



Dokumentation zum Projekt Mensch-Maschine-Systemtechnik

WEITERENTWICKLUNG EINER REVISIONSDARSTELLUNG FÜR DAS REVISIONSVERWALTUNGSSYSTEM R43PLES

Gruppe 6: Dominik Baumann, Johannes Börnicke, Katrin Messbauer,
Angela Wobar

INHALTSVERZEICHNIS

1	Motivation	5
2	Stand der Technik	7
2.1	Grundlegende Richtlinien	7
2.2	Bisherige Visualisierungen	7
2.3	Weitere Literatur	8
3	Anforderungsdefinition	9
3.1	Einleitung	9
3.1.1	Zielsetzung	9
3.1.2	Produktziele	10
3.2	Allgemeine Beschreibung	10
3.2.1	Produktumgebung	10
3.2.2	Benutzereigenschaften	10
3.2.3	Produktfunktion	10
3.2.4	Allgemeine Einschränkungen und Randbedingungen	10
4	Umsetzung	11

1 MOTIVATION

Um mit mehreren Personen koordiniert an großen Softwareprojekten arbeiten zu können, ist ein Versionsverwaltungssystem unerlässlich. Insbesondere für das Semantic Web mangelt es derzeit noch an einem solchen System, weshalb es in der Industrie noch nicht akzeptiert ist. Das Semantic Web stellt eine Erweiterung des Webs dar, mit der Informationen einfacher austausch- und verwertbar werden sollen, indem zu Begriffen zusätzliche semantische Informationen hinzugefügt werden. In [4] wird mit R43ples ein Konzept für ein solches Versionsverwaltungssystem vorgestellt. Zur Visualisierung werden hier gerichtete Graphen verwendet. In der vorliegenden Arbeit soll die vorhandene Visualisierung erweitert werden, um die Changesets, der einzelnen Revisionen sichtbar zu machen. Dabei werden zunächst bestehende Konventionen und Ästhetikkriterien für Graphen untersucht und die bisherigen Arbeiten an R43ples, sowie andere Konzepte für Versionsverwaltungssystemen für das Semantic Web bzw. Linked Data im Allgemeinen analysiert werden, um auf dieser Basis anschließend eine eigene Visualisierung zu entwerfen und zu implementieren.

2 STAND DER TECHNIK

In [1, 5] wurden bereits Visualisierungen für Das Revisionsverwaltungssystem R43ples entwickelt. Diese sollen im Folgenden analysiert werden, nachdem zunächst einige allgemeine Richtlinien für das Erstellen von Visualisierungen von Graphen dargelegt wurden. Anschließend werden noch weitere Visualisierungen von Linked Data aus der Literatur diskutiert.

2.1 GRUNDLEGENDE RICHTLINEN

Ein Graph ist definiert als eine Menge an Knoten und Kanten, wobei jede Kanten zwei Knoten aus dieser Menge verbindet [2, Seite 2]. Sie dienen zur Veranschaulichung von Relationen zwischen Objekten und können dabei u.a. in gerichtete und ungerichtete Graphen unterteilt werden. Bei ungerichteten Graphen wird nicht zwischen Start- und Zielknoten unterscheiden, bei gerichteten Graphen ist eine solche Unterscheidung vorhanden [3]. Für die Aufgabenstellung liegt also ein gerichteter Graph vor, da jeder Knoten einen Vorgänger und einen Nachfolger hat und diese klar unterscheidbar sind.

Für das Zeichnen von Graphen existieren einige Konventionen, die z.B. in [6] aufgeführt sind. Hier soll die Konvention, dass alle Kanten als gerade Linie gezeichnet werden sollen, eingehalten werden. Zudem soll auf eine planare Zeichnung geachtet werden, was bedeutet, dass sich Kanten höchstens an Knoten schneiden sollen. Neben den Konventionen gibt es einige Ästhetikkriterien, die ebenfalls bestmöglich umgesetzt werden sollen. Dazu zählt die Minimierung der Schnittpunkte von Kanten, der Fläche und der Gesamtkantenlänge.

2.2 BISHERIGE VISUALISIERUNGEN

In [5] ist eine Möglichkeit zur Visualisierung bereits dargestellt. In diesem Entwurf steht der Graph im Mittelpunkt. Die einzelnen Knoten sind mit der Revisionsnummer beschriftet und durch die Pfeile ist ersichtlich, wie der Verlauf der Branches ist. Gleiche Branches sind zudem durch gleiche Farben gekennzeichnet. Durch einen Haken können zusätzlich die Tags angezeigt werden. In einem mouseover Feld werden bei Bedarf weitere Informationen angezeigt. Die Commits sind von unten nach oben angeordnet. Hierbei ist aber die zeitliche Abfolge der Commits in unterschiedlichen Branches nicht ersichtlich.

Die zweite Visualisierung in [1] stellt den Graphen mehr an die Seite und stellt mit Kommentar zu jedem Commit sowie dem Autor zusätzliche Informationen dar. Die zeitliche Abfolge ist hier ersichtlich, zu dem ist eine Einteilung in Monate gegeben. Der Graph wird durch diese Darstellung allerdings relativ klein und der Text dominiert die Visualisierung.

2.3 WEITERE LITERATUR

In [7] wird mit W3C PROV ein neuer Standard eingeführt, um Versionskontrollsysteme wie Git einfacher benutzen und veröffentlichen zu können, vgl. Abb. 2.1. Dort sind viele Informationen des semantischen Netzes dargestellt, durch verschiedene Formen getrennt. Damit ist das Format weniger intuitiv und es braucht eine gewisse Einarbeitungszeit, bis damit umgegangen werden kann. Dieses Modell ist eher als Werkzeug für Entwickler gedacht und damit als Vorlage für R43ples nicht geeignet, da hier mehr auf den Anwender eingegangen werden soll, der die wesentlichen Informationen dargestellt haben möchte anstatt einer ausführlichen Beschreibung des semantischen Netzes.

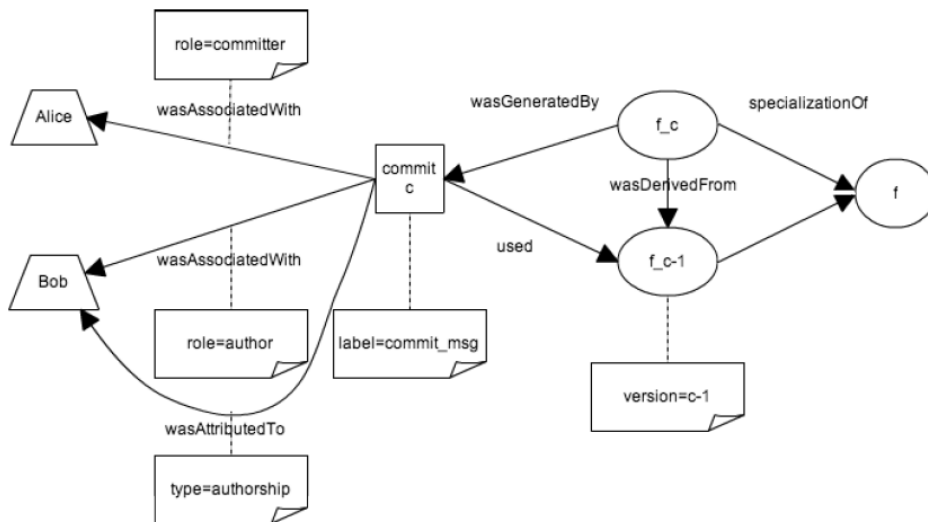


Abbildung 2.1: Darstellung eines Graphen mit Git2PROV

3 ANFORDERUNGSDEFINITION

Ideen:

- Mouseover für mehr Informationen
- Evtl. durch Klick dauerhaftes Hinzufügen der Informationen
- Show und Hide Haken
- evtl Änderung der Form auf Rechteck, um mehr Platz für zusätzliche Informationen zu erhalten
- Commits und Tags sollen weiterhin klar unterscheidbar sein, evtl. durch Farbe
- Vermeiden von Kreuzungen
- Ausgleich zwischen Information und Übersichtlichkeit finden
- Neuestes nach oben
- Evtl. Master in der Mitte, alles andere drum herum anordnen
- Evtl. Einfügen einer Suche
- Entscheidendes Ziel der zusätzlichen Informationen: Schnelles Finden der Version, die wiederhergestellt werden soll
- add und remove durch Symbol darstellen
- Mit Klick/Mouseover Informationen, was geaddet bzw. removed wurde
- Sortierung nach Datum? Evtl. auch grobe Zeiteinteilung

3.1 EINLEITUNG

3.1.1 Zielsetzung

In der vorliegenden Arbeit soll die im vorangegangenen Abschnitt analysierte, vorhandene Visualisierung für das semantische Revisionsverwaltungssystem R43ples erweitert werden, sodass die Inhalte der Changesets der Revisionen sichtbar werden. Der neue Entwurf soll sich dabei gut in die bisherige HTML-Oberfläche integrieren und gut skalieren in Bezug auf die Größe der Changesets.

3.1.2 Produktziele

Die Entwicklung soll als Fork auf GitHub durchgeführt werden, als Implementierungssprache ist Javascript vorgesehen. Es existiert eine HTML-Oberfläche, die weiter verwendet werden soll.

3.2 ALLGEMEINE BESCHREIBUNG

3.2.1 Produktumgebung

Revisionsverwaltungssysteme werden im Allgemeinen vorwiegend auf Desktop-PCs oder Laptops verwendet, evtl. auch auf Tablets. Deshalb ist es sinnvoll, von einem Bildschirm im Querformat auszugehen. Deshalb soll der Graph, statt wie in der bisherigen Visualisierung, anstatt von oben nach unten von links nach rechts angezeigt werden, um den Platz optimal auszunutzen und so auch lange Graphen ohne Scrollen darstellen zu können.

3.2.2 Benutzereigenschaften

Anders als die in Abschnitt 2.3 beschriebene Visualisierung, die eher für Entwickler gedacht ist, ist R43ples für Anwender gedacht, die keine ausführliche Darstellung des semantischen Netzes, sondern eine übersichtliche Visualisierung der wichtigen Informationen benötigen.

3.2.3 Produktfunktion

1. Übersichtlichkeit und Lesbarkeit der Graphen durch Einhalten der Richtlinien und Kriterien aus Abschnitt 2.1, sowie durch Beschränken auf die entscheidenden Informationen
2. Automatische Anordnung der Revisionen nach Zeitpunkt des Commits und der Verbindungen
3. Hervorheben des Master Branches, kennzeichnen der Branches mit Namen und Gruppierung durch einheitliche Farbgebung
4. Direkt dargestellt werden soll zudem die Revisionsnummer
5. Hinzufügen weiterer Informationen wie Changeset oder Kommentar durch Mouseover bzw. Klicken

3.2.4 Allgemeine Einschränkungen und Randbedingungen

4 UMSETZUNG

LITERATURVERZEICHNIS

- [1] Buntkiel, L., A. Lehmann, M. Zhang, S. Schönfeld und F.-J. Straube: Konzeption und Gestaltung einer Revisionsdarstellung für das semantische Revisionsverwaltungssystem R43ples. Dokumentation zum Projekt Mensch-Maschine-Systemtechnik, 2015.
- [2] Büsing, C.: Graphen- und Netzwerkoptimierung. Spektrum Akademischer Verlag, 2010.
- [3] Diekert, V., M. Kufleitner und G. Rosenberger: Elemente der diskreten Mathematik. De Gruyter, 2013.
- [4] Graube, M., S. Hensel und L. Urbas: R43ples: Revisions for Triples - An Approach for Version Control in the Semantic Web. Proceedings of the 1st Workshop on Linked Data Quality (LDQ), 1215, 2014.
- [5] Hoffmann, F., M. Müller, W. Nebyliza und J. Riffert: Konzeption und Gestaltung einer Revisionsdarstellung für das semantische Revisionsverwaltungssystem R43ples. Dokumentation zum Projekt Mensch-Maschine-Systemtechnik, 2015.
- [6] Jänicke, H.: Darstellung von Graphen. Vorlesungsunterlagen Visualisierung I. Uni Heidelberg.
- [7] Nies, T. D., S. Magliacane, R. Verborgh, S. Coppens, P. Groth, E. Mannens und R. V. de Walle: Git2PROV: Exposing Version Control System Content as W3C PROV. Poster and Demo Proceedings of the 12th International Semantic Web Conference, 2013.