5.3 Monitored Success Factors

This document summarises the success factors monitored by the three prototypes of the CODE platform:

- Mendeley Desktop/Client and Server API for semantically enriching research publications
- MindMeister.com Collaborative Mapping Platform for generating semantic web enabled mind-maps and mind-map-based presentations
- 42-data.org - a data-centric question and answer portal for people believing in data

CODE features have been successfully deployed in the three platforms, and users are engaging with them. CODE had set very ambitious success factors where some of them could be met, while others have not been achieved yet. The two SME partners Mendeley and MindMeister are committed to continue developing and promoting the CODE features as users have found them valuable.

<table>
<thead>
<tr>
<th>Project Acronym</th>
<th>CODE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Grant Agreement number</td>
<td>296150</td>
</tr>
<tr>
<td>Project Title</td>
<td>Commercially Empowered Linked Open Data Ecosystems in Research</td>
</tr>
<tr>
<td>Date</td>
<td>2014-04-30</td>
</tr>
<tr>
<td>Nature</td>
<td>R (Report)</td>
</tr>
<tr>
<td>Dissemination level</td>
<td>RE (Restricted)</td>
</tr>
<tr>
<td>WP Lead Partner</td>
<td>Mendeley</td>
</tr>
<tr>
<td>Revision</td>
<td>Final</td>
</tr>
<tr>
<td>Authors</td>
<td>Maya Hristakeva, Michael Granitzer, Roman Kern, Michael Hollauf</td>
</tr>
</tbody>
</table>

Statement of originality: This document contains original unpublished work except where clearly indicated otherwise. Acknowledgement of previously published material and of the work of others has been made through appropriate citation, quotation or both. This document reflects only the author’s views and the European Community is not liable for any use that might be made of the information contained herein. © CODE Consortium, 2012
Project Officer & Project Coordinators

Project Officer | Stefano Bertolo | European Commission  
DG CONNECT Unit G3  
EUROFORUM 01/293 - 10 rue Robert Stumper, L-2557 Luxembourg  
stefano.bertolo@ec.europa.eu

Project Coordinator | Stefanie Lindstaedt | Know-Center GmbH  
Inffeldgasse 13, 8010 Graz, Austria  
+43 316 873-30800 (phone)  
+43 316 873-1030800 (fax)  
slind@know-center.at

Scientific Coordinator | Michael Granitzer | University of Passau  
Innstrasse 33a, D-94032 Passau  
+49(0)851-509-3305  
michael.granitzer@uni-passau.de

Document Revision History

<table>
<thead>
<tr>
<th>Revision</th>
<th>Date</th>
<th>Author</th>
<th>Organization</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>2nd draft</td>
<td>16.04.2014</td>
<td>Michael Granitzer</td>
<td>University of Passau</td>
<td>University of Passau, Description</td>
</tr>
<tr>
<td>3rd draft</td>
<td>28.04.2014</td>
<td>Michael Hollauf</td>
<td>MeisterLabs</td>
<td>MindMeister Description</td>
</tr>
<tr>
<td>4th draft</td>
<td>28.04.2014</td>
<td>Roman Kern</td>
<td>KnowCenter</td>
<td>Code Annotator Tool</td>
</tr>
<tr>
<td>5th draft</td>
<td>29.04.2014</td>
<td>Maya Hristakeva</td>
<td>Mendeley</td>
<td>Update Success Factors, Add Sprint Planning</td>
</tr>
<tr>
<td>6th draft</td>
<td>30.04.2014</td>
<td>Maya Hristakeva</td>
<td>Mendeley</td>
<td>Updated Introduction, Success Factors, Partner Descriptions, Summary</td>
</tr>
</tbody>
</table>

Final version 30.04.2014
# Table of Contents

1. Introduction - 3 -

2. Success Factors - 4 -

3. Mendeley Desktop and Platform API - Enhanced Data Extraction from PDFs - 7 -
   3.1 Monitoring Mechanisms - 7 -
   3.2 Performance of CODE Features - 9 -
   3.2.1 Mendeley Desktop - 9 -
   3.2.2 Mendeley API - 10 -
   3.3 Enrichment Service - 11 -

4. MindMeister: Semantic Mind Maps - 14 -
   4.1 Monitoring Mechanisms - 14 -
   4.2 Performance of CODE Features - 14 -
   4.2.1 Wunderkind - 14 -
   4.2.2 PDF Import - 14 -
   4.2.3 RDF Export - 15 -

5. 42-data: A data centric question and answering portal - 16 -
   5.1 Monitoring Mechanisms - 16 -
   5.2 Performance of CODE Features - 16 -

6. Sprint Planning - 19 -

7. References - 20 -

8. Appendix - 21 -
1 Introduction

This deliverable reports on the success factors monitored by the three prototypes of the CODE platform:

- Mendeley Desktop/Client and Server API for semantically enriching research publications
- MindMeister.com Collaborative Mapping Platform for generating semantic web enabled mind-maps and mind-map-based presentations
- 42-data.org - a data-centric question and answer portal for people believing in data

All three platforms are characterized by their production and consumption of Linked Open Data. They target to evaluate the value of linked open data within research and data intensive processes.

CODE features have been successfully deployed in the three platforms, and users are engaging with them. Some of the highlights on the success factors are as follows:

- Developed scalable Enrichment Service which can be adapted to different workloads (low, medium, high), and at peak times could enrich almost 900,000 PDFs using 512 workers (64 machines each with 8 cores).
- Made available 87 LOD endpoints via the Balloon Service, 46 of which are used in the 42-data Platform and the Query and Visualisation Wizard.
- Generated a large number of Aggregated Data Sets consisting of Data Cubes, enriched PDFs and mind maps as SKOS thesauri.
- Achieved good uptake of CODE features deployed in Mendeley receiving 690,000 API calls by over 8,000 external users in less than two months after launch.
- Made 120,000 public mind maps available in the Linked Open Data cloud
- Developed a new portal, 42-data, for data centric discussions and socializing around data, which has 41 registered users and 205 created resources.
- Provided access to over 10,000 data cubes in the Linked Open Data Cloud and released 4,000 data cubes from Open Access articles via the 42-data SPARQL Endpoint.
- Delivered three new cutting-edge features to MindMeister end users, resulting in very positive user feedback and increased publicity of the server.

We focused on deploying CODE enabled services that are stable and have good quality features. As a result, public launches of some CODE features took longer than expected. Due to some of these delayed releases, we were unable to achieve the target number of “Validated Annotations” and “Number of Recommendations Accepted”.

CODE services for consuming and producing Linked Open Data will be sustained and extended by the two SME Partners Mendeley and Meisterlabs after the lifetime of the project. CODE is a starting point to having research papers on the factual level in the future.
## 2 Success Factors

In this section we report on the overall success factors achieved by the three prototypes: Mendeley Desktop/Client and Server API, MindMeister Semantic Mind Maps, and 42-Data Question Answering Portal. For each success factor we give a quantitative estimate and a qualitative description how it has been calculated.

<table>
<thead>
<tr>
<th>Success Factor</th>
<th>Goal Phase 3</th>
<th>Status</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algorithmic Annotations and Integrations per Day</td>
<td>500,000</td>
<td></td>
<td>Annotations include tables, figures, table of content and entities. The numbers are monitored by the enrichment service.</td>
</tr>
<tr>
<td>Accessible LOD Repositories</td>
<td>20</td>
<td>87</td>
<td>The number of repositories includes only repositories of good service quality. Of the previously reported 229 services, we removed endpoints with low reliability due to slow response time and unreliable uptimes. Currently 46 endpoints are offered via 42-Data and Query Wizard Services, which provide more reliable performance and support fuzzy search. The numbers are monitored by the Balloon Service.</td>
</tr>
<tr>
<td>Validated Annotations</td>
<td>1,000,000</td>
<td></td>
<td>The number of validated annotation is composed as follows:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- Manual PDF annotations have been created in Phase II as test data for computer science extractors</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- Data Cubes consist of annotations and integrations to the LOD cloud. We validated 60,000 annotations in approximately 30 cubes.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- We converted public mind-maps to RDF using the SKOS Format. Those mind-maps are available as SPARQL endpoints and since they are manually created they also form validated annotations. However, only 200 of them have been exported by users and reused.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- We manually controlled 1,000 PDFs and rejected wrong annotations of tables, figures and table of content.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- External users have viewed 9,700 of the annotated PDFs and have not reject the annotations.</td>
</tr>
<tr>
<td>Aggregated Data Sets</td>
<td>10,000</td>
<td></td>
<td>The number of aggregated data sets is constituted as follows:</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- The 4,000 data cubes have been created as part of the development process of the 42-data portal. The data cubes have average quality according to manual inspection. Some of the data cubes are noisy and need cleaning.</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- 42-data resources are the atomic units in answers at 42-data</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>- Given the current enrichment and disambiguation service, we enriched all open-access publications available in Mendeley. The resulting 1.2mil</td>
</tr>
</tbody>
</table>

30/04/2014
research papers have been annotated with tables, figures and table of content information. In addition, 10,000 PDFs were annotated in JSON-LD format, and are available via Mendeley’s API.
- We converted public MindMeister mind-maps to RDF using SKOS. Those mind-maps are available as SPARQL endpoints and since they are manually created they also form validated annotations.

### Query API Calls from external users

<table>
<thead>
<tr>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>250,000</td>
<td>- 24,000 SPARQL MindMeister calls</td>
</tr>
<tr>
<td></td>
<td>- 10,530 SPARQL Data Cubes calls</td>
</tr>
<tr>
<td></td>
<td>- 200 Mind Map exports</td>
</tr>
<tr>
<td></td>
<td>- 690,000 Mendeley API calls</td>
</tr>
</tbody>
</table>

We count the following numbers for the API Calls:
1. SPARQL calls to the 42-data endpoints (including Data Cubes and MindMeister Thesaurus)
2. Number of requested RDF Maps at MindMeister
3. Number of API calls accessing PDF Annotations at Mendeley

### External User/Domain Experts

<table>
<thead>
<tr>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>5,000</td>
<td>- 8,225 Mendeley users</td>
</tr>
<tr>
<td></td>
<td>- 72 MindMeister users</td>
</tr>
<tr>
<td></td>
<td>- 42 42-data users</td>
</tr>
</tbody>
</table>

The following user counts are measured
1. Number of Mendeley Desktop users using enriched PDFs
2. Number of MindMeister users using enriched mind-maps
3. Number of 42-data users

### Number of Recommendations Accepted

<table>
<thead>
<tr>
<th>Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>100,000</td>
<td>None</td>
</tr>
</tbody>
</table>

The number has not been tracked yet due to delayed launch date of MindMeister services (see Section 4 for more details). The number would have been measured as
1. Number of mind map extensions through structured data (Wunderkind)
2. Number of disambiguation requests for table disambiguation
3. Number of requests to the Balloon service

### Income to cost ratio

<table>
<thead>
<tr>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sustainable</td>
</tr>
</tbody>
</table>

- Sustainable for Mendeley
- Sustainable for MindMeister
- Negative for 42-data

Although, CODE Services are not currently generating income for Mendeley and MindMeister, they can sustain the deployed features with their current infrastructure, thus not incurring any significant additional costs. Given the value CODE features have for the users, Mendeley and MindMeister plan on not only maintaining but also enhancing them further after the lifetime of the project. In contrast, 42-Data would need more funding to continue developing the platform. For more details see D 7.2.

### Data Quality

<table>
<thead>
<tr>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Good</td>
</tr>
</tbody>
</table>

- Good for enriched papers
- Average for Wunderkind
- Average for Data Cubes

We conducted the following data quality estimates
- We manually evaluated the annotations of 1000 tables, figures, table of contents and entities through the Mendeley QA team. The result show, that tables, figures and table of contents are of high enough quality to be exposed to the public Mendeley users. However, despite acceptable quantitative evaluations, entities lack the necessary accuracy and completeness to be exposed to the average user.
- We conducted quantitative evaluations of enrichment and integration algorithms and
CODE has set very ambitious goals where some of them could be met, while others have not been achieved. The ecosystems developed aimed for two routes as illustrated in Figure 1:

- First, we integrated CODE services into the partner platforms in order to ensure uptake. This required focusing mainly on stability and quality of features. As a result, we achieved good uptake especially for mining research papers. Usage numbers are still growing and will continue to do so in the near future. Hence, CODE services for consuming and producing Linked Open Data will be sustained and extended by the two SME Partners Mendeley and Meisterlabs after the lifetime of the project.

- Second, we focused on using CODE services in a new, experimental platform that combines Web 2.0 features with (Linked Open) data. This resulted in a new platform called 42-data. For such a new platform we have not been able to achieve high uptake yet, but are confident to do so in the future. The success factors show that uptake increases, but only at a slow pace. The community is interested in such a platform, but yet the incentives for investing in labour intensive analysis work are still missing.

**Figure 1: Positioning of the ecosystems developed in CODE w.r.t. up-take and stability**
3 Mendeley Desktop and Platform API - Enhanced Data Extraction from PDFs

Mendeley’s mission is to open up research in order to help researchers better organise their research, collaborate with one another and discover new research. Mendeley produces tools that help researchers with these tasks. Many of these tools consume and produce data that is stored on Mendeley’s platform. As Mendeley’s community grows and more people make use of the tools offered, the richer the data becomes. It also powers new tools, such as recommender systems, that require such rich data collections. Mendeley is opening up much of this data through their API so that third parties can also build tools that help researchers using the data that has been collected and perhaps also feeding back more data.

Mendeley has integrated a number of the CODE project services into its main product, Mendeley Desktop, and its API. Mendeley Desktop has so far been downloaded by over 2.5 million people throughout the world and is regularly used by many of them. The services integrated into Mendeley Desktop are provided for free to Mendeley users and are exposed through Mendeley’s API, supporting the platform model. Where services developed through the CODE project are available, a link is included to a user-friendly description of them and accreditation is given to the CODE project, its participants and funders.

The enhanced data extraction from PDFs tools reached a reasonable level of quality where Mendeley’s internal Quality Assurance team was happy for them to be integrated and released in a main Mendeley Desktop build. This includes extracting the table of contents, tables and figures from PDFs and presenting them to researchers as both summarisation and navigation supports. The tools were run over all Open Access articles in Mendeley’s catalogue resulting in over one million enrichments. All of these enriched documents can be freely downloaded and viewed using Mendeley Desktop and Mendeley’s API. By providing this enhanced data through the API, third parties are no longer restricted to using just the metadata of articles in their applications but can also use their tables of contents, tables and figures. Feedback mechanisms were also implemented in Mendeley Desktop that help users to see which articles have enriched data and to report errors in them. These errors can then be used to help focus on the areas of the extraction tools that require improvement. This version of Mendeley Desktop with enhanced data extraction from PDFs was made available to all Mendeley users on March 11th, 2014.

This chapter includes a description of the CODE services that have been integrated into Mendeley Desktop and Mendeley’s API, and presents statistics on usage of the following CODE features: Tables of Contents, Tables and Figures. We also discuss the mechanisms that were used to monitor the success factors.

3.1 Monitoring Mechanisms

We monitor CODE feature success factors via Mendeley’s event logging framework. We have been collecting usage data of the CODE features integrated with Mendeley Desktop since its public release on March 11th 2014. As of April 28th 2014, we have had over 8,000 unique users using the CODE features with on average 200 daily users. Logging of API calls for accessing CODE enrichments of documents started on April 3rd 2014. Since then, the service has executed 690,000 successful requests (30,000 on average daily).
In addition to usage log monitoring, we manually evaluated the annotations of 1000 tables, figures, table of contents and entities through the Mendeley QA team. The result showed that tables, figures and table of contents are of high enough quality to be exposed to the public Mendeley users. However, despite acceptable quantitative evaluations, entities lacked the necessary accuracy and completeness to be exposed to the average user.

We generate weekly event-related reports for CODE feature usage, which contain:

- Absolute number of times that an event is triggered
- The unique number of papers upon which the event is triggered
- The unique number of users who triggered the events

The actions included in the reports are as follows:

- API calls for PDF annotations
- Table of Contents Viewed
- Table of Contents Selected
- Table of Contents Error reported
- Figure Viewed
- Figure Clicked
- Figure Error reported
- Table Viewed
- Table Clicked
- Table Error reported
- Table Copied contents

The generated report is imported in Tableau, where we have two Dashboards: one for tracking trends of the actions described above (Figure 2) and one for getting cumulative statistics for a given time period (Figure 3).

**Figure 2: CODE Feature Event Trends**
3.2 Performance of CODE Features

3.2.1 Mendeley Desktop

Mendeley Desktop’s document viewer has four tabs: “Details”, “Notes”, “Contents” and “Enrichments”. The usage data presented in this section is for the period March 11th 2014 (official Mendeley Desktop release with CODE features) to April 28th 2014.

3.2.1.1 Table of Contents

The “Contents” tab contains the table of contents that have been extracted by KNOW-Center’s extraction service. The headings in the table of contents become links that you can click on to take you directly to those sections in the pdf viewer. Users can also report errors with Table of Contents. For more detailed description of this feature refer to [Stegmaier2014].

For the period March 11th 2014 (official Mendeley Desktop release with CODE features) to April 28th 2014, about 15.4% of the users who open the “Contents” tab, interacted with the feature by selecting different heading in the table of contents. Overall this feature had 12.15% click-through-rate.

<table>
<thead>
<tr>
<th>&quot;Contents&quot; Tab Events</th>
<th># Unique Users</th>
<th># Unique Documents</th>
<th># Events</th>
</tr>
</thead>
<tbody>
<tr>
<td>TOC Viewed</td>
<td>7,810</td>
<td>9,710</td>
<td>39,790</td>
</tr>
<tr>
<td>TOC Item Selected/Clicked</td>
<td>1,210</td>
<td>1,300</td>
<td>4,830</td>
</tr>
<tr>
<td>TOC Error Reported</td>
<td>10</td>
<td>10</td>
<td>10</td>
</tr>
</tbody>
</table>

Table 1: “Contents” Tab Events for period March 11th 2014 to April 28th 2014
3.2.1.2 Tables and Figures

The “Enrichments” tab shows tables and figures that have been extracted by KNOW-Center’s extraction service. The tables and figures become links that you can click on to take you directly to them in the pdf viewer. Not only is the table extracted but its contents are also recognised, allowing them to be copied and reused in other applications. Users can also report errors with the extracted Figures and Tables. For more detailed description of this feature refer to [Stegmaier2014].

For the period March 11th 2014 (official Mendeley Desktop release with CODE features) to April 28th 2014, about 16% of the users who open the “Enrichments” tab, interacted with the feature by clicking on a table or a figure. Overall this feature had 19.4% click-through-rate for figures or 10.5% for tables.

<table>
<thead>
<tr>
<th>&quot;Enrichments&quot; Tab Events</th>
<th># Unique Users</th>
<th># Unique Documents</th>
<th># Events</th>
</tr>
</thead>
<tbody>
<tr>
<td>&quot;Enrichments&quot; Tab Viewed</td>
<td>2,480</td>
<td>2,610</td>
<td>6,140</td>
</tr>
<tr>
<td>Figure Selected/Clicked</td>
<td>395</td>
<td>455</td>
<td>1,190</td>
</tr>
<tr>
<td>Figure Error Reported</td>
<td>22</td>
<td>28</td>
<td>58</td>
</tr>
<tr>
<td>Table Selected/Clicked</td>
<td>315</td>
<td>355</td>
<td>650</td>
</tr>
<tr>
<td>Table Data Copied</td>
<td>37</td>
<td>42</td>
<td>65</td>
</tr>
<tr>
<td>Table Error Reported</td>
<td>17</td>
<td>18</td>
<td>31</td>
</tr>
</tbody>
</table>

Table 2: “Enrichments” Tab Events for period March 11th 2014 to April 28th 2014

3.2.1.3 Overall Statistics

For the period March 11th 2014 (official Mendeley Desktop release with CODE features) to April 28th 2014, over 8,000 unique users were exposed to the CODE features, and the click-through rate over all CODE features was 14.7%. Over 1,500 unique users clicked on a CODE feature (table of contents, figure or a table).

<table>
<thead>
<tr>
<th>CODE Features Events</th>
<th># Unique Profiles</th>
<th># Unique Documents</th>
<th># Events</th>
</tr>
</thead>
<tbody>
<tr>
<td>CODE_Feature_Viewed</td>
<td>8,225</td>
<td>9,710</td>
<td>45,930</td>
</tr>
<tr>
<td>CODE_Feature_Clicked</td>
<td>1,550</td>
<td>1,740</td>
<td>6,740</td>
</tr>
<tr>
<td>CODE_Feature_Error</td>
<td>38</td>
<td>49</td>
<td>99</td>
</tr>
</tbody>
</table>

Table 3: “Enrichments” or “Contents” Tab Events for period March 11th 2014 to April 28th 2014

3.2.2 Mendeley API

The tables of contents, tables and figures that have been extracted using the KNOW-Center’s services are available through Mendeley’s API. The entities that were extracted using the KNOW-Center’s services and disambiguated using Passau’s services are also available. When a user views a PDF document in Mendeley Desktop, an API call is made to request PDF enrichments. Since, the CODE tools were only run over the Open Access articles in Mendeley’s catalogue, a subset of the API requests would return enrichments.
For the period April 3rd 2014 (API event logging released) to April 28th 2014, we successfully returned PDF enrichments for about 690,000 API requests. There were over 60,000 unique users who were exposed to the CODE features. The average number of daily API events is 30,000, unique documents is 8,000, and unique users is 5,000.

<table>
<thead>
<tr>
<th>API Calls</th>
<th># Unique Users</th>
<th># Unique Documents</th>
<th># Events</th>
</tr>
</thead>
<tbody>
<tr>
<td>Found CODE Enrichments</td>
<td>61,340</td>
<td>108,570</td>
<td>688,560</td>
</tr>
</tbody>
</table>

Table 4: API Events for period April 3rd 2014 to April 28th 2014

3.3 Enrichment Service

Typically the Enrichment Service is invoked in batches of a few thousand PDF articles. These are then processed by the enrichment architecture, which has been specifically tailored towards scalability. The architecture consists of a central queuing system, where incoming requests are persisted. Next to the queue there are a number of worker nodes. The number of workers is variable and thus can be adapted to the current load. The workers themselves operate in a single threading mode, hence each physical node may host multiple workers.

Table 5 gives the raw numbers from one of the enrichment batches sent from Mendeley to the Enrichment Service. In this case the batch consisted of about 15,000 articles, where more than two-thirds of them could be successfully processed and the results of the enrichment were successfully reported back. Within the articles, more than 2 million entities have been identified. Most of the cases, where the enrichment failed was due to download errors, where the PDF could not be retrieved from the Amazon storage area.

<table>
<thead>
<tr>
<th>Measure</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Processed articles</td>
<td>14,533</td>
</tr>
<tr>
<td>Successful enrichments</td>
<td>10,044 (69.11%)</td>
</tr>
<tr>
<td>Extracted entities</td>
<td>2,480,108</td>
</tr>
<tr>
<td>Failure</td>
<td>4,489 (30.89%)</td>
</tr>
<tr>
<td>Download Error</td>
<td>4,299 (29.58%)</td>
</tr>
<tr>
<td>Processing Error</td>
<td>186 (1.28%)</td>
</tr>
<tr>
<td>Response Error</td>
<td>4 (0.03%)</td>
</tr>
</tbody>
</table>

Table 5: PDF Enrichment Service Statistics

The Enrichment Service is typically invoked by backend systems for thousands of PDF documents, therefore scalability is of higher priority (how many articles can be processed) than latency (how long the processing takes).
### 5.3 Monitored Success Factors

Date: 2014-04-30

<table>
<thead>
<tr>
<th>Measure</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average Total Time</td>
<td>49.58 [sec]</td>
</tr>
<tr>
<td>Average Download Time</td>
<td>2.22 [sec]</td>
</tr>
<tr>
<td>Average Processing Time</td>
<td>46.37 [sec]</td>
</tr>
<tr>
<td>Average Image Cache Time</td>
<td>&lt;0.01 [sec]</td>
</tr>
<tr>
<td>Average Result Sending Time</td>
<td>0.93 [sec]</td>
</tr>
</tbody>
</table>

**Table 6: PDF Enrichment Service Processing Times**

The Enrichment Service is hosted on a cluster at the Know-Center, consisting of a number of class servers, for which the number of workers can be easily adapted to the current workload. In **Table 7** the number of scientific articles per day is presented:

<table>
<thead>
<tr>
<th>Number of Workers</th>
<th>Number of Articles per Day</th>
</tr>
</thead>
<tbody>
<tr>
<td>72</td>
<td>125,464</td>
</tr>
<tr>
<td>512</td>
<td>892,187</td>
</tr>
</tbody>
</table>

**Table 7: Daily PDF Enrichments Processing**

Figure 4 shows an estimate of the monthly cost for the Enrichment Service, if it is hosted by the Amazon cloud services and totals at $4,484.10.
Figure 4: Enrichment Service Cost Estimate
4 MindMeister: Semantic Mind Maps

Through CODE MindMeister now provides a semi-automatic extension of mind maps with different Linked Open Data Sources like Freebase. Mind maps, which constitute thesauri, are exported using the SKOS Vocabulary and are then available for re-use in the semantic web. For research papers, MindMeister offers a new import gateway with automated table-of-contents extractions as well as a presentation mode on top of enriched research papers and CODE’s linked data search and visualisation facilities.

4.1 Monitoring Mechanisms

MeisterLabs monitors the uptake of the implemented functionalities though different means. Primarily, the MindMeister event logging framework (“Bee”) is used, but we also implemented standard log file mechanisms, e.g. for the RDF export usage. The “Bee” framework logs all relevant events performed by users on MindMeister. It is easily extensible and logs the following properties per event:

- Date and time
- User ID
- Team ID
- Event Type

It has an AJAX-based web-based query interface that lists all actions performed by users and updates itself automatically. The usage of the Wunderkind feature is logged into Bee, taking into account both fired queries and accepted annotations. Also PDF import uses the Bee database and logs the successful import of PDF documents as well as the extracted and created nodes.

We opted to decouple the logging of RDF exports into a separate log file as the University of Passau crawler performs a lot of automated exports, thus logging into Bee could affect its performance badly. The log files store IP address, date and time, and map ID.

4.2 Performance of CODE Features

4.2.1 Wunderkind

Wunderkind is implemented as a special persistent footer bar in MindMeister. Users first have to activate “Experimental Features” in the settings dialog and then see a new button in the map footer. We log how many users enable the experimental setting, how many queries are performed and how many suggestions are accepted.

As of writing of this document, the Wunderkind feature has not been announced to the MindMeister user base yet. We have decided to postpone launch until mid May to be able to implement some last quality improvements to the functionality. Although Wunderkind is being marketed as “alpha” version, MindMeister is a production tool that many people pay for and therefore expect a minimum quality of service, even in experimental features. We want to make sure not to disappoint our users with a new announcement. However, we are confident that we will be able to present initial usage numbers at the final CODE review meeting at the end of May.

4.2.2 PDF import

Similarly to Wunderkind, the functionality to upload PDF documents has also not been launched yet to the user base. However, we expect the usage of this feature to be low based on the limited intersection of MindMeister users and users who have access to research papers that fit the format...
required for automated extraction. Once the content extraction is generalized to more common PDF documents we expect a larger uptake of users.

4.2.3 RDF export
The RDF export functionality has been available some time on MindMeister and was announced in a blog post on February 19th, 2014. At the moment of writing over 120,000 public maps are available as SKOS thesauri on the web and have been crawled for the University Passau endpoint. In addition to nearly 700,000 automated exports, 72 users from IP addresses all over the world have exported 200 maps to RDF since February.

As has been noted in other documents, manual RDF exports are of limited value to most users and benefits arise mainly through being able to access the maps via the Linked Open Data cloud and the University Passau endpoint. With regards to manual export, we expect much higher uptake through the OPML export interface, which we plan to build on top of the RDF export in the near future.
5 42-data: A data centric question and answering portal

42-Data integrates all CODE services in order to bring Linked Open Data to Linked Data laymen. 42-Data constitutes a data bookmarking and data-centric question and answering portal. Researchers are enabled to explore the Linked Open Data cloud and to collect data from different sources like research papers, or non-semantic open data portals. With the collected data they can engage in discussions using data to underpin their opinion. The portal emphasizes a donation based revenue model. The model has been chosen to allow micropayments among participants without enforcing legal contracts for requesting/delivering data.

5.1 Monitoring Mechanisms

42-data consists of the following components:

- SPARQL endpoint containing Mind-Maps as SKOS Thesauri and RDF Data Cubes created through the portal.
- The portal itself, containing resources (i.e. data centric parts of answers and questions) and users.
- The CODE Query and Visualisation Wizard which can be used from within 42-data or outside of it.
- The CODE Balloon and CODE Disambiguation Services. Details on the Balloon service can be found under http://zaire.dimis.fim.uni-passau.de:8080/balloon/endpoints (username: code, pwd: 123456).

SPARQL Endpoints and web-services are provided using a standard Tomcat server. We utilised log analysis tools to analyse access counts to those services. For the portal itself we used Google Analytics.

5.2 Performance of CODE Features

The public open beta of 42-data was delayed by 3 months, and started on 24th March 2014. Numbers reported are in the period from 24th March 2014 to 31st April 2014. The following numbers and corresponding success factors can be reported:

- Number of accessible LOD endpoints
  - Balloon: 87
  - Used in 42-data: 46
  - Used in Query Wizard: 46
- Algorithmic Annotations per Day
  - Disambiguation Services: around 500,000
    - Estimated through algorithmic tests
- User and usage on 42-data as of 29th April 2014
5.3 Monitored Success Factors

Date: 2014-04-30

- **42 registered users**
  - 409 Unique Visitors, 7584 Pageviews, 6.24 Pages/Session
  - Timeline reveals that everytime we engage in Social Media dissemination we engage new users. However, user registrations are still quite low.

- **Validated Annotations**
  - Number of Data Cube Triples validated by users is approximately 60,000 in 30 data cubes
    - Manual evaluation and annotation of triples using the CODE Disambiguation service

- **Aggregated Data Sets**
  - Number of Data Cubes available via CODE SPARQL Endpoint: 3,928
  - 42-data Resources: 205

- **Query API Calls**
  - SPARQL Calls on mind maps: 24,000
    - Evaluated through log file analysis
  - SPARQL Calls on Data Cubes: 10,530
    - Evaluated through log file analysis
  - Number of Queries consumed through CODE Query Wizard: 93,087
    - Estimated through log file analysis

- **Data Quality**
  - Data Cubes have average quality. The quality has been estimated through random sampling and manual judgement of the University of Passau Team. There have been a significant number of cases where the column types could not be guessed correctly or where a table only consists of numbers.

- **Income to Cost Ratio**
  - Negative. Currently no income is generated with 42-data.

- **Number of Recommendations accepted**
  - NA.
    - We have planned to evaluate this success factor on the Wunderkind services. However, that service goes live on mid of May, so no numbers are available at the time writing this report.

Although 42-data could not achieve full uptake yet, visitor numbers point towards a community interested in data-centric discussions and data bookmarking. Every time 42-data has been
disseminated, we hit a user spike and got interesting replies in social media. However, due to the lack of community and content only few users register. Community building remains the main challenge of 42-data, as in every newly established ecosystem.
6 Sprint Planning

Sprint planning continues as a light-weight model that’s appropriate for all partners. As every partner prefers to use their own system this mostly involves coordination through in person meetings every quarter and regular bi-weekly teleconferencing calls. Results are summarized in the project TRAC and in the issue tracking systems of every partner. We will present the details of sprint planning at the final review meeting, whereas the last Sprint Plan is attached in the Appendix.
7 References

8 Appendix

The appendix covers the sprint-planning document from October 13 to January 14. The sprint planning has been conducted using Google Docs. The appendix has been exported as Word Document “as is” for documentation purpose only.

Scenario 1: Semantic Research Papers
Scenario 2: Semantic MindMaps and Presentations
Scenario 3: Semantic Q&A Platform

Scenario 1: Semantic Research Papers

Lead: KC / Mendeley
This scenario focuses on adding summarisation and navigation support along with special term definitions (e.g. entities) for researchers within Mendeley Desktop. It includes adding to Mendeley Desktop:

1. Table of contents
2. Tables
3. Figures
4. System entities
5. User entities

Overview KC (WP2):
### 5.3 Monitored Success Factors

**Date:** 2014-04-30

---

**Sprint 1: 14 – 25 Oct**

**KC (WP2)**
- Initial work on showcase app for user models
- Send the link to SVN to share POJOs
- Work on deliverable (half-year report)
- Sprint planning

**UP (WP2)**
- Extend API to include categories

**Mendeley**
- Break work into subtasks and prioritise
- Merge the prototype code from the bespoke version of Mendeley Desktop into the master development branch (which means disabling features and adding them back in by rewriting one by one with more stable code)
- Split the Summary tab into a Table of Contents tab and a Summary tab (figures, tables and entities)
- Work on UX to resolve issues with the current prototype such as helping the user to know when extracted data is available for papers and when it is not
- Replace in-memory caching for the enrichment services with redis to make it more stable
- Mendeley Desktop shows tables and figures in the summary tab (not yet linkable to sections in the pdf)
- Mendeley Desktop shows annotated entities as highlighted text in the pdf viewer
- Mendeley Desktop shows table of contents in Table of Contents tab

Questions:

1. When are we expecting to have the response from the enrichment service in JSON-LD format?

Sprint 2: 28 Oct – 8 Nov

KC (WP2)
- JSON-LD (mzechner)
- Showcase app: integrate Mendeley API
- Add captions to NER → linked conceptions to LOD (sklampfl)
- Blacklisting of non-relevant concepts (algorithms & datasets, mkroell)
- Upload presentation to Dropbox (mzechner)
- Evaluation of table extraction on CS papers (sklampfl)

Mendeley
Likely to be a February release

Implemented fetching of system entities in MD and display in Summary tab

Implemented browsing through mentions of system entities in MD in Summary tab

Selection handling to navigate to entries in MD when selected in Summary tab

Assorted UI improvements in MD (hide Contents/Summary tabs when not in use, show progress indicators when fetching data etc.)

Open definition of entity in external browser from MD

Adding API endpoints for user entity creation and deletion

Redis instead of in-memory caching

Created job to export bio-sciences papers

For a given paper, return the system and user entities (deployed)

Working on batch job for returning user and system entities

Questions:

1. Current returned entities have dbpedia links. dbpedia pages are not friendly to present to a user. Would prefer to have Wikipedia link where possible - who should handle the mapping and how?

2. Has anyone looked at the quality of the results that are returned by the NER yet? Algorithm - Batting average, campaign setting, idea, magnitude of descent, new york mets? Metric - Fender Precision Bass,

3. TODO - where does the .csvs for tables come from in ScienceDirect (author created or publisher?)

UP (WP2)

Download arxiv Data (source and pdf)

Sprint 3: 11 – 22 Nov

KC (WP2)

Flag validated annotations (based on categories, mzechner)

Table extraction, focus on improve coverage (sklampfl)

Annotate grant agreements (mkroell) -> moved to sprint #4

Look out for cooperation with JUCS (rkern)

Examples for annotated papers (mkroell)

BioNLP selection of task / initial extraction (rkern)

Showcase app: Integrate models for NER / Relationships (rkern)

Headings & TOC extraction (sklampfl)

Mendeley

UI polish for notification bars, table of contents display and Summary display

Unit tests for table of contents and entity fetch

Local caching of retrieved entities

Presentation of entity definitions in Summary tab (v1, to be revised later)

UI for browsing entity mentions in PDF viewer

Platform services for serving ToC, tables, figures and system entities (including meanings) developed and tested

Working on providing system entities dump

Working on batch enrichment for OA comp sci and bio sci articles

Likely to be a February release

Current plan to enrich OA articles
5.3 Monitored Success Factors
Date: 2014-04-30

Commercially Empowered
Linked Open Data Ecosystems in Research

Monitoring Success Factors

Date: 2014-04-30

UP (WP2)

● Provide first RDF Summarization service.
  ○ Note: found an easy solution consisting of url rewriting for dbpedia urls
  ○ Given a URL returned by the disambiguation service, the Summarization service will
    ■ Return a human readable description (Html Text)
    ■ A link out that points to a human readable web page
    ■ If these informations can not be obtained for a given Semantic Web URL, we
      will return Null
    ■ The service will be provided as source code and as an endpoint for testing.

● Extract Latex Tables and perform rule based annotation

Sprint 4: 25 Nov – 6 Dec

KC (WP2)

● BioNLP relationships (rkern)

● Annotate grant agreements (mkroell) -> new in sprint #4

● Improve the table region extraction (sklampfl) -> new in sprint #4

● Assess the costs of the service (mzechner) -> moved to sprint #5

● Improve computer science annotations (mkroell)

● Improve showcase app, integrate MindMeister (rkern) -> done in sprint #3

● Investigate methods to predict the quality of TOCs (sklampfl) -> started, moved to sprint #5

● Showcase app: integrate images from PDF (rkern)

● Showcase app: export to Latex (sklampfl) -> moved to sprint #5 (sklampfl -> rkern)

● DBPedia -> Wikipedia URL mapping (mzechner) -> waiting for UP

● Integrate new Mendeley API (mzechner, sklampfl) -> started, cont. in sprint #5

Mendeley

● PDF download service is now behind our new API

● Setup OAuth and example code for Graz

● Provided system entity dump

● Batch enrichment
  ○ Documentation for API usage

○ Queue consumer complete (Storm)

● UI improvements for system entities (more consistency)

● Bug fixing on MD

● More efficient processing of the JSON so less lag from interface side

● To hit February release, need to have batch enrichment done before end of next week
  (including both sides)

UP

● Extend the rule base for converting simple tables to rdf cubes
  ○ Rules now cover vertical dimension. Sculley paper can be annotated automatically.

● First quality estimates
  ○ Will be delayed due to problems with latex parsing.
  ○ Only 33k Computer Science papers are available from arxiv

Sprint 5: 9 – 20 Dec

KC (WP2)

● Feature freeze on showcase app (sklampfl, rkern)

● API integration must be finished before 19.12. and batch enrichment started

30/04/2014

- 25 -
Commercially Empowered
Linked Open Data Ecosystems in Research
5.3 Monitored Success Factors
Date: 2014-04-30

Integration query wizard to showcase app (extra feature)? (rkern) -> done in sprint #3
Dropwizard integration (mzechner) -> no longer needed by Mendeley
Integration relationships between computer science entities (mkroell)
Integration of feedback via snapshots (mzechner) -> pending on Mendeley, moved to phase III
Assess the costs of the service (mzechner) -> moved from sprint #4, moved to phase III
Showcase app: export to Latex (rkern) -> moved from sprint #4, moved to phase III

Mendeley
- Setting up queue consumer for the batch enrichment of millions of pdfs
- Making table of contents, tables, figures and entities available through our developer’s API
- Gathering 1,000+ set of articles for QA testing
- QA testinging of the table of contents, table, figure and entities extraction and workflow in Mendeley Desktop

UP
- Investigate learning based approaches for converting complex tables to cubes
  - will be delayed after christmas
  - We got 16k tables having more than 3 rows and with mostly numeric values out of 33k Computer Science arXiv paper. So we have 1 useful table per 2 computer science paper
  - Still open to cubify them.
- Next Step:
  - Rule-based Annotation
  - Import into Big Data (mid of January)

Data Quality Bugs to be Addressed
Here are a number of data quality issues that our QA team has flagged up. Do you have an estimate of when they will be addressed?
1. Download the latest developer build (http://www.mendeley.com/download-mendeley-desktop/preview/)
2. Join this group (http://www.mendeley.com/groups/4050221/) that contains a large number of pdfs that have been selected for testing the CODE features

ISSUE 1 - PDF text parsed and displayed as a table in 'Summary' tab
Another issue that came up when going through our testing set of PDFs: occasionally, a 'table'-type item would be created from seemingly normal-looking PDF text (often appearing in close proximity to an actual table, or describing the contents of a table).

Tags: Documents affected by this issue are tagged with 'text parsed as table' in the CODE project test group.

Comments: May happen if text is not detected correctly as main text or a spurious caption is found.
- Validity check for table structure (estimated effort 1-2 sprints)

Update: This is still an open problem and we are working on this. We have several ideas:
- extend neighborhood relation (in some cases blocks belonging to the same table are not related by the neighborhood relation)
- detect tabular structure (e.g., by correlations of whitespaces across neighboring lines, or by comparing distributions of word gaps)
ISSUE 2 - Entity highlighting misaligned in PDF viewer

An issue whereby all entities listed in a given PDFs 'Summary' tab have their highlighting misaligned with the actual PDF content.

In the two example cases, the PDF highlighting is below and to the left (consistently across the whole PDF file) of the actual figures and tables which are meant to be highlighted.

Attaching some screenshots, as well as the affected PDFs (both are also part of the CODE-testing group on staging)

Bonus issue: Note that in the case of the '92a430af5c83abeb.pdf' file, the PDF highlighting doesn’t match the content displayed in the 'Summary' tab either

**Tags**: Documents affected by this issue are tagged with 'misaligned figure highlighting' in the CODE project test group.

**Comments**: As we directly use the coordinates as reported by PDFBox (without any additional transformations) we can only speculate on the reason for the mis-alignment. Maybe it is due to us using the crop box as reference (as far as I understood the PDFBox code) and your library uses the media box? We could compare the output of PDFBox and another PDF parsing library (e.g. JPod) to see whether there is a disagreement between different implementations.

**Update**: We solved this problem by using only the crop box provided by PDFBox on our side (we were mixing crop box and media box). However, there are still documents in the CODE project test group where there is a significant offset. We do not yet know the source of this error.

ISSUE 3 - Incomplete table highlighted as enrichment item.

When going through our example enriched PDF set. We encountered some cases where tables appearing in the 'Summary' tab would only contain a part of the 'actual' PDF table.

**Tags**: Documents affected by this issue are tagged with 'incomplete table highlights' in the CODE project test group.

**Comment**: Related to issue 1 - the proposed solution would also affect this issue, but we would have to investigate. Still tables spawning multiple pages will not be detected by our current approach.

**Update**: In the documents of the test group this has various reasons. In the document (Baker 2007) the remaining parts of the table are not provided by PDFBox for some reason. In another document (Erbe 2002) the header decoration was included into the table, which could easily be fixed. Other examples require a more sophisticated table region detection (see Issue 1).

ISSUE 4 - Figure enrichments only partially highlighted in the PDF viewer

An issue which affects a number of tested PDFs, whereby only a small fragment of a PDF figure would get highlighted as an 'enrichment' item listed in the 'Summary' tab in MD.

See attached screenshots for specific examples.

**Tags**: Documents affected by this issue are tagged with 'partially highlighted figure' in the CODE project test group.
Comment: Currently we only collect embedded bitmap images, but not rendered content. We assume that in the reported cases the figure itself is drawn by means of PDF operations, but eventually contains embedded bitmap images. Here the images closest to the caption will be picked. A solution would be a bounding box detection for vector images, which is a non-trivial feature.

Update: We improved the detection of the image region by clustering all embedded bitmap images on a page. We then assigned each figure caption to the closest cluster (see Issue 5). This fixed the problem for figures in documents of the CODE project test group that consisted only of embedded images. However, we did not yet consider vector images, so figures consisting (partly) of vector images might still be partially highlighted or missing.

ISSUE 5 - Figures associated with incorrect caption text/label

This is an issue that would occur in some PDFs, especially ones that would have a number of figure items in quick succession. Occasionally, a figure item would be associated with the description text of another figure (e.g., the description of a figure element directly above). See examples/screenshots for a clearer idea.

Tags: Documents affected by this issue are tagged with 'mislabelled figures' in the CODE project test group.

Comment: Currently we don’t exploit the geometric relationship of the image/caption across multiple figures (e.g., caption always on top). We could quickly add some additional heuristical rules to improve the caption to image assignment, but need to conduct some test to assess their effectiveness.

Update: We used a matching algorithm to find the best assignment of captions to images on a single page. The best assignment is the one with the minimal sum of distances between caption and corresponding image bounding boxes.

ISSUE 6 - Entire PDF page highlighted as a ‘Figure’ enrichment

Going through the testing set of enriched PDFs, some PDFs would list entire pages as ‘Figures’.

Tags: Documents affected by this issue are tagged with 'whole page marked as figure' in the CODE project test group.

Comment: This is solved. In some papers page backgrounds seem to be embedded as an image. We handled this by ignoring embedded images spanning the whole page dimension.
Scenario 2: Semantic MindMaps and Presentations

Lead: MeisterLabs

This scenario is aimed at creating semantic mind maps from PDF documents (Table of Contents) or linked data enrichment. It should be possible to create presentations from mind maps and integrate WP4 visualizations (as images). This requires the following steps:

- add enrichment features to mind map editor
- represent and expose mind maps and related data as RDF
- enhance external view for mind maps (incl. save / login features, adding slides automatically, including images with backlink, etc)
- enhance the Wunderkind feature using semantic web sources (and possibly Mendeley)

Sprint 1: 14 – 25 Oct

MeisterLabs

- RDF Export of Maps
  - Improvements, Person Export Planning
- Embedded Map Display
  - Map editable when logged in and anonymous (done)
  - Add links and images via JSON (done)
  - Automatic presentation creation (done)
- Wunderkind
  - Started work using wikidata

KC (WP4)

- Query Wizard
  - Improve display of MindMap in Query Wizard (done)
- Vis Wizard
5.3 Monitored Success Factors

Date: 2014-04-30

- Add Visualization to new MindMap (done)

Sprint 2: 28 Oct – 8 Nov

**MeisterLabs**

- RDF Export of Maps
  - Define format for node URLs in MindMap JSON (done)
  - Create MindMeister RDF Dump (with Florian/Emanuel) (done)
- Embedded Map Display
  - Login Button (done)

- Save to My Maps option (done)

Sprint 3: 11 – 22 Nov

**MeisterLabs**

- REST API call mm.maps.getPublicList: added date limit parameter (done)
- Bug fixes of RDF export (rdfs:label) (done)
- Wunderkind
  - Public maps source improvements (done)
  - DBpedia as source tests (done)
- Presentation stuff
  - Export presentations as images zip (done)

**KC (WP4)**

- Query Wizard
  - Provide access to MindMeister SPARQL Endpoint (done)
Sprint 4: 25 Nov – 6 Dec

MeisterLabs

- Expose user channel page as RDF (done)

User channel has been updated in latest release with multiple new features such as maps I liked, or recommended maps for me (based on a comparison of my maps with other public maps, will be delivered later)

User channel has been exposed as RDF, see e.g. http://www.mindmeister.com/de/users/channel/mhollauf.rdf

- See how to generate (presentation) mindmap from research paper (PDF import, extraction, mind map generation) - currently in progress / discussion with Roman, will try use
synchronous PDF extractor service
- Back link to visualisation from image (should already work, if not, contact Patrick) – should work already
- WunderKind work - try Passau endpoint, Google KG

Sprint 5: 9 – 20 Dec
MeisterLabs
- Launched and publicised RDF export (many questions for OPML :-) ) - more to come
  - UP publishes paper
  - Publicise to
    - W3C community
30/04/2014 - 33 -

MM API partners
Blog post

- Wunderkind:
  - Tried reaching the UP Dbpedia SPARQL endpoint
  - Google Knowledge Graph would be interesting! - Florian sends info about KG endpoint

- Working on new Wunderkind UI
- PDF extraction / import with Roman

UP

- Google Knowledge Graph --> Currently being indexed
- Create SPARQL Query to show relations among mindmap nodes

KC (WP4)

- Vis Wizard
  - Generate MindMap for multiple charts (including preview images and backlinks)

Sprint 6: 9 – 30 Jan

Scenario 3: Semantic Q&A Platform

Lead: UP

[Insert short description here]
Sprint 1: 14 – 25 Oct
UP
  ● see gantt
KC (WP4)
  ● Query Wizard
    ○ “Aggregate dataset” functionality (done)
  ● Vis Wizard
    ○ Coordinated brush foundation (done)

Sprint 2: 28 Oct – 8 Nov
UP
  ● ...
KC (WP4)
  ● Query Wizard
    ○ Add multiple aggregated values for a measure (done)
    ○ Include metadata when cubifying via instant visualization (done)
    ○ Include metadata when cubifying via aggregation (done)
    ○ “Group by this column” Button (done)
    ○ Numeric filter (done)
    ○ Filter out empty results (done)
    ○ Display SPARQL query runtimes in console (done)
  ● Vis Wizard
    ○ Aggregate dataset functionality (done)
    ○ Coordinated brush by multiple view for Stream Graph (done)
    ○ Coordinated brush by multiple view for Grouped Bar Chart (done)

Sprint 3: 11 – 22 Nov
UP
  ● ...
KC (WP4)
  ● Query Wizard
    ○ Add “Open in Browser” for URI drop downs (done)
    ○ Display SPARQL query runtimes in user interface (done)
    ○ Provide dataset description when visualizing results (done)
    ○ Display SPARQL queries (done)
    ○ Display message if endpoint is slow (done)
    ○ Display spinner in aggregation modal (done)
    ○ Add search filter (done)
    ○ Save cube & query to Q&A Portal (done)
    ○ Add date & datetime filter (done)
  ● Vis Wizard
    ○ Coordinated brush by multiple view for Parallel Coordinates (done)
    ○ Coordinated brush by multiple view for Scatterplot matrix (done)
    ○ Coordinated brush by multiple view for Pie chart (done)
    ○ Coordinated brush by multiple view for Geomap (done)
Sprint 4: 25 Nov – 6 Dec

UP
- First Deployment
- Profile Page ready

KC (WP4)
- Query Wizard
  - Export displayed data as JSON-LD (done)
- Vis Wizard
  - Filtering functionality (done)
  - Multiple coordinated view for more than one Cube (done)

Sprint 5: 9 – 20 Dec

UP
- PayPal based donations
- Bookmarking of Vis Wizard results
- Bookmarking of Query Wizard results
- MindMap integration

KC (WP4)
- Query Wizard
  - Replace Endpoint IDs with URLs (done)
  - Get data with single SPARQL query (done)
- Vis Wizard
  - Coordinated brush by multiple view for D3 Barchart (done)

**UPDATED SPRINT PLANNING**
5.3 Monitored Success Factors

Date: 2014-04-30

2014 Task List

UP
- Q&A Portal Advertisement and Dissemination
- Python Data Cube Extension and Ipython notebook support (mgrani)
- Data Cube Extensions (for Data Mining)
- Data Cube Update?
  - Could we allow to update cubes from outside the consortium?

KC
- WP2
  - Integration of feedback via snapshots (mzechner)
  - Assess the costs of the services (mzechner)
  - Showcase app: export to Latex (rkern)
  - Re-run table evaluation with fresh data from Mendeley, e.g. (sklampfl)
  - Trac update, e.g. documentation (mzechner)
  - Investigate methods to predict the quality of TOCs (sklampfl, rkern)
  - Integrate ML methods for CS NER (mkroell)
  - Improve relations between CS entities (mkroell)
- Query Wizard
  - Dependent columns
  - Filter out wrong statements (a.k.a. crowd-sourced quality feedback)
  - Add concepts to be extended by their properties (e.g. Traversal of thesaurus)
Commercially Empowered Linked Open Data Ecosystems in Research

5.3 Monitored Success Factors
Date: 2014-04-30

- Display all available object URIs in a filter dialog
- “Focus on” feature for multi-URI cells
- Sorting
- BALOON support (needs evaluation once Q&A portal is live)

- Vis Wizard:
  - Integration of the visualizations in the Q&A platform
  - Usability, Bug Fixing
  - Concrete Graphics (what’s that?)

MeisterLabs
- UI Design / CSS Creation 42data
- WunderKind Enhancements
  - use auto disambiguation with confidence barrier
  - parallel query of multiple sources
  - Google Knowledge Graph inclusion
  - new UI
- OAuth2 interface for MindMeister (with dynamic redirect)

KC - WP2 - Sprints

**Sprint #1/14 - 13.01. - 24.01.**

- Enrichment without entities (sklampfl)
- Investigate encoding problem (sklampfl)
- Re-run table evaluation on modified Mendeley data set (sklampfl)
- Remove duplicate annotations (sklampfl)
- Improve usability of showcase app (rkern, sklampfl)

**Sprint #2/14 - 27.01. - 07.02**
5.3 Monitored Success Factors

Date: 2014-04-30

- UDL Paper review (sklampfl)
- Cost of service (mzechner)
- Advanced table evaluation (sklampfl)
- Improve relationships between NE classes (mkroell) -> moved to sprint #3
- TRAC update (mzechner) -> moved to sprint #3
- Compare bboxes of NE between PDFBox and JPod (rkern)
- Improve heuristics for caption/image assignment (sklampfl)
- Add tabular structure detection to improve table region detection (sklampfl) -> moved to sprint #3
- Add showcase app to DS.2 (rkern) -> new
- JSON-LD Issue (cors) (mzechner) -> new
- Relation annotator (mkroell) -> new
- Include flag for enabling entity detection (sklampfl) -> new
- Activate image cache for synchronous calls (sklampfl) -> new

Sprint #3/14 - 10.02. - 21.02.

- KWIC User Interface ½ (rkern)
- Better URL for showcase app (sklampfl, georg)
- Check login at new showcase URL (sklampfl)
- TRAC update (mzechner) -> moved to sprint #4
- Update URL at CODE homepage (rkern, phoefer)
- Plan export model to Enrichment Service (sklapfl)
- Add relations to annotated papers ½ (mkroell)

Sprint #4/14 - 24.02. - 07.03.
5.3 Monitored Success Factors

Date: 2014-04-30

Sprint #4/2014 - 24.02 - 07.03

- KWIC User Interface 2/2 (rkern)
- Next/Prev in the annotation view (sklampfl)
- Similar document search (rkern)
- Tables extraction paper ½ (sklampfl)
- CS entities paper ½ (mkroell)
- TRAC update (mzechner) -> moved to sprint #5

Sprint #5/2014 - 10.3 - 21.3

- Table extraction paper 2/2 (sklampfl)
- Computer science paper 2/2 (mkroell)
- Import SKOS thesaurus (rkern)
- Share model (rkern)
- Improve KWIC keyboard interactions (rkern)
- Add relations to annotated papers (mkroell)
- Screencasts for annotator tool (sklampfl)
- TRAC update (mzechner) -> moved to sprint #6

Sprint #6 - 24.3. - 14.4.
- Update Trac (mzechner)
- SADAATL paper 1/2 (mkroell)
- Send Roman consumed hours (all)
- Integrate shared model in Enrichment Service
- D6.2 collection of success factors (rkern, sklampfl)