Semantisches Wissensmanagement im Unternehmen: Konzepte, Technologien, Anwendungen

Prof. Dr. Stefan Linus Zander

Kapitel 3.2: Datenmodellierung mit Semantic MediaWiki

Vorbemerkung

Ein Großteil unseres menschlichen Daseins und Wirken definiert sich über Beziehungen zu anderen "Dingen" (hier im weitest möglichen Sinne zu verstehen). Möchte man diese Gegenstandsbereiche in technischen Systemen möglichst präzise und “naturgetreu” abbilden, so braucht es ausdrucksmächtige Beschreibungsformate und Datenstrukturmodelle. Viele bekannte Datenstrukturmodelle und Formate sind zwar sehr effizient in ihrer maschinellen Verarbeitung, ihnen fehlt es aber an der notwendigen Ausdrucksmächtigkeit und Erweierbarkeit, d.h., diese sind nicht in der Lage, die komplexen Beziehungen zwischen Dingen präzise und widerspruchsfrei (=unambiguitiv) abzubilden.

Beispiel
Versuchen Sie nachfolgend beschriebenen Sachverhalt in einem Datenmodell ihrer Wahl möglichst widerspruchsfrei und eindeutig abzubilden.

“In seiner konstituierenden Sitzung vom 25.03.2019 beschloss der Fachbereichsrat in Bezug auf Berichtspunkt Nr. 5 des Protokolls vom 17.02.2019 mit 10 ‘Ja’-Stimmen, 0 Enthaltungen und keiner Gegenstimme:
Prof. Dr. Kai Renz wird zum kommenden Sommersemester neues Mitglied des Stundenplanerteams.“

Überlegungen zum Datenmodell

Bei der Durchführung des vorherigen Beispiels werden Sie sehr schnell feststellen, dass die Repräsentation in Form eines konzeptuellen Graphen mit Knoten und Kanten die Beziehungen zwischen den “Dingen” in einer natürlichen Darstellung am nächsten kommt2.

Mit den Informationen aus Kapitel 2: Technologische Grundlagen können wir diesen Graphen weiter verfeinern.

- Ausschnitte aus Gegenstandsbereichen lassen sich am besten unter Reduzierung struktureller Heterogenität als konzeptueller Graph darstellen
- Beziehungen sollten als First-Class Elemente behandelt werden
- URIs/IRIs ermöglichen die eindeutige Identifizierung von “Dingen”
- Zusammen mit dem Konzept der Content Negotiation ermöglichen sie eine Unterscheidung zwischen technischen- und realweltlichen Repräsentationen

- Maschinen-verarbeitbare Semantik bedeutet: DEFINITIONS HAVE CONSEQUENCES – d.h., die Semantik eines Elements ist definiert durch die sich aus der Anwendung des mit dem Element assozierten Regelsatzes ergebenden Konsequenzen.

Wie werden Informationen in SMW kodiert ?

Semantic MediaWiki encodes information in the form of facts

Example: Berlin has a population of 3,520,031 people

A fact can be represented in a **triple-based lingusitic structure**:

1. The subject of a sentence, e.g., "Berlin"
2. The predicate, e.g., "has a population"
3. The object, e.g., "3,520,031 people"

These three parts can be split into a relation and a statement:

- **Relation** ~> Berlin has a population.
- **Statement** ~> The population is “3,520,031”.

This triple can be expressed with elements from SMW's KRF:

1. Subject is always a Wikipage
   - e.g. Page `Berlin` in the main namespace
2. Predicates are pages in the `Property` namespace
   - e.g., Page `Property:has Population`
3. Objects are represented as **literals** or other **wiki pages**
   - e.g., value “3,520,031” in case the object is a literal
   - e.g., Page named "Germany" in case object is another Wikipage

Encoding Information as Facts

Semantic MediaWiki encodes information in the form of facts

Example: Berlin has a population of 3,520,031 people

A fact can be represented in a **triple-based lingusitic structure**:

1. The subject of a sentence, e.g., "Berlin"
2. The predicate, e.g., "has a population"
3. The object, e.g., "3,520,031 people"

These three parts can be split into a relation and a statement:

- **Relation** ~> Berlin has a population.
- **Statement** ~> The population is “3,520,031”.

A statement consists of a property and its value:

- **Property** (i.e. predicate), e.g. “Has population”
- **Value** (i.e. object or literal), e.g. “3,520,031”

Adding a statement (property-value pair) to a page (the subject) is called annotation.

As a consequence, facts are encoded as **property-value pairs** on wiki pages.

Encoding Facts in Semantic MediaWiki

This triple can be expressed with elements from SMW's KRF:

1. Subject is always a **wiki page** or a **subobject** (we will learn about subobjects in Chapter 4)
   - e.g. a wiki page named `Berlin` in the main namespace
2. Predicates are **wiki pages** in the `Property` namespace
   - e.g., a wiki page named `has Population` in the `Property` namespace
3. Objects are represented as **literals** or other **wiki pages**
   - e.g., value “3,520,031” in case the object is a literal
   - e.g., a page named "Germany" in case the statement's object is another wiki page

Examples

(page – property – value):

```
Cologne    has population    1.017.155
Germany    has capital       Berlin
Spinoza    born on           24 Nov 1632
```

The triple-based model employed by Semantic MediaWiki is inherited from W3C's Resource Description Framework (RDF) specification.

A page-property-value triple resembles the subject-predicate-object triple pattern of RDF.

Part 2: How to Encode Information in Knowledge Graphs

Motivating Example

Matthias Frank is an employee of the FZI Research Center for Information Technology working on the BigGIS project. BigGIS is an ongoing research project started at April 2016 and deals with real-time big data and semantic technologies.

Steps:

1. Identify Instances (Resources)
2. Identify Concepts (Classes)
3. Identify Properties and Data (Numbers, Literals etc.)
4. Represent as Conceptual Graph with Stereotypes
5. Transform into Serialized Graphs using Mediawiki syntax

Step 1: Identify Instances

Matthias Frank is an employee of the FZI Research Center for Information Technology working on the BigGIS project.
BigGIS is an ongoing research project started at April 2016 and deals with real-time big data and semantic technologies.

Step 2: Identify Concepts

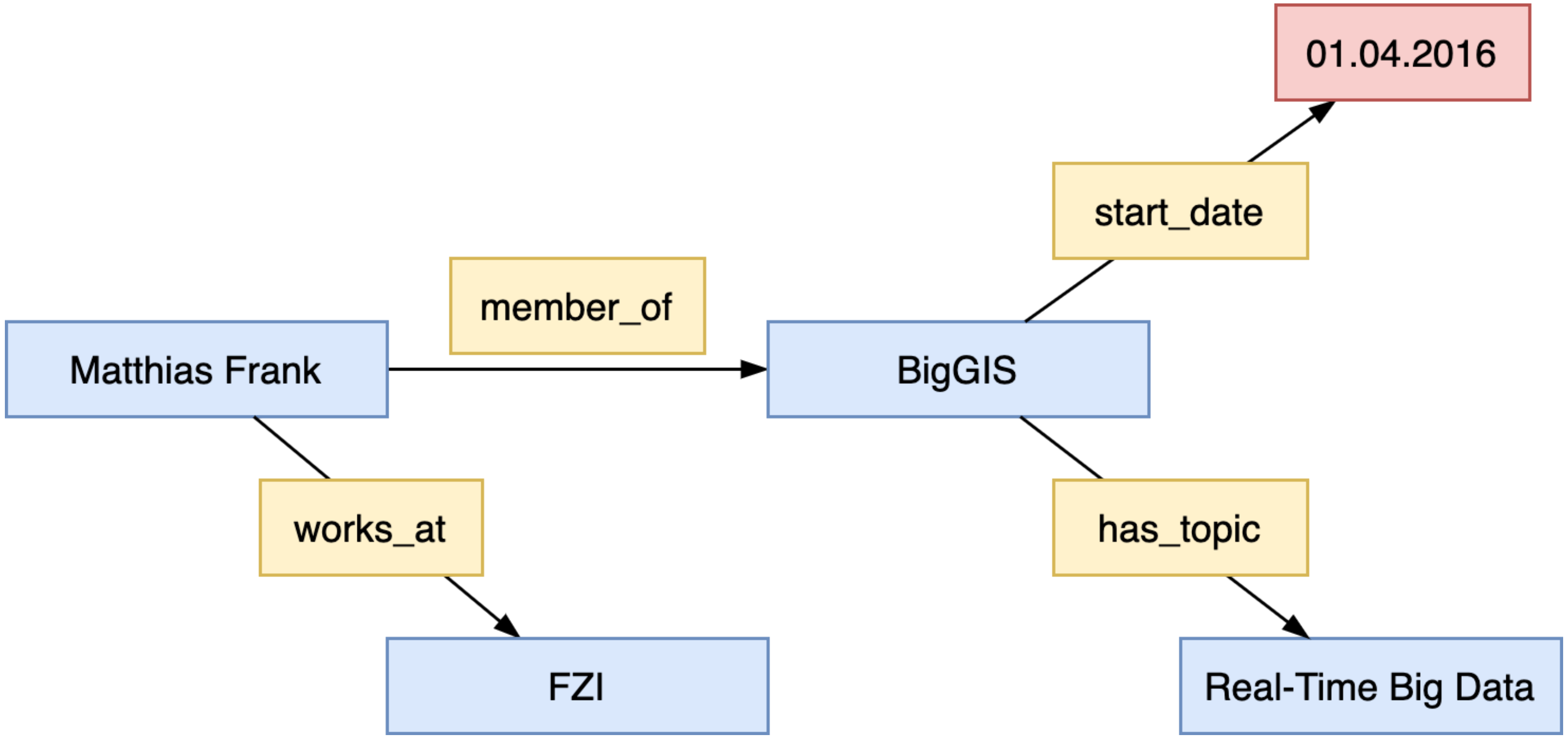
[[Matthias Frank]] is an employee of the [[FZI Research Center for Information Technology]] working on the [[BigGIS]] project.
[[BigGIS]] is an ongoing research project started at April 2016 and deals with real-time big data and semantic technologies.

Step 3: Identify Properties

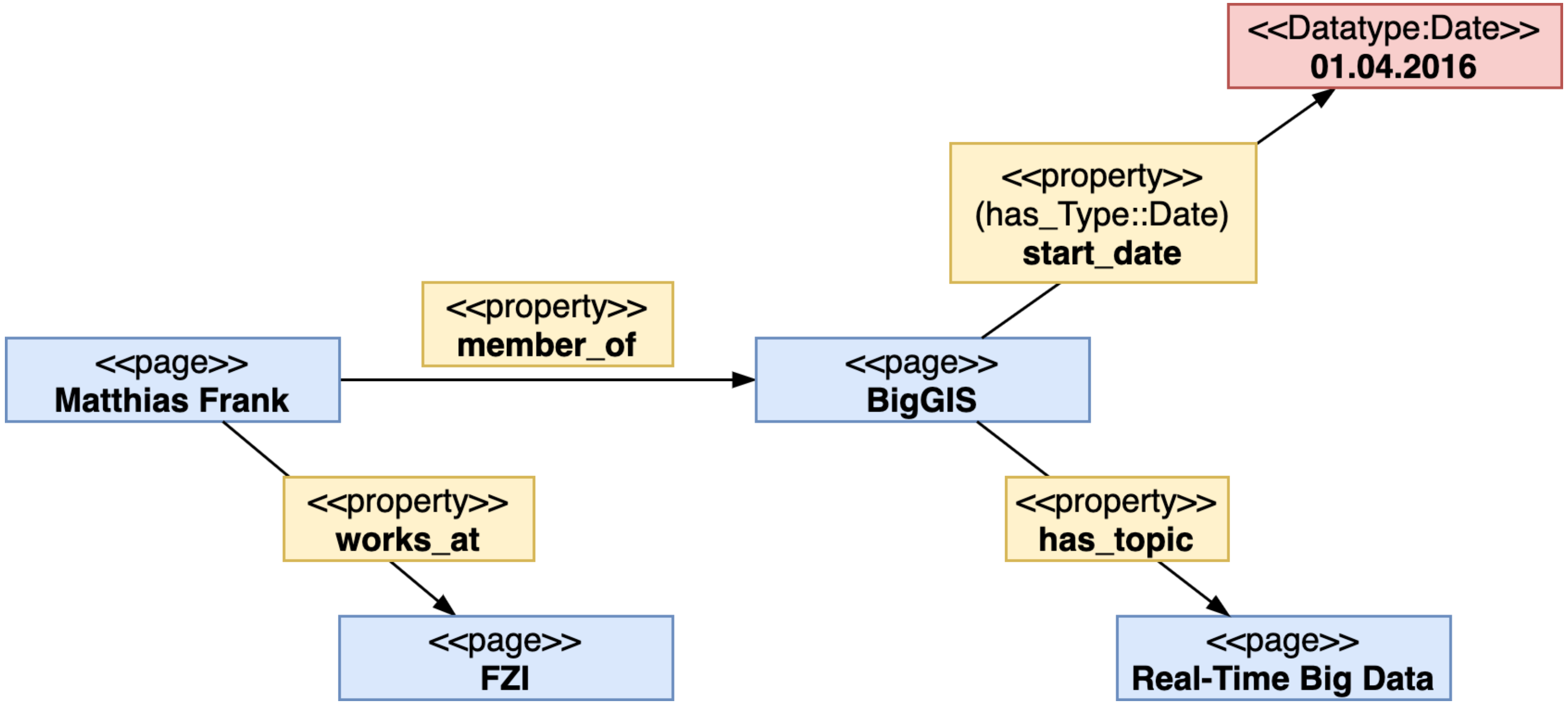
Matthias Frank is an employee of the FZI Research Center for Information Technology working on the BigGIS project.
BigGIS is an ongoing research project started at April 2016 and deals with real-time big data and semantic technologies.

Step 4a: Representation as Conceptual Graph

Matthias Frank is an employee of the FZI Research Center for Information Technology working on the BigGIS project. BigGIS is an ongoing research project started at April 2016 and deals with real-time big data and semantic technologies.



Step 4b: Conceptual Graph with Stereotypes



Step 5: Transformation into Serialized Graphs

It is important to separate the facts on their respective pages

Syntax on the `Matthias Frank`-Page:

`[[Matthias Frank]]` is an employee of the `[[employed_at::FZI Research Center for Information Technology]]` working on the `[[works_at::BigGIS]]` project.

`[[Category:Employee]]`

Syntax on the `BigGIS`-Page:

`[[BigGIS]]` is an ongoing research project started at `[[started::April 2016]]` and deals with `[[has_topic::Real-Time Big Data]]` and `[[has_topic::Semantic Technologies]]`.

`[[Category:Research Project]]`

Try for Yourself

Bayern Munich is a professional football club based in Munich, Germany, that plays in the Bundesliga.
Bayern Munich was founded in 1900. They play at the Allianz Arena. The club has won 30 national league titles.

Tasks

1. Identify all instances, properties, classes, literals etc.
2. Create the conceptual graph
3. Think about what classes, properties and literals are needed
4. Complement the above text using the Semantic MediaWiki syntax of the respective elements

TODO: Incorporate

- How can we state that Bayern Munich has won 30 Bundesliga Championship titles ?
- How can we express the years in which a Bundesliga title was won ?
- How can we express in which season + year a Bundesliga title was won ?
- How can we express in which seasons Bayern Munich finished second ?
- How can we count all national and international titles won by a German football club ?

All this calls for subobjects!!!

Why Knowledge Graphs are useful

Knowledge graphs have many advantages

- Like most graph models, it more intuitively captures the way we think about the world as humans (as networks, not as tables), making it easier to design, capture, and query data.
- As a data model supported by W3C standards, it allows us to create interoperable data and systems, all using the same standard to represent and encode data.

Sidenote:
There are many other methods that have been developed to handle this kind of complexity in RDF, including singleton properties and named graphs/quads. Additionally, an entirely different type of non-RDF graph model, labeled property graphs, allows users to attach properties directly to relationships.
However, labeled property graphs don't allow for interoperability at the same scale as RDF — it's much harder to share and combine different data sets, and moving data from tool to tool isn't as simple.

Source: https://enterprise-knowledge.com/rdf-what-is-it-and-why-do-i-need-it/