

Vortrag zum Forschungssemester

Prof. Dr. Stefan Linus Zander

Eine Übersicht der Forschungsaktivitäten und Tätigkeiten im WiSe 2020/2021

Tätigkeiten

- Markdown in der Lehre
- Wissensgraph-basiertes Tool zur Verwaltung von Abschlussarbeiten
- Umbau des WPF "Semantisches Wissensmanagement im Unternehmen"
- sonstiges

Tätigkeiten

- Markdown in der Lehre
- Wissensgraph-basiertes Tool zur Verwaltung von Abschlussarbeiten
- Umbau des WPF "Semantisches Wissensmanagement im Unternehmen"
- sonstiges

Teil 1: Markdown für die Lehre

Warum Markdown-basierte Präsentationsframeworks für Lehrinhalte ?

- Trennung von Inhalt von Darstellung und Layout
- Zielformat-unabhängige Beschreibung von Inhalten
- Gemeinsame Erstellen von Lehrunterlagen
- Versionierung
- Weniger "sperrig" als LaTeX Beamer
- Template-basierte Darstellung
- Viele Tools frei verfügbar und Open Source
- Nutzung von offenen Formaten und Standards
- Automatisierte Generierungsworkflows und Deployment



Bildquelle: <https://css-tricks.com/wp-content/uploads/2016/01/choose-markdown.jpg>

Methodology

Betrachtete Ansätze

- Marp – (<https://marp.app/>)  
- Slideas – (<https://www.slideas.app/>) 
- R Markdown – (<https://rmarkdown.rstudio.com/>) 
- Slidev – (<https://sli.dev/>) 

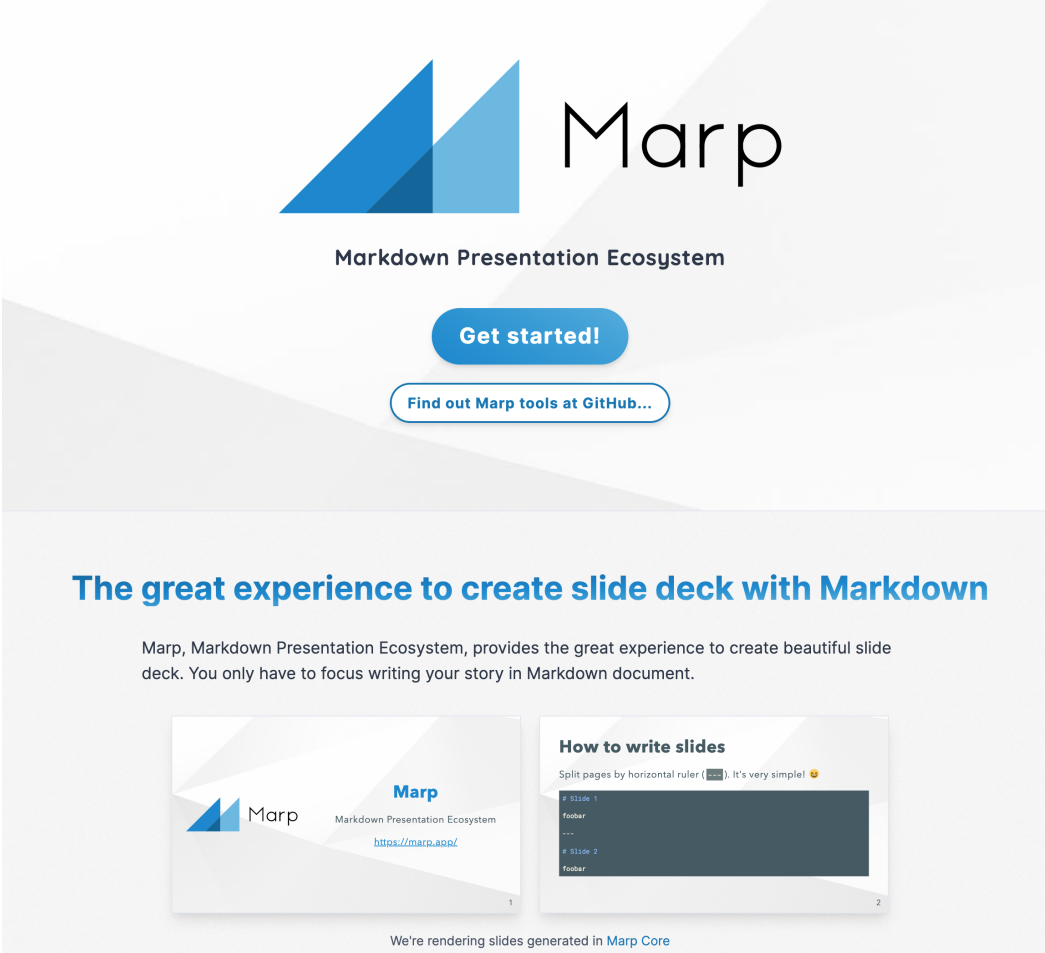
Forschungsfragen

- Ist eine Erhöhung der Produktivität feststellbar?
- Sind aktuelle Ansätze "ready" für einen Einsatz in der Lehre?

Proof of Concept

- Umbau der Unterlagen meines Moduls "Semantisches Wissensmanagement im Unternehmen"¹ (> 200 Slides)

¹ Online abrufbar unter: <https://projects.fbi.h-da.de/~s.zander/smw/html/>



The screenshot shows the Marp website. At the top, there is a logo consisting of two blue triangles forming a larger triangle, followed by the word "Marp" in a large, black, sans-serif font. Below the logo, the text "Markdown Presentation Ecosystem" is displayed. There are two buttons: a blue "Get started!" button and a white button with a blue border that says "Find out Marp tools at GitHub...". Below this, a section titled "The great experience to create slide deck with Markdown" is shown. It contains a paragraph: "Marp, Markdown Presentation Ecosystem, provides the great experience to create beautiful slide deck. You only have to focus writing your story in Markdown document." Below the text are two preview images of slides. The first slide is titled "Marp" and shows the Marp logo and the text "Markdown Presentation Ecosystem" with the URL "https://marp.app/". The second slide is titled "How to write slides" and shows a code snippet: "Split pages by horizontal ruler (---). It's very simple! 🙌", followed by "# Slide 1", "foobar", "...", "# Slide 2", and "foobar". At the bottom of the preview images, it says "We're rendering slides generated in Marp Core".

Wie sieht der Quellcode einer Markdown-Slide aus ?

Wissensarten

Grundsätzlich lassen sich **2 Wissensarten** unterscheiden¹:

Explicit Knowledge (=Disembodied Knowledge)

- ... kann kommuniziert werden
- ... kann auf verschiedenen Ebenen formalisiert werden
- ... kann in verschiedenen Formaten abgespeichert werden

Tacit / Implicit Knowledge (=Embodied Knowledge)

- ... im Gehirn gespeichert
- ... schwer zu kommunizieren
- ... nicht/kaum zu formalisieren

Explizites Wissen

...ist Faktenwissen, eindeutig kodifizierbar und mittels Sprache eindeutig kommunizierbar.

Implizites bzw. verborgenes Wissen

...ist Erfahrungs- und Handlungswissen, das an Personen gebunden ist;
...es ist nur schwer und nicht vollständig explizierbar (d.h., in Worte zu fassen)

¹ nach Michael Polanyi (1966), "The Tacit Dimension", University of Chicago, Press: Chicago

Quellcode

```

---
# Wissensarten

Grundsätzlich lassen sich ==2 Wissensarten== unterscheiden^1^:

:::: twocolumns
::: blue
**Explicit Knowledge** (= _Disembodied Knowledge_)
- ... kann kommuniziert werden
- ... kann auf verschiedenen Ebenen formalisiert werden
- ... kann in verschiedenen Formaten abgespeichert werden
:::

::: green
**Tacit / Implicit Knowledge** (= _Embodied Knowledge_)
- ... im Gehirn gespeichert
- ... schwer zu kommunizieren
- ... nicht/kaum zu formalisieren
:::
::::

_Explizites Wissen_
...ist Faktenwissen, eindeutig kodifizierbar und mittels Sprache eindeutig kommunizierbar.{.skip}

_Implizites bzw. verborgenes Wissen_
...ist Erfahrungs- und Handlungswissen, das an Personen gebunden ist;
...es ist nur schwer und nicht vollständig explizierbar (d.h., in Worte zu fassen)

::: footnotes
^1^ nach Michael Polanyi (1966), "The Tacit Dimension", University of Chicago, Chigago Press
:::

```

Weitere Beispiele

Was ist ein Wiki ?

Definition Ein **Wiki** ist ein **webbasiertes Hypertext-System** mit einer eigenen Markup-Sprache, das es Benutzenden ermöglicht, Webseiten direkt (d.h. ohne Programmierung) und online in einem Web-Browser zu erzeugen, zu lesen, zu verknüpfen und zu ändern.

Quelle: Angelehnt an Karin Haenelt, "Semantik im Wiki am Beispiel des MediaWiki und Semantic MediaWik", Fraunhofer, 2011.

Komponenten eines Wiki-Systems

Es gibt unterschiedliche Implementierungen von Wiki-Systemen;
Die am häufigsten verwendeten **Komponenten** sind

- Datenbank
- Versionsverwaltung
- Suchfunktion

MediaWiki

MediaWiki ist die bedeutenste und weitverbreiteste Implementierung eines Wiki-Systems

- Open-source Wiki-Software
- Technische Basisplattform der Wikipedia
- Genutzt von zehntausenden Webseiten und Organisationen¹

¹ Quelle: <https://www.mediawiki.org/wiki/MediaWiki/de>

Beispiel: Definition von eigenen Annotationen

```
::: definition
```

```
Ein ==Wiki== ist ein ==webbasiertes Hypertext-System== mit einer eigenen  
Markup-Sprache, das es Benutzenden ermöglicht, Webseiten direkt (d.h. ohne  
Programmierung) und online in einem Web-Browser zu erzeugen, zu lesen, zu  
verknüpfen und zu ändern.
```

```
Quelle: Angelehnt an Karin Haenelt, "Semantik im Wiki am Beispiel des MediaWiki und Semantic MediaWik", Fraunhofer, 2011.
```

```
:::
```

Lernzielkontrolle

Aufgaben

Beantworten Sie folgende Fragen:

- Wie lässt sich Wissen messen oder bewerten?

Überlegen Sie sich Ansätze und Möglichkeiten dies zu tun und diskutieren Sie diese.

- Lässt sich Wissen überhaupt managen? Falls ja, wie bzw. durch welche Maßnahmen?

Denken Sie kritisch über diese Frage nach; lässt sich Wissen wirklich managen und falls ja, in welcher Form und Ausprägung? Welche Ansätze (technisch / organisatorisch / personell / etc.) halten Sie hierfür für geeignet? Sammeln Sie eine Reihe von Ansatzpunkten und diskutieren Sie diese innerhalb ihrer Gruppe

- Warum ist Wissensmanagement mehr als eine technische Plattform?
- Überlegen Sie sich eine Reihe von organisationalen Maßnahmen zur Erhöhung der Informationsqualität
- Wie lässt sich Information Overload reduzieren?

MediaWiki: The Structure of Page Names

Page names consist of 3 different parts

Namespace:Title/Subpagetitle

Example: "User:Denny/Tests"

1. Namespace

- Prefixes, separated from title by colon :
- Not all prefixes that end in : are namespaces!
 - Available prefixes provided by MW, more can be added in configuration
 - Default: Main (empty), User, Category, Template, Help, MediaWiki, File, Special, Project (sitename)
- Purpose: distinguish basic "content types"
- Can have aliases (e.g. File: and Image:)

2. Pagetitle

- Defined during page creation
- MediaWiki determines whether page already exists
 - For existing pages, the page's content will be displayed
 - For non-existent pages, the edit view will be displayed

Page names are case-sensitive

MediaWiki distinguishes between upper- and lower-case letters in page names!

3. Subpagetitle

- Postfixes, separated from title by slash /
- Not all postfixes after / are subpages!
 - Enabled for certain namespaces
 - By default only for User and all Talk pages
- Often not appropriate for organising pages (rigid, hierarchical content structure)
- Small difference to pages with / in title
 - For example when moving pages
- Used in Wikipedia for multilingual page content

Silent Annotations using #set

The `#set` parser function allows to annotate data, i.e. assign values to a property, **silently** and avoid the `[[Property::value]]` syntax.

Example

```
{{#set:
  Has population=2,229,621
  |Located in country=France
}}
```

Equivalent Property Notation:

```
[[Has population::2,229,621| ]]
[[Located in country::France| ]]
```

The `set` parser function does **not display** anything, but saves the very same properties as data.

It is also possible to set **multiple values** to the same property:

```
{{#set:
  Has postcode=75001
  |Has postcode=75002
  ...
}}
```

```
{{#set:
  |Has text=fc00:123:8000::/%6;2001:db8::1428:57ab;2001:db8:0:8d3:0:8a2e:70:7344
  |+sep=;
}}
```

For full compatibility use the separator `|+sep=...` function to separate multiple values

Sources: (1) https://www.semantic-mediawiki.org/wiki/Help:Setting_values/Working_with_the_separator_parameter and (2) https://www.semantic-mediawiki.org/wiki/Help:Setting_values

Wildcards and Search Operators

Wildcards are written as `+` and allow *any value* for a given condition¹.

For example, `[[Born in::+]]` returns all pages that have any value for the property `Born in`.

Comparators are *special symbols* like `<` or `>`². They are placed after `::` in *property conditions*.

- `>>` and `<<`: "greater than" and "less than"
- `>` and `<`: "greater than or equal" and "less than or equal" by default
- `≥` and `≤`: "greater than or equal" and "less than or equal"
- `!`: "not" ("unequal")
- `~`: «like» comparison for texts and pages
- `!~`: «not like» comparison for texts and pages

When applying **comparators** to pages, then the *title of the page* (without namespace prefix) is used.

Comparators work only for **property values** and not for conditions on categories.

¹ Please note that `+` can only be used by itself¹.

² See https://www.semantic-mediawiki.org/wiki/Help:Search_operators

Annotation von Codefragmenten

```
package org.java.examples.singleton;

/**
 * Thread-safe implementation of a singleton
 * Does not require explicit synchronization
 * Uses the initialization-on-demand holder idiom that
 * makes uses of the class loader's own synchronisation mechanism
 * @author stefan zander
 */
public class ThreadSafeSingleton {

    private ThreadSafeSingleton(){ }

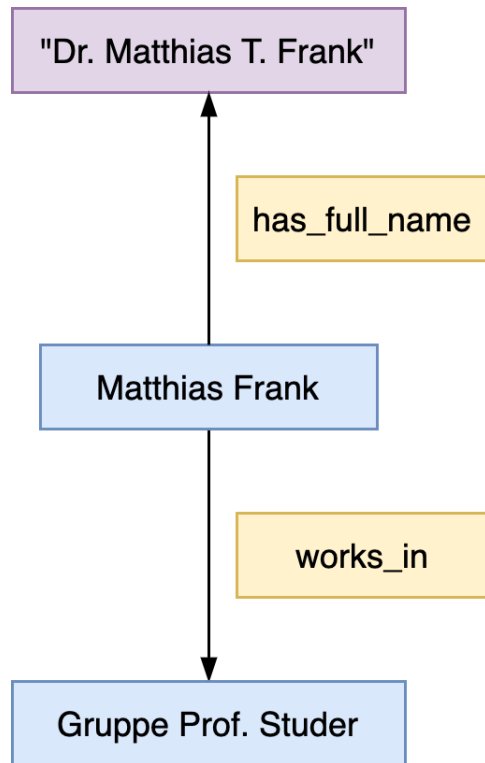
    private static class Holder {
        private static final ThreadSafeSingleton INSTANCE = new ThreadSafeSingleton();
    }

    public static ThreadSafeSingleton getInstance() {
        return Holder.INSTANCE;
    }

    // Aufruf wie gehabt mit ThreadSafeSingleton instance = ThreadSafeSingleton.getInstance()
}
```

Formulating Queries I

First consider the **structural composition** of the data model used to represent the universe of discourse (or an excerpt of it).



This **graphical representation** serves as basis for the formulation of **query conditions**.

- Therefore, think of query conditions as
~> **graphs with conditional node values**.
- Also consider the **structural semantics** of involved elements.

Structural Semantics of the knowledge graph

- `Matthias Frank` is a wiki page in the `main` namespace
- The page holds two assertions
 - a `has_full_name`-property the value of which is a Literal (datatype: `Text`)
 - a `works_in`-property the value of which is the wiki page representing Prof. Studer's research group

Resumé

Resumé

👍 Wann ist der Einsatz von Markdown-basierten Präsentationsframeworks sinnvoll ?

- Folien mit vielen Codebeispielen
- Mischung von Text und mathematischen Formeln
- Standardisierte Darstellung durch Stilregeln
- Nutzung der eigenen Toolchain
- Inhalte mit erhöhtem Änderungsaufwand
- Gemeinsames Erstellen von Lehr- und Lerninhalten
- Integration von unterschiedlichen Medientypen
 - Youtube, Codepen, etc.

👎 Welche Nachteile gibt es ?

- Frameworks teiw. noch unausgereift
 - viele "Hacks"
 - Dinge funktionieren nicht wie erwartet
- Kein bzw. nur rudimentäres "progressive disclosure"
- Weniger Designflexibilität
- teilw. ungewünschte Seiteneffekte
- Verlust der Übersichtlichkeit bei größeren Foliensätzen (mMn > 40 Folien)
- Präsentations-Modus abhängig vom Framework und
 - u.U. reduzierter Toolumfang

Beantwortung der Forschungsfragen

Forschungsfrage 1: Ist eine Erhöhung der Produktivität feststellbar ? --> teilweise

- Hohes Ausmaß an "Ramp-up"-Arbeit notwendig (Erstellung von Stilregeln / Austesten von Erweiterungen / etc.)
- Man geht doch gelegentlich dazu über, die Darstellungsweise zu optimieren
- In jedem Fall aber zusätzliche Motivation durch Freude bei der Erstellung

Forschungsfrage 2: Sind aktuelle Ansätze "ready" für einen Einsatz in der Lehre ? --> unter best. Voraussetzungen und Exp.-Freudigkeit ja

- Für techn.-mathematische Inhalte in jedem Fall
- Schnelles Deployment auf untersch. Zielmedien und bessere Änderungsnachvollziehbarkeit für Lehrende und Studierende
- Abhängig vom fachlichen / technischen Hintergrund der jew. Person

Teil 2: (Live-Demo)

Wissensgraph-basiertes Tool zur Verwaltung von Praxisphasen und Abschlussarbeiten

Last but least...





Vielen Dank!

Playground: Vorteile (persönliche Meinung)

- Konzentration auf Inhalte nicht auf deren Darstellung ✓
- Schnelle Generierung von neuen Lehrmaterialien ★★ ★
- Einheitliche Darstellung durch Verwendung von Tempaltes und definierten Stilregeln ✓
- Einfache Syntax ohne **umfangreiches** Tagging (vgl. LaTeX-Beamer)
- **Attraktive** Darstellung von `Programcode`
- Einfache Einbindung von **Web-basierten** Inhalten und Formaten

- Konzentration auf Inhalte nicht auf deren Darstellung
`{color:green}:fa-check:{color}`
- Schnelle Generierung von neuen Lehrmaterialien
``
`:fas-star: :fas-star: :fas-star:`
``
- Einheitliche Darstellung durch Verwendung von Tempaltes und definierten Stilregeln `*:fa-check:*{.red}`
- Einfache Syntax ohne `{color:red}umfangreiches{color}` Tagging (vgl. LaTeX-Beamer)
- `[Attraktive]{.red}` Darstellung von ``Programcode``
- Einfache Einbindung von `*Web-basierten*{.red}` Inhalten und Formaten