

Semantic Web

Stephan Pavlovic

openTree

Konzept

Patrick Erasmus
David Bellingroth

29. November 2011

Inhaltsverzeichnis

Einführung	3
Hintergrund und Status Quo	3
Zielsetzung	3
Einsatzkontext	3
Ziele	4
Die Datenbank	4
Die Webanwendung	4
Daten bearbeiten	4
Karte	4
Jüngster gemeinsamer Vorfahre	4
Technische Grundlagen	5
Systemarchitektur	5
Datenmodell	6
Eingesetzte Technologien	7
Datenbank	7
Webanwendung	7

Einführung

Hintergrund und Status Quo

Genealogie oder „Ahnenforschung“ ist vor allem in Deutschland ein beliebtes Hobby. Es existieren viele Vereine, die sich diesem interessanten Forschungsgebiet gewidmet haben. Das Interesse an der Ahnenforschung erwacht meist in der eigenen Familie beschränkt sich aber längst nicht darauf.

Durch das Aufkommen des Internets bekamen die Ahnenforscher eine großartige Möglichkeit sich zu vernetzen und Informationen auszutauschen. Allerdings mangelt es immer noch an einer zentralen Datenbank für genealogische Informationen, wie sie in anderen Domänen bereits verfügbar sind. Ein Beispiel für solche Datenbanken ist das freie Kartographierungsprojekt OpenStreetMap. Jeder kann zum Datenbestand von OpenStreetMap beitragen, so dass die Karten des Projekts die von Google Maps und Konsorten bereits in vielen Gebieten an Genauigkeit und Detailreichtum übertreffen.

Zielsetzung

Ziel des Projekts openTree ist die Entwicklung einer Webbasierten Plattform und gleichzeitig einer offen verfügbaren Datenbank, die es ermöglicht Informationen über Familienverhältnisse und Stammbäume zu erfassen, abzufragen und zu visualisieren. Die Daten sollen außerdem mittels einer Semantic Web konformen Technologie abrufbar sein und sich so leicht in andere Anwendungen einbinden lassen.

Die openTree Webanwendung bietet die Möglichkeit, Daten in die Datenbank einzutragen, sowie Informationen aus der Datenbank in visuell aufbereiteter Form zu präsentieren. Denkbar wären beispielsweise eine Stammbaumsansicht oder eine Karte, welche die Verteilung von Nachnamen visualisiert.

Einsatzkontext

Die Plattform richtet sich primär, aber nicht ausschließlich, an Ahnenforscher und solche, die es werden wollen. Daneben sind natürlich eine Vielzahl anderer Verwendungszwecke vorstellbar.

Ziele

Die Datenbank

Erstes Ziel des Projektes ist die Einrichtung einer Semantic Web/Linked Data konformen Datenbank, bzw. RDF-Stores in den Nutzer allgemeine Daten über Personen und insbesondere deren verwandschaftliche Beziehungen untereinander eintragen können. Um die Nutzung mittels beliebiger Software zu erlauben, wird ein SPARQL-Endpoint zur Verfügung gestellt über den die Daten abgefragt und modifiziert bzw. erweitert werden können.

Die Datenbank setzt auf eine Ontologie, die so gestaltet ist, dass sich möglichst viele fehlende Informationen durch einen Reasoner mittels logischer Schlussfolgerungen ermitteln lassen. So wird beispielsweise die Information dass Peter der Onkel von Anna ist dadurch geschlussfolgert, dass Peter der Bruder von Hans ist und Hans der Vater von Anna.

Die Webanwendung

Zweites Ziel des Projektes ist die Entwicklung einer Webbasierten Anwendung, die dem Nutzer verschiedene Möglichkeiten des Zugriffs auf die Daten der Datenbank bietet. Einige werden hier genannt:

Daten bearbeiten

Der Nutzer kann mittels Webinterface neue Daten in die Datenbank eintragen sowie bestehende Daten bearbeiten. Dies stellt die Standardmethode zum Zugriff auf die Daten dar. Natürlich können Drittanbieter eigene Interface entwickeln, die diese Funktionalität bieten.

Karte

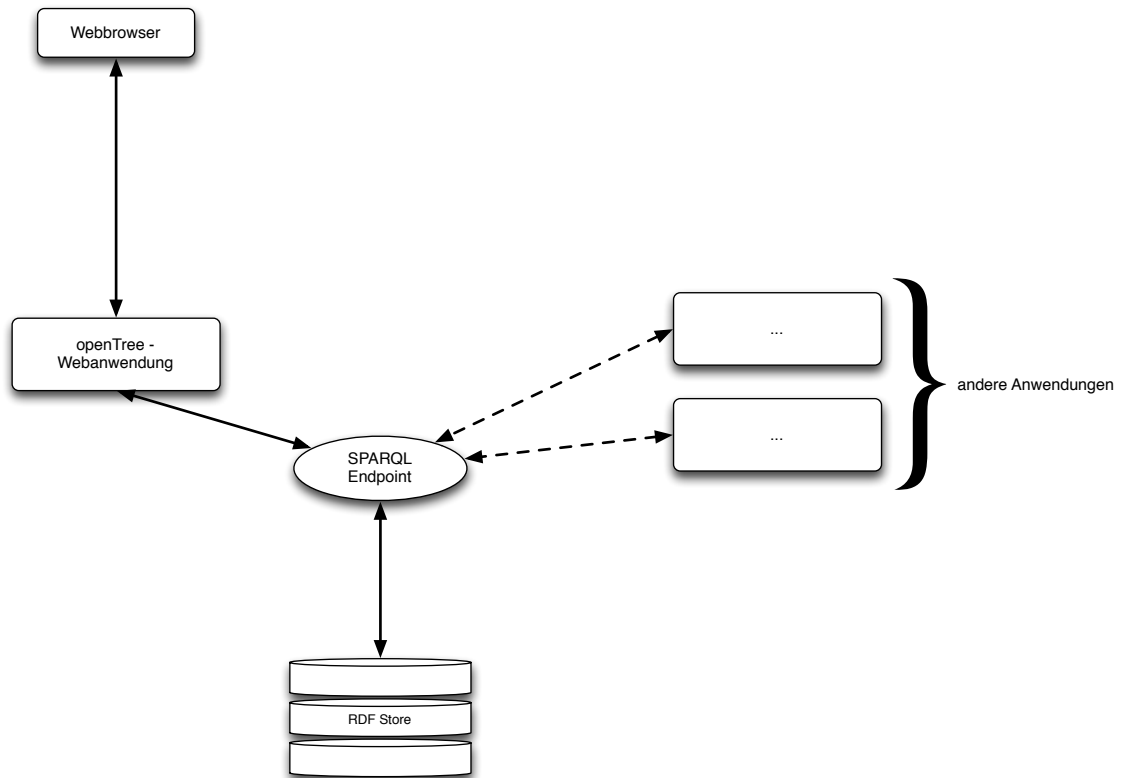
Der Nutzer hat die Möglichkeit, die geographische Verbreitung eines Vor-/Nachnamens auf einer Karte zu visualisieren. Möglicherweise ist zusätzlich eine Ergänzung um eine temporale Komponente möglich, z.B. in Form einer Zeitachse.

Jüngster gemeinsamer Vorfahre

Nutzer haben die Möglichkeit, zwei Personen auszuwählen und sich deren letzten gemeinsamen Vorfahren anzeigen zu lassen.

Technische Grundlagen

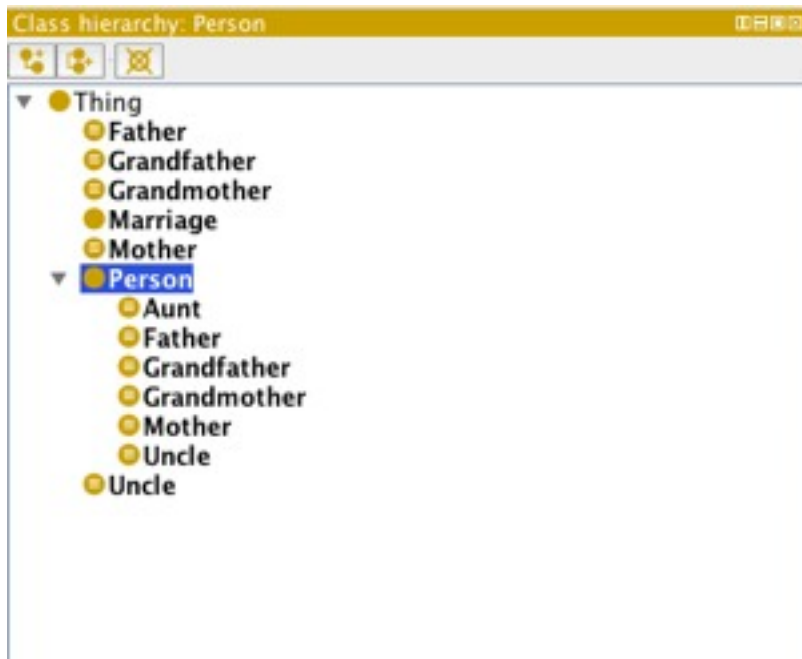
Systemarchitektur



Das System besteht aus einem RDF-Store, der mittels eines SPARQL-Endpoints angesprochen werden kann. Über den SPARQL Endpoint können Anwendungen die Daten des RDF-Store nutzen und erweitern.

Die Webanwendung greift ebenfalls über den öffentlich zugänglichen SPARQL Endpoint auf die Daten zu. Sie kommuniziert per HTTP(S) mit dem Webbrowser des Users.

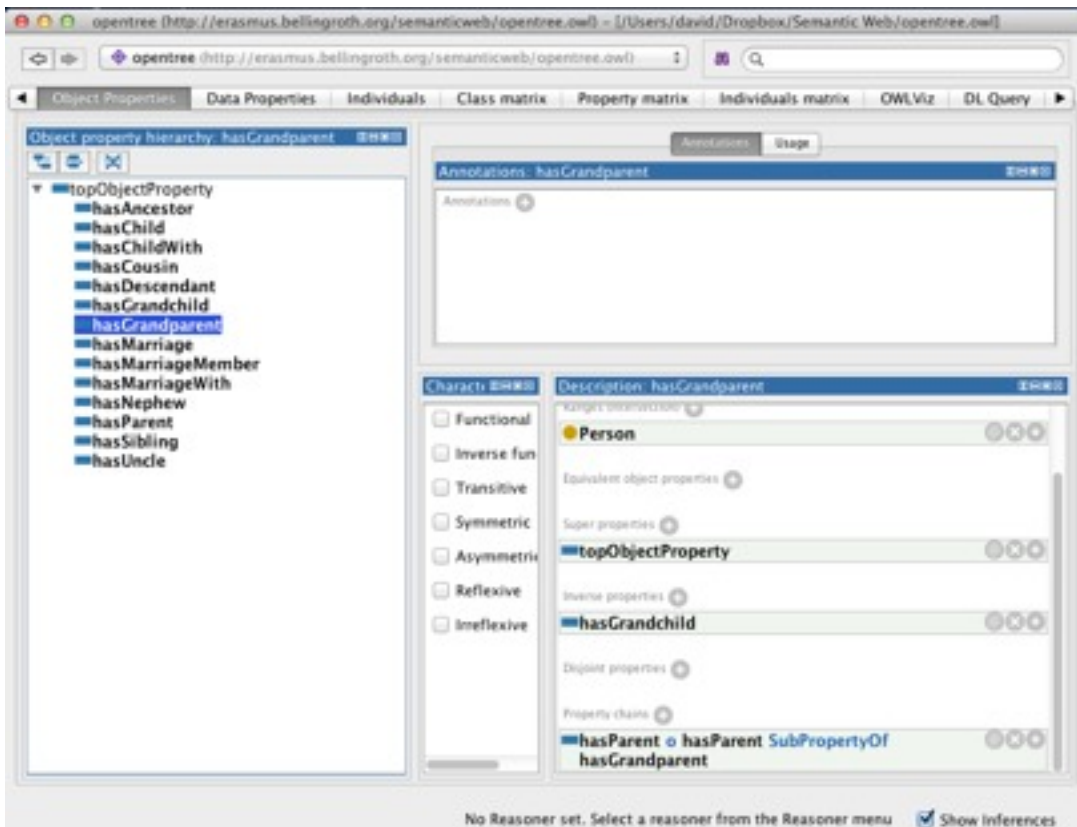
Datenmodell



vorläufige Klassenhierarchie

Das Datenmodell umfasst eine OWL Ontologie, welche die grundlegenden Verwandtschaftsbeziehungen und einige Basismerkmale von Personen beinhaltet. Diese werden in Form von Datenproperties der Person zugeordnet

Ausgangspunkt aller Beziehungen ist die Klasse Person. Sie enthält die wesentlichen Merkmale einer Person, wie Name, Geburtsdatum, Geschlecht etc.. Neben dieser Klasse bestehen noch andere Klassen, darunter eine Marriage-Klasse zur Abbildung von Ehen. Den wichtigsten Faktor der Ontologie stellen allerdings die Beziehungen, das heißt Objekteigenschaften, der Klasse Person dar.



Object Properties

Diese Objekteigenschaften bilden die verschiedenen verwandschaftlichen Beziehungen ab. Dabei wird in großem Maße auf Property Chains und Inverse Properties gesetzt, so dass ein Reasoner die Möglichkeit hat fehlende Beziehungen zu ergänzen, falls sie sich aus den bestehenden Beziehungen logisch schließen lassen.

Eingesetzte Technologien

Datenbank

Für die Speicherung sowie den Zugriff auf die Tripel wird wahrscheinlich der Sesame RDF Store verwendet. Er lässt sich ohne größeren Aufwand auf einem Java Application Server deployen und relativ einfach konfigurieren. Da es selbst keinen OWL kompatiblen Reasoner aufweist, ist es sinnvoll Sesame mit einem RDF-Repository zu kombinieren, das Reasoning-Fähigkeiten aufweist. In Frage kommt hier beispielsweise OWLIM.

Webanwendung

Zur prototypischen Realisierung der Webanwendung kommt Ruby on Rails, ein in der Programmiersprache Ruby geschriebenes Framework für Webapplikationen zum Einsatz. Rails bietet eine Vielzahl von Möglichkeiten und setzt auf das „coding by convention“-Prinzip. Dadurch werden dem Entwickler bei der Implementierung einer Webanwendung viele Schritte abgenommen, was einer schnellen Prototypisierung zuträglich ist.