenarchitektur zusammenfassen. Die Klassen „ServerProcess“ und „RequestHandler“ bilden das Paket „Server“, die Klassen „NLPModule“ und „QuestionMapper“ das Paket „NLPModule“ und die Klassen „TrurtleQueryInterface“ und „VirtuosoQueryInterface“ mitsamt des von ihnen implementierten Interfaces „TripleStoreInterface“ das Paket „TripleStoreInterface“. Die Pakete entsprechen den in 3. beschriebenen Schichten der Anwendungsschicht.

## 5.Datenmodell

Die Java-klasse HtmlDocument dient dem Zweck, serverseitig Html-dokumente zu generieren. Neben der Klasse HtmlDocument und JSON-objekten für die Darstellung von Tabellen und Konfigurationsdaten treten Daten nur in Form einfacher Zeichenketten auf.



## 6. Testkonzept

Wie in unserem Konzept zur Qualitätssicherung festgelegt, werden Komponententests für einzelne Klassen mit Hilfe von JUnit durchgeführt. Zum Integrationstest einzelner Module werden spezielle Testklassen geschrieben, welche typische Testanfragen auf den entsprechenden Methoden der im Modul enthaltenen Klassen ausführen.

Funktionale Systemtests werden manuell durchgeführt, indem manuell geprüft wird, ob die Antworten auf alle möglichen Testanfragen fehlerfrei die gewünschten Informationen enthalten.

## 7. Glossar

### Client-Server-Architektur

Mit Client-Server-Architektur ist gemeint, dass eine Aufgabe im Netzwerk auf Client und Server verteilt wird. Die verwendete Software unterteilt dabei in Client und Server. Der Client fordert dabei vom Server einen Dienst an, welcher entsprechend die Anforderung beantwortet. Client- und Serverteil eines Programms sind meist auf verschiedenen Rechnern eines Netzwerks. Eine auch für dieses Softwareprojekt sinnvolle gängige Unterteilung ist es, die GUI und alle mit Eingaben verbundenen Programmteile auf dem Client laufen zu lassen, jedoch die Datenbank und die Suchalgorithmen auf einem Server zu implementieren.

### Apache Jena

Apache Jena ist ein Open Source-Framework in Java das zur Konstruierung von Semantic Web Anwendungen entwickelt wurde. Es verfügt über eine Programmierschnittstelle zum Speichern und Laden von RDF-Daten. Des Weiteren verfügt die Programmierumgebung über SPARQL, OWL(Web Ontology Language), GRDDL Frameworks. Die generelle Funktionsweise umfasst Repräsentation der RDF-Graphen als abstrakte Modelle im Speicher oder Datenbanken die zum Teil auf OWL beruhen, die Abfrage via SPARQL und die Manipulation durch das SPARUL Modul. Dabei zählt Apache Jena zu den beliebtesten RDF-Frameworks.

### Virtuoso

Virtuoso ist ein „Universaler Datenserver“, der verschiedene Modelle unterstützt. Unter anderem kann Virtuoso als Triplestore für RDF-Daten genutzt werden. An diesen können SPARQL-queries gestellt werden. Im Unterschied zu Jena ist Virtuoso eine eigenständige Serveranwendung.

## SPARQL

Die SPARQL Protocol And RDF Query Language ist eine Abfragesprache für RDF. Hierfür wird eine Engine benötigt, welche die queries verarbeitet. Engines sind z.B. in Apache Jena oder Virtuoso enthalten.

## Triple Store

Ein Triple-Store ist eine Datenbank, in der Daten in Form von Triples gespeichert sind und entsprechend für die Abfrage durch Abfragesprache bereit stehen. Dabei hat ein Tripel immer die Form Subjekt-Prädikat-Objekt.

## Hashmap

Eine Hashmap dient zum Speichern von beliebigen Java Objekten. Dabei wird zu den Objekten (Schlüsseln) jeweils ein Wert berechnet, dank dem es in konstanter Zeit eingefügt und entfernt werden kann. Dabei wird die Hashmap bzw. Hashtabelle auf eine bestimmte Größe ausgerichtet und muss bei überschreiten eines festgelegten Füllgrades neu berechnet werden.

## JSON

JSON (JavaScript Object Notation) ist ein standardisiertes Datenaustauschformat mittels dessen sich Anwendungen unabhängig von ihrer Programmiersprache austauschen können. Dabei werde sowohl Schlüssel/Wert Paare (siehe Hashmap) als auch Listenstrukturen wie Arrays oder Vektoren verwendet. Dabei ist jedes JSON-Dokument ein gültiges Java-Script.

## Bytestream

Als Bytestream (deutsch: Bytestrom) wird eine Sequenz aus 8 Bit großen Bytes bezeichnet. Diese Datenströme werden allgemein von einem Medium ins andere übertragen. In der Regel wird ein Bitstrom zu einem Bytestrom gegliedert und diese gliedern sich weiter in Blöcke und Datenpakete inunteraschiedlicher Protokolle und Formate.

## RDF / Turtle

Das Ressource Description Framework ist ein Modell zum beschreiben von Aussagen bzw. Wissen. Genauer handelt es sich um Tripel der Form Subjekt – Prädikat – Objekt, welche eine Relation zwischen Subjekt und Objekt repräsentieren. Eine Menge solcher Tripel beschreibt somit einen gerichteten Graphen, wobei Subjekte und Objekte die Knoten und Prädikate die Kanten (von Subjekt zu Objekt) bezeichnen. Turtle ist eine spezielle Formatierung von RDF-Daten.



## HTTP

HTTP ist die Abkürzung für Hyper Text Transfer Protocol ist ein zustandsloses Protokoll zur Übertragung von Daten auf der Anwendungsschicht über ein Rechnernetz. In den meisten Fällen wird dieses Protokoll dazu verwendet, um Webseiten auf einen Webbrowser zu laden.

### HTTP-Request/Response

Im Http wird zwischen Request (Anfrage) und Response (Antwort) unterschieden, wobei der Client eine Request an den Server sendet und dieser mit einer Response antwortet. Diese Nachrichten müssen dabei dem Http Standard mit Header und Body genügen.

## **8. Quellen**

<http://schabby.de/hashmap/>

<https://jena.apache.org/>

<http://www.json.org/>

[https://www.w3.org/2001/sw/wiki/Apache\\_Jena](https://www.w3.org/2001/sw/wiki/Apache_Jena)