

# Protokoll 1. Meilenstein Softwaretechnik-Praktikum SS2013

## Gruppe Semantic Chess

24.01.2013

gezeichnet: Franz Teichmann

anwesende Personen:

B	Dr. Axel Ngonga
	Dr. Jens Lehmann
T	Franz Teichmann
S	Lasse Kohlmeyer
	Felix Schmiedt
	Stefan Wetzig
	Marcel Kisilowski
	Sascha Hildebrandt
	Erik Körner
	Gerard Treptow

Gliederung:

- 20 Minuten Vortrag
- 20 Minuten Diskussion

Teil 1 - Vortrag:

- Begrüßung
- Abschluss erster Etappe (Recherche)
- Gliederung des Vortrages entspricht Lastenheft
- Zielbestimmung:
  - Informationen zu Schachpartien und Spielern
  - Wie spielen Top-Spieler?
  - Nachspielen von Partien
  - Spiel gegen K.I.
- Produkteinsatz:
  - Konvertierung von Daten: PGN -> Turtle
  - Abfrage der Daten
  - Analyse von Schachspielen und Ergebnissen, Verknüpfungen finden
  - Datenbank verlinken (DBpedia)
- Designübersicht:
  - Vorstellung Benutzergruppen
  - Gewinnstatistiken, Suche in Datenbank
  - Admin: - Veränderung der Datenbank
- Weiterspielen einer recherchierten Partie gegen die K.I.
- Abspeichern des gespielten Spieles als RDF
- Hinzufügen von Informationen (Handynummer)
- Löschen von hinzugefügten linkings?
- Funktionalität und Arbeitsaspekte + Aufwandsverteilung

- Konvertierung 25%, Linking 20%, Abfrage 15%, Schachengine 15%, Weboberfläche 40%

Muss-Ziel: - Spiel gegen eine Maschine

Kann-Ziel: - Spiel Maschine vs. verbesserte Maschine  
- Spieler vs. Spieler

Vorprojekt: - Konvertierung Abschließen

- Weboberfläche so weit es geht, evtl. 50% schaffen

Über 100% Aufwand: enthält Kann-Ziele

Teil 2 - Diskussion:

S: Gedanken zur Verteilung, Feedback

B: - Anmerkung: Weboberfläche: existierende Komponenten wiederverwenden, einbauen von chessboards- sparen von Entwicklungszeit (Schachfiguren)

- Achten auf Lizenzen!

- open source Variante bevorzugen - einfache Weiterverwendung

- vorgeschlagene Schachengines, Frage nach Recherche der engines

S: - Entscheidung für erste Engine

- geschrieben in cpp, höchstes Rating, kleine engine, verständlicher

B: - Hintergrund der Engine? Heuristiken?

S: - Anfangs-Explorer

B: - existierende Protokolle der Schach-Engines, Implementierung der Interfaces der Protokolle, Fokus auf einfache Engine

- relativ geringe Erwartungshaltung gegenüber der Engine, Kann-Teil

- Verbesserung der Engines ist Forschungsteil

- Durchgehen der Folien

- use: case diagramm

- Was sind Muss, was sind kann Teile?

S: - größtenteils alles Muss-Teile, nur Stichpunkte

B: - Begründung fehlender Teile notwendig, daher besser weniger use cases

- Bedeutung von Linking?

S: - Verknüpfungen semantischer Datensätze

B: - zu jedem Spieler eine Uri, Konvertierung der Daten, kann lokal geschehen

- Sollen die Entitäten integriert werden

S: - verlinken wäre einfacher, speicherplatzgünstiger

- B: - für anzeige auf der Oberfläche, Entnahme aus DBpedia?
- anderer Service muss stets online sein, Anfrage schwieriger
  - Vorschlag: parsen der Daten
  - fallen kaum ins Gewicht bezüglich Speicher
- schon existierende Details zur Ontologie?
- S: - eingelesen zur Syntax
- Varianten von Uris
  - Basis-Uri für den Anfang?
- B: - Basis Uri muss für Publikation: Uri der Weboberfläche / Data sein
- Slash- Variante ist günstiger
- S: - Chessplayer gibt es bereits als Entität
- B: - darauf kann verwiesen werden, vor allem günstig für Entnahme von Instanzen
- S: - Orte in PGN müssen keine Städte sein
- B: - Erkennung von Duplikaten, kann übernommen werden
- B: - Limes entweder Kommandozeile oder Weboberfläche, Kommandozeile stabiler, Automatisierung des Linkings über Oberfläche aufwändig
- Linking geringere Priorität als Suche, nicht zwingend über die Weboberfläche
  - Gefahr, dass zu viel Arbeit entsteht
  - Konvertierung soll laufen, hat höchste Priorität
- S: - Wie groß ist die Datenbank
- B: - 2 Mio. Spiele
- B: - Szenario 9,10,11 nicht ganz so hohe Priorität, eventuell schon vorhanden
- B: - Es existieren bereits SparQL- Editoren, eventuell anwendbar, auto-completion
- Nutzer sollte bei der Eingabe unterstützt werden
- B: - Welche Gedanken haben Sie sich zum Data-Update gemacht?
- Kann alles der Admin erledigen.
- S: - Frage nach Empfehlung eines triple-Stores
- B: - Virtuoso, da meiste Erfahrungen
- akademisch keine Lizenzprobleme
  - Wenn Probleme mit Virtuoso können Sie gern nachfragen
- S: - Verwendung von Java, welche Funktionalitäten werden unterstützt
- B: - RDF- Unterstützung, queries, Modelle, etc.
- schwieriger in anderen Sprachen, Serverteil soll in Java geschrieben werden
  - Client evtl. andere Sprache

- S: - Was beinhalten Advanced queries
- B: - User ist in der Lage Datensätze genauer zu spezifizieren, da er weiß wie die Datensätze aussehen  
- normale query
- S: - Aufwandseinteilung realistisch?
- B: - linking weniger Zeit, Engine mehr, Konvertierung evtl. mehr Zeit  
- Eventuell Spezialfälle, wenn Spieler gleichen Namen haben  
- es gibt Identifikatoren, nicht im pgn file, geburtsdatum als Vorschlag  
- evtl. widersprechen sich einträge, kommt nicht oft vor
- Ontologie soll so implementiert werden, dass auch andere sie benutzen können  
- Übertragbarkeit wichtiger  
- Programm soll einarbeitbar für andere Menschen sein
- S: - Recherche für Serverkomponenten: - Was würde sich anbieten für gesamtes System? Spring, Wicket
- B: - Java-Framework, Wadin, Spring als Alternativen, Entscheidung frei überlassen  
- Vor/ Nachteile abwägen, was die Einarbeitung eingeht

- Überarbeitung Use-Case, Prioritäten und Muss/kann  
- Überarbeitung Aufwandseinteilung, Qualitätsanforderungen  
wichtig: Gedanken zur Ontologie + Nachfragen

8-10 Punkte