



1914  
1918  
ONLINE

International  
Encyclopedia  
of the First World War

# PROJECT DOCUMENTATION

*Karsten Borgmann, Susanne Büchner, Ilker Egilmez, Martin Gruner, Christoph Schimmel*  
June 2015

## Table of Contents

1. The Project
2. Technical Documentation
  - 2.1. Why a Wiki
  - 2.2. Description of SMW Forms, Templates and Properties
    - 2.2.1. Content Bundles
    - 2.2.2. Forms
      - 2.2.2.1. Query Forms
      - 2.2.2.2. Project Forms
    - 2.2.3. Templates
      - 2.2.3.1. Main Templates
      - 2.2.3.2. Layout Templates
      - 2.2.3.3. Module Templates
      - 2.2.3.4. Query Templates
    - 2.2.4. Properties
  - 2.3. Dynamic Default Values for Properties

- 2.4. Workflow
  - 2.4.1. General Workflow
    - 2.4.1.1. Preparation
    - 2.4.1.2. Bibliography
    - 2.4.1.3. Compiling an Article and Metadata
  - 2.4.2. Additional Workflow
    - 2.4.2.1. Adding Geodata
    - 2.4.2.2. Adding Related Events
- 2.5. Generate DOI
  - 2.5.1. Important Details
  - 2.5.2. Workflow
    - 2.5.2.1. Conversion to XML
- 2.6. Data Structure
  - 2.6.1. Metadata
    - 2.6.1.1. EntityType
    - 2.6.1.2. EntitySubType
    - 2.6.1.3. AdditionalSubType
  - 2.6.2. Keyword Vocabulary
    - 2.6.2.1. GND subject headings
    - 2.6.2.2. Library of Congress
    - 2.6.2.3. Rameau subject
- 2.7. Basic Interfaces & Extensions
  - 2.7.1. Zotero
  - 2.7.2. Geodata
  - 2.7.3. Timeline Events
  - 2.7.4. PDF Print Preview
- 3. User Documentation

## 1. The Project

On the 100th anniversary of the start of the First World War, “1914-1918-online. International Encyclopedia of the First World War” presents a virtual, open access handbook and encyclopedia on the First World War to both the academic community and the general public. The multi-perspective, refereed reference work is the result of an international collaborative project initiated by Freie Universität Berlin (Friedrich-Meinecke-Institut and Center for Digital Systems) in cooperation with the Bavarian State Library and funded by the German Research Foundation. “1914-1918-online” is edited by Ute Daniel, Peter Gatrell, Oliver Janz, Heather Jones, Jennifer Keene, Alan Kramer, and Bill Nasson, in association with more than seventy thematic and regional section editors. The publication language is English.

“1914-1918-online” sets new standards for e-Humanities. The project has developed and tested model navigation procedures for thematic knowledge spaces that enable users to find their way through complex, non-linear texts. Focusing on themes (e.g. propaganda), it is possible to navigate through the material by following threads for certain questions. It will also become possible to integrate diverse texts in broader contexts in a way that is appropriate to the hypertextuality of the medium.

## 2. Technical Documentation

### 2.1. Why a Wiki?

At the beginning of the project, several software options were evaluated. Semantic MediaWiki (SMW) was chosen for the following reasons:

- it provides nonlinear access to a website's content
- easy and simple editing functions
- free and open source software with a large number of developers and is based on php, MySQL and Apache, etc.
- several similar projects also use SMW, which makes it easier to tackle bugs and to compile features
- it visualizes the relations between a website's articles and pages
- simplifies the editorial work for international projects with decentralized processing
- enables extensions and further programming methods
- allows a clear distribution of user rights
- offers good options for individual data models and for a flexible classification of objects
- features a broad set of interfaces for the import and export of data
- presents wide functionalities in the field of Web 2.0
- improves the user's orientation and intuitive navigation on a website with a lot of nonlinear and intertwined content

## 2.2. Description of SMW Forms, Templates and Properties

### 2.2.1. Content Bundles

Content Bundles are groups of wiki pages with various functionalities (Properties, Templates, Forms etc.) that can be exported and installed in another MediaWiki platform.

### 2.2.2. Forms

Forms provide a form-based user interface to edit wiki articles. The extension *Semantic Forms* allows you to have forms for adding, editing and querying data on your wiki, without any programming. Forms can be created and edited not just by administrators, but by users themselves. The most important forms within this project are the "Location"-form for creating and editing geodata, and the "TimelineEvent"-form for creating and editing Timeline Events. When editing a Timeline Event, there is the option of connecting this data with other articles or geodata existing in the project. The same applies to the "Location"-form, where the admin user can connect a geolocation to selected articles.

#### 2.2.2.1. Query Forms

Query Forms are forms that are used to initiate a SMW inline query.

- ArticleQuery: (Query form for articles)
- ExternalLinksQuery: (Query form to search for external links)
- ImageQuery: (Query form to search for images)
- PersonQuery: (Query form to search for persons)

#### 2.2.2.2. Project Forms

- Form:ArticleQuery: (Query form for articles)
- Form:ExternalLinksQuery: (Query form to search for external links)
- Form:ImageQuery: (Query form to search for images)
- Form:LitFeedSource: (Form to input a ZOTERO collection tag.)
- Form:Location: (Form to create or edit Geodata)
- Form:PersonQuery: (Query form to search for persons)
- Form:ProjectBug: (Form to create or edit a project task.)
- Form:ProjectDocumentEE: (Form to create or edit Encyclopedic Entries (Lemmata))
- Form:ProjectDocumentHB: (Form to create or edit ProjectArticles (Handbook Articles))
- Form:ProjectDocumentIntroduction: (Form to create or edit ProjectArticles (Introduction Articles))
- Form:ProjectDocumentation: (Form to create or edit a project task.)

- Form:ProjectDocumentationQuery: (Query form for articles)
- Form:ProjectFeature: (Form to create or edit a project feature.)
- Form:ProjectFeatureQuery: (Query form for articles)
- Form:ProjectItem: (Form to create or edit a project task.)
- Form:ProjectItemQuery: (Query form for articles)
- Form:ProjectMilestone: (Form to create or edit a project milestone)
- Form:ProjectPerson: (Form to enter information about a person related to the project)
- Form:ProjectSprint: (Form to create or edit a project sprint)
- Form:ProjectTarget: (Form to create or edit a project milestone)
- Form:ProjectTask: (Form to create or edit a project task.)
- Form:ProjectUserStory: (Form to create or edit a project task.)
- Form:TimelineEvent: (Form to create or edit Timeline Events.)
- Form:WorkBib: (Form for (external) images saved in the local wiki.)
- Form:WorkBibliographyEntry: (Form to change the display of an Bibliography Entry)
- Form:WorkExternalLink: (Form to create or edit "External Links" (related web-resources etc.))
- Form:WorkImage: (Form for (external) images saved in the local wiki.)

### 2.2.3. Templates

Templates are standard wiki pages whose content is designed to be transcluded (embedded) inside other pages. In this wiki there are four major types of templates: MainTemplates, LayoutTemplates, ModuleTemplates and QueryTemplates. You can read more about templates and how they work on the [MediaWiki Help-Page](#).

#### 2.2.3.1. Main Templates

Main templates define the layout "frame" and all the properties for the main content groups of the wiki-project. You might think of them as objects that include some common and some special page attributes. The attributes for the main content types are set in the metadata section of the Main Template. The Main Templates also allow customization/arrangement of the basic page layout / css structure.

- Concept: Template to define pages of class "Concept" (Categories, Lists, Reports etc.). All parameters this template takes are the names of the properties that define the Category Concept. Please check the page source for more information.
- Geo: Template to define locations used in article maps. Its Parameters: HasCoordinates. All other parameters this template takes are the names of the properties that define the category

Category:GeoCoordinate. Please check the page source for more information.

- PageDocumentation: Main Template to define properties and layout of wiki-articles used for documentation.
- Part of bundle: Relates a system element to a content-bundle.
- Rationale: Gives a short definition of the function of the page.
- TechnicalComment: Longer description of technical particularities of the page.
- ContentType: Relates the system article to a content area of the bundle.
- ProjectDocument: Template to define pages of class ProjectDocument (Publications). All parameters this template takes are the names of the properties that define the category ProjectDocument. Please check the page source for more information.
- Timeline: Template to define Timeline Events. All parameters this template takes are the names of the properties that define the category TimelineEvent. Please check the page source for more information.
- Work: Template to define pages that include descriptions of external resources (Images, Webpages, Databases). All parameters this template takes are the names of the properties that define the Category Work. Please check the page source for more information.

### 2.2.3.2. Layout Templates

Layout Templates are containers for particular page content. In this wiki there are three containers that can be used: Template:PageHeader, Template:Content and Template:Appendix.

1. PageHeader: Navigation-Elements, edit-links, Article-Title
2. Content: The main content of the article.
3. Appendix: Everything that is related to the article (images, appendix etc.)

### 2.2.3.3. Module Templates

Module templates define various functional elements within the different layout areas of the wiki pages and are called from the layout templates.

- AddButton
- AddEditButtons
- AddEditButtonsWork
- AdminButtons
- ArticleCitation
- BibliographyListAppendix

- DisplayMap
- DocumentationToc
- EditListButton
- ExternalLinksListAppendix
- H3
- ImageListAppendix
- PageDocumentationMeta
- PageHeading
- PersonData
- ProjectDocumentToc
- PublicationList
- ResourceListAppendix

#### 2.2.3.4. Query Templates

Query Templates receive their values from SMW inline-queries or Query Forms and are used to format the output of the query.

- ArticleReportQt: Formats a tabular output for Template:ArticleReport.
- DisplayTwoValues: Simply displays two values as a result of a query that has two query parameters.
- ExternalLinksListAdminQuery: Query template to display admin lists of external links in content area.
- ListItem: Query Template to display all kinds of articles as a list item.
- ProjectItemListContentQt: Formats the results for Template:ProjectItemListContent
- ResourceListAdminQuery: Query template for form article query
- ResourceListAppendixQt: Formats the results for Template:ResourceListAppendix

#### 2.2.4. Properties

Properties define the SMW page attributes. For more information on how properties work within a SMW, please read [this](#). Furthermore, there are Special Properties that are needed to make the whole SMW system work. There are different categories of properties:

1. Generic properties for all wiki elements
2. Properties used in project publications
3. Properties that describe "person"-related information
4. Properties needed for displaying geographical information
5. Properties needed for displaying event related information

6. Properties needed for article workflow
7. Properties from domain "work"
8. Properties imported from external namespaces
9. Properties currently not used (deprecated)

Below is a full list of properties created within this project:

- AcceptanceDate: (Date when a publisher has given permission to publish a resource.)
- AccessDate: (Date when the resource was last accessed)
- AcquisitionCost: (Price for acquiring an image or other resource.)
- AdditionalAuthors: (Helper Property. Legacy value from JIRA import field 'Additional Authors'). See <http://en.wikipedia.org/wiki/JIRA> for more information on the proprietary software product JIRA.
- AdditionalSubType: (Defines specific sub-types for the main content classes.)
- AddressLocalityName: (Saves place of publication. Part of Cite Template)
- Affiliation: (Information about affiliation of person)
- AlternativeFirstName: (Alternative first name of a person)
- AlternativeLastName: (Alternative last name of a person)
- AppendixComplete: (Saves value for publishing status "media file complete")
- ArticleCitation: (Saves the "recommended citation" of an article in a property.)
- ArticleName: (Saves the article name (Page title).)
- ArticleOutline: (Saves content of article outline from JIRA)
- ArticleType: (Saves values for different project publication formats, e.g. "handbook article", "blog-article" etc.)
- At: (Page reference within the source. Used in Wikipedia Citation template <http://en.wikipedia.org/wiki/Template:Citation/core>)
- AuthorAgreementSigned: (Saves value for publishing status when author has documented his agreement.)
- Autocomplete on: (Use this property to annotate a property article if you want to overwrite the default auto completion constraints that are used for this property in Automatic Semantic Forms. You can use the same constraints, which are also allowed for the constraints parameter for form input fields. This only works if Halo Autocompleting is used.)
- BelongsToMainRegionalArticle: (Saves the "Father" / "mother" Article)
- BelongsToMainThematicArticle: (Saves the "Father" / "mother" Article of a Regional Thematic Article.)
- BelongsToTermImport: (Relates a term import run to its term import definition. This property is set by the system and should not be changed by the user)



- BelongsToTermImportWithLabel: (Relates a term import run to its term import definition. This property is set by the system and should not be changed by the user)
- BibDisplayText: (Saves text for "literal" bibliographic information to be displayed instead of the bibliographic information stored in the database)
- BibliographyAdded: (Saves value for publishing status when the article bibliography is complete.)
- BiboAuthorList: (An ordered list of all authors. Normally this list is seen as a priority list that orders authors by importance.)
- BiboIssue: (Part of Cite Template)
- BiboVolume: (Part of Cite Template)
- Call: (Which number should be called (This property is used by the Automatic Semantic Forms sandbox.))
- Coauthors: ((deprecated) Names of coauthors. Last names of further editors. Used in Wikipedia Citation template Cite book. Was dropped in <http://en.wikipedia.org/wiki/Template:Citation/core> in favor of calling editor 1-4 names. Is preserved as legacy parameter.)
- CollectionItemTag: (Saves value for a tag in an external collection.)
- ContactPerson: (Persons with more information about a piece of content.)
- Content hash: (Part of Cite Template)
- ContentCategories: (Saves value for content categorization among project documents and multimedia materials. Needed to control dynamic form generation provided by the ASF extension.)
- ContentType: (Defines if this element is considered as content, project-related content or system related article.)
- CorrespondingAuthor: (Helper Property. Legacy value from JIRA import field 'Corresponding Author'.)
- CreatorURL: (URL of the person responsible for creating the resource.)
- CtagTagged: (The (uncontrolled) user tags or keywords that apply to a resource.)
- Date: (Date of the authorship. Is used as date of publication here since Zotero RDF uses dcterms:date. Used in Wikipedia Citation template. <http://en.wikipedia.org/wiki/Template:Citation/core>)
- DctermsCreator: (An entity primarily responsible for making the resource.)
- DctermsDescription: (An account of the resource.)
- DctermsIdentifier: (An unambiguous reference to the resource within a given context.)
- DctermsIsReferencedBy: (Article that references, cites, or otherwise points to the described resource.)
- DctermsIssued: (Date of formal issuance (e.g., publication) of the resource.)
- DctermsLicense: (A legal document giving official permission to do something with the resource.)

- DctermsPublisher: (Examples of a publisher include a person, an organization, or a service. ProjectDocuments are published by the project, any other resources by the source publisher.)
- DctermsReferences: (A related resource that is referenced or otherwise pointed to by the described entity.)
- DctermsRights: (Information about rights held in and over the resource. Typically, rights information includes a statement about various property rights associated with the resource, including intellectual property rights.)
- DctermsRightsDescription: (Description of usage rights.)
- DctermsSource: (A related resource from which the described resource is derived.)
- DctermsTitle: (Used to describe the label of a resource. Should be differentiated from ArticleName used for URL-friendly Wiki Article Names)
- DctermsType: (This is a special property delivered with SMW.)
- DdcGeographicAreaCode: (Saves area code from gnd (DDC-Hilfstafel 2))
- Delimiter: (Users may enter several values in a form input field for a property in an automatically created Semantic Form. Use the delimiter annotation to choose the delimiter)
- DescribesEntity: (Relates a project document or an image to an entity.)
- EditorGiven1: (First name of first editor. Used in Wikipedia Citation template. <http://en.wikipedia.org/wiki/Template:Citation/core>)
- EditorGiven2: (First name of second editor. Used in Wikipedia Citation template. <http://en.wikipedia.org/wiki/Template:Citation/core>)
- EditorGiven3: (First name of third editor. Used in Wikipedia Citation template. <http://en.wikipedia.org/wiki/Template:Citation/core>)
- EditorGiven4: (First name of fourth editor. Used in Wikipedia Citation template. <http://en.wikipedia.org/wiki/Template:Citation/core>)
- EditorSurname1: (Last name of first editor. Used in Wikipedia Citation template. <http://en.wikipedia.org/wiki/Template:Citation/core>)
- EditorSurname2: (Last name of second editor. Used in Wikipedia Citation template. <http://en.wikipedia.org/wiki/Template:Citation/core>)
- EditorSurname3: (Last name of third editor. Used in Wikipedia Citation template. <http://en.wikipedia.org/wiki/Template:Citation/core>)
- EditorSurname4: (Last name of fourth editor. Used in Wikipedia Citation template. <http://en.wikipedia.org/wiki/Template:Citation/core>)
- EditorsNote: (Remarks about the status of the document in the publishing workflow.)
- Enclosures: (Links to files that were embedded in an imported Feed Item.)
- EndDate: (End date of an event.)
- EntitySubType: (Defines sub-type and category value for the main content classes (ProjectDocument,Work,Concept etc..))

- EntityType: (Saves value for the main entity type described by the wiki-article. Equals the main template-name and the category-name of the article. Properties and templates are described as "Wikipage".)
- EventDate: (Saves temporal values for single events that are described in an article or work.)
- ExternalUrl: (Saves values of URLs for resources of class external resource)
- Field sequence number: (The Automatic Semantic Forms Extension orders form input fields for properties alphabetically. Use the Field sequence number annotation on property articles to reorder their form input fields.)
- FileFormat: (Describes the technical file format of media files.)
- Filename: (Saves name of uploaded files.)
- FmiEntityLabel: (Saves legacy entity label used by the FMI editors)
- FoafCurrentProject: (A current project this person is currently working on.)
- FoafDepiction: (This is a special property delivered with SMW.)
- FoafDepictionFor: (Relates an image as a depiction for one or many articles.)
- FoafFamilyName: (The family name of some person.)
- FoafGivenName: (The given name of some person.)
- FoafPastProject: (A project this person has previously worked on.)
- FoafTitle (The title of a person)
- Form input help: (Use this property to annotate a property if you want a help icon with a help message to appear next to the form input field of the property in an automatically created Semantic Form.)
- Form input label: (Use this property to annotate a property article if you want to choose an alternative label for its form input field in an automatically created Semantic Form.)
- FormHelpText (Text explaining the form and how it should be filled out)
- Given1: (First name of first author. Used in Wikipedia Citation template. <http://en.wikipedia.org/wiki/Template:Citation/core>)
- Given2: (First name of second author. Used in Wikipedia Citation template. <http://en.wikipedia.org/wiki/Template:Citation/core>)
- Given3: (First name of third author. Used in Wikipedia Citation template. <http://en.wikipedia.org/wiki/Template:Citation/core>)
- Given4: (First name of fourth author. Used in Wikipedia Citation template. <http://en.wikipedia.org/wiki/Template:Citation/core>)
- GndAcademicTitle: (Saves academic titles of a person.)
- GndDateOfEstablishment: (Date when an organization was established.)
- GndDateOfTermination: (An account of the resource.)
- GndFunctionOfThePerson: (The most significant position of a CV.)
- GndGeographicAreaCode: (Saves area code from GND.)
- GndGeographicAreaCodeLabel: (Saves label for area code from gnd.)

- GndIdentifier: (Identifier from the German National Library Authority File GND.)
- GndInText: (GND-ID within article text. BSB-Annotation)
- GndOrganizationType: (Saves organization class from gnd.)
- GndPreferredNameForTheCorporateBody: (Preferred name for the corporate body (GND:Bevorzugter Name der Koerperschaft))
- GndPreferredNameForThePerson: (Saves name as given in the GND)
- GndPreferredNameForTheSubjectHeading: (Saves subject heading from GND)
- GndReady: (Flags if GND and BSB-DDC is done)
- GndSpatialAreaOfActivity: (An account of the resource.)
- GndSubjectHeadingIdentifiers: (Saves subject heading id from gnd.)
- GndSucceedingCorporateBody: (An account of the resource.)
- GndVariantNameForTheCorporateBody: (An account of the resource.)
- GndVariantNameForThePerson: (Additional family names.)
- HandbookArticleType: (Defines sub-type and category value for the main content classes.)
- Has Import Date: (Denotes the import date of a term import run. This property is set by the system and should not be changed by the user.)
- Has URL: (This property stores the URL of a feed or a feed item.)
- Has author: (This property links to the author of a feed item.)
- Has content: (This property stores the content of a feed item.)
- Has contributor: (This property stores who the contributor of a feed item was.)
- Has copyright: (This property stores the copyright information for this feed item.)
- Has id: (This property stores the unique id of a feed item.)
- Has label (Identifier if this property has a label or not)
- Has latitude: (This property stores the value of the latitude annotation of a feed item.)
- Has local publication date: (This property stores the local publication date of this feed item.)
- Has longitude: (This property stores the value of the longitude annotation of a feed item.)
- Has permalink: (This property stores the permalink of this feed item.)
- Has publication date: (This property stores the publication date of this feed item.)
- Has source: (This property stores the original source of a feed item if it was republished.)
- Has subject: (This property stores the subject value of a feed item.)
- Has sub object (This property has a sub object value)
- Has tag: (This property stores the values of the category annotations of an imported feed item.)

- HasAlternativePagenames: (Saves value for synonyms and redirects of the article.)
- HasBibEntries: (Saves page name of related Bibliography Entries.)
- HasCaption: (The caption for an image)
- HasCategories: (Saves values for categorization of any ProjectDocument)
- HasCollectionSource: (List of institutions as main online resources for images, links etc.)
- HasCoordinates: (Saves geo coordinates for entries and articles)
- HasCountries: (Saves the countries belonging to an article)
- HasDDCSubjectCatalogue: (Saves DDC-SubjectCatalogue label (Sacherschliessung) by BSB)
- HasDDCSubjectCatalogueId: (Saves DDC-SubjectCatalogue ID (Sacherschliessung) by BSB)
- HasDdcGermanDescription: (German description of Dewey Decimal Classification (DDC))
- HasDdcGermanLabel: (German Label of DDC)
- HasDoi,DOI (Digital Object Identifier)
- HasDoiURN: (Saves URL with DOI identifier.)
- HasDoiVersion: (DOI Version)
- HasDomains: (Relates a property to a systematic content area of the wiki. Used for arranging the documentation.)
- HasDynamicExternalLinks: (Saves page name of related External Links.)
- HasEndDate: (Saves End Date of an Event on the Timeline.)
- HasExternalLinks: (Flags documents that are referred to in webpage descriptions (External Links).)
- HasGndIdentifier: (Identifier from the German National Library Authority File GND.)
- HasHistoricalPersonName: (Historical name of a person.)
- HasId: (Miscellaneous identifier, other than e.g. DOI ISBN etc. Use for archive call numbers. Used as "ID" in Wikipedia Citation template. <http://en.wikipedia.org/wiki/Template:Citation/core>)
- HasIsbn: (International Standard Book Number)
- HasKeyEvent: (Saves name of an "Event" article.)
- HasKeyLocations: (Saves name of Location.)
- HasKeywords: (Saves value for controlled keyword vocabulary.)
- HasLanguage: (Used to describe the language of a resource in common language not in a controlled vocabulary e.g. DctermsLanguage)
- HasLcshIdentifier: (Identifier from the LCSH)
- HasMainRegionalSections: (Saves regional category information about an article or work.)

- HasMainThematicSections: (Saves thematic category information about an article or work.)
- HasMedia: (Saves page name of a related media file.)
- HasNegativeRegionalSections: (Saves negative regional category information about an article or work.)
- HasNegativeThematicSections: (Saves negative thematic category information about an article or work.)
- HasNegativeTopics: (Saves negative topic/taxonomy category information about an article or work.)
- HasParentEvents: (Parent Events to a TimelineEvent.)
- HasRameauldentifier: (Saves from the Répertoire d'autorité matière encyclopédique et alphabétique unifié (RAMEAU) subject classification.) See RAMEAU for more information.
- HasReferredAdditionalSubTypes: (Saves Referred additional SubType about an article or work.)
- HasReferredEntitySubTypes: (Saves Referred Entity Subtype about an article or work.)
- HasReferredEntityType: (Saves Referred Entity Type about an article or work.)
- HasRegion: (Saves regional sub classification)
- HasRegionCategoryLevel: (Level of category region)
- HasRegionalCategory: (Saves regional category information about the project document.)
- HasRegionalSurveyArticle: (Saves name of a regional survey article that should be referenced by a regionalized thematic article.)
- HasRegions: (Saves value for controlled regions-vocabulary)
- HasStartDate: (Saves Start Date of an Event on the Timeline.)
- HasSubThematicSections: (Saves the sub thematic content sections.)
- HasSummary: (Saves article summary)
- HasSynonym: (Synonym pages for an article (Redirections))
- HasTaxonomy: (Taxonomy-Keywords from FMI.)
- HasThematicSurveyArticle: (Saves name of a thematic survey article that should be referenced by a regionalized thematic article.)
- HasTimelineEvents: (Saves name of Timeline-Events.)
- HasTopCategory: (Relates a category to the next category level.)
- HasTopicCategoryLevel: (Saves hierarchy-level of ontology categories)
- HasTopics: (Saves values for controlled subject term vocabulary developed for the project.)
- HasType: (Assigns a type to a property)
- HasURN: (Field to save URN)
- HasVGWorld: (ID for VG Wort)
- HasVersionNumber: (An account of the resource.)

- Hide free text: (You can use this property to annotate a category. Set the value to 'true' in order to hide the free text input field in automatically created forms for that category.)
- ImageOrientation: (Saves orientation of an image (landscape, portrait))
- ImagePublishingStatus: (Describes different involved in publishing project images)
- ImageType: (Saves value describing the sub-categorization of the image category)
- IncludedWorkTitle: (Used for title of book sections, chapter or article-titles included in a book. See <http://en.wikipedia.org/wiki/Template:Citation/core>)
- IntDocVersion: (Internal Article Version)
- IntDocVersionLastUpdate: (Last Internal Article Version which was registered at Dara)
- Is uploadable: (Use this property to annotate a property if you want a file upload button to appear next to the form input field of the property in an automatically created Semantic Form.)
- IsForPublicUse: (Flags a resource ready to be published (Open Access).)
- IsGlossary: (Saves value if Encyclopedic Entry is also a Glossary Term.)
- IsInvisible: (Flags a resource as invisible.)
- IsMainRegionalSection: (Flags if Category is Main Regional Section)
- JiraKey: (Helper Property. Legacy values from JIRA import field 'Key'.)
- LastDaraUpdateDate: (Saves time of update of a resource to DA-RA)
- LastRequestDate: (Date when a publisher or author was contacted the last time.)
- LastRetrievalDate: (Date when an URL was retrieved the last time.)
- LastRetrieved: (Date when an image was retrieved from the source the last time.)
- LcshReady: (Flags if LCSH-subject classification is finished.)
- LicenseNote: (Editors' remarks to the license status.)
- LinkTo: (Universally defines other wiki-pages that are linked from the article text.)
- LinkedOrganization: (Marker for the linked encyclopedic entries of type "Organization")
- LinkedPerson: (Marker for the linked glossary entries of type "Person")
- MainName: (Saves main name in case a person has more than one given name.)
- MapLabel: (Used to describe the label of a marker on maps.)
- MappedAdditionalSubType: (Defines (mapped) specific sub-types for the main EntitySubTypes classes.)
- MergeBackIntoDev: (Flags wiki elements where parts or functions were altered in test or productive installations and must be manually merged back into development branch.)

- No automatic form edit: (Use this property to annotate a category if you do not want Semantic Forms to be created automatically for instances in this category.)
- NoBibliography: (Flags documents that do not include any bibliography entries (Selected Bibliography).)
- NoExternalLinks: (Flags documents that do not include any webpage descriptions (External Links).)
- NoTopicSet: (Tags documents where no ontology topics were set.)
- OntologyTermsAdded: (Saves value for publishing status when all ontology terms have been added to an article.)
- OrganizationType: (Saves value for sub-categorization of an organization, e.g. "Historical Party", "Project Partner" etc.)
- ParsedEventDate: (Saves parsed temporal values for single events that are described in a resource.)
- Part of bundle: (Defines a wiki element or content page as part of an installation bundle.)
- PartOfBibliography: (Saves flag if a bibliographic item should or should not show up in bibliography.)
- Periodical: (name of the periodical, journal, or magazine. This parameter controls whether or not the template will generate a citation in the style appropriate to a periodical, journal, magazine, or other serial publication.

If selected:

- PermissionDate: (Date when a publisher has given permission to publish a resource.)
- PersonType: (Saves value for sub-categorization of an organization, e.g. "Historical Party", "Project Partner" etc.)
- PersonsReady: (Flags if GND-Identifier for persons is set by BSB)
- ProjectArea: (Defines the project area a task is assigned to.)
- ProjectArticleType: (Saves category value for subcategories of the main project article format.)
- ProjectBugUrl: (Link to page where an error occurs.)
- ProjectDayEstimate: (Gives an estimate of man-days a task might take.)
- ProjectDocumentText: (Saves document text in a text property when no "freetext" input is used.)
- ProjectDocumentType: (Saves the main project document (publication) a project document belongs to.)
- ProjectDocumentationType: (Defines major type of the documentation e.g. "Developer" or "End-User". Deprecated use AdditionalSubType instead)
- ProjectEditor: (Field to save editor's name)
- ProjectItemCreationDate: (Time project item was filed.)
- ProjectItemPartOf: (Relates a project task to a project phase (Milestone))



- ProjectItemPartOfRelease: (Relates a project item to a planned release.)
- ProjectItemPartOfSprint: (Relates a project item to a sprint.)
- ProjectItemType: (Defines the type for a project item.)
- ProjectPersonRole: (Saves value for the role a project-related person plays in a project.)
- ProjectPersonType: (Deprecated. Use "AdditionalSubType" instead.)
- ProjectPublicationFormat: (Saves value for the display-template and the main project publication formats.)
- ProjectTask: (Short title for a task.)
- ProjectTaskAssignedTo: (The project member a task is assigned to.)
- ProjectTaskDescription: (Description of the project task.)
- ProjectTaskDue: (Due date of the project task.)
- ProjectTaskPriority: (Saves a string value to define the priority of a task or bug.)
- ProjectTaskSolution: (Possible solutions for problems defined by the task.)
- ProjectTaskStatus: (Describes status of completion for a task.)
- ProjectUserStoryPoints: (Number to estimate the complexity of a user story.)
- PropertyType: (Defines properties that are only temporarily used, e.g. during import.)
- PublicationPlace: (City of publication. Used in Wikipedia Citation template. <http://en.wikipedia.org/wiki/Template:Citation/core>)
- PublisherURL: (URL of the publishing institution)
- PublishingStatus: (Describes different stages involved in publishing project content.)
- RameauReady: (Flags if RAMEAU is done)
- Rationale: (Short description of a wiki element that is part of the installation.)
- Rdagr2DateOfBirth: (Date of birth of a person.)
- Rdagr2DateOfBirthText: (Date of birth of a person as text for unclear data)
- Rdagr2DateOfDeath: (Date of death of a person.)
- Rdagr2DateOfDeathText: (Date of death of a person as text for unclear data)
- Rdagr2PlaceOfBirth: (Place of birth of a person.)
- Rdagr2PlaceOfDeath: (Place of death of a person.)
- RdfsLabel: (May be used to provide a human-readable version of a resource's name.)
- ReferredByArticleName: (Relates an article or work to the appendix section of another article in the wiki.)
- RegionallsoCodeISO3166: (An account of the resource.)

- RelatedArticles: (Relates to other articles within any internal and external pages.)
- ResponsibleGeneralEditor: (Legacy value from JIRA import field 'Responsible General Editor'.)
- ResponsibleSectionEditors: (Legacy value from JIRA import field 'Responsible Section Editors'.)
- RightsCleared: (Saves value for publishing status when all legal issues related to an article are cleared.)
- RightsDescription: (Extended description of terms & conditions, license information etc.)
- RightsHolder: (Person holding usage rights.)
- RightsPending: (Saves value for publishing status if there is an ongoing request for rights clearance.)
- SchemaAffiliation: (An organization that this person is affiliated with. For example, a school/university, a club, or a team, see: <http://schema.org/affiliation>)
- SchemaHomeLocation: (A contact location for a person's residence, see: <http://schema.org/homeLocation>)
- SectionIncludesCategories: (Saves values for categories included in an appendix or resource sections. Needed for queries.)
- ShowAsShortcut: (True if Article label should show up in shortcut list.)
- ShowAsShortcutLabel: (Label for shortcut link.)
- SortBy: (Defines a string value from available properties used for sorting)
- SortByDate: (Defines a date value from available date properties used for sorting)
- SourceDescription: (Extended description of the source in the context of the publisher.)
- SourceFileUrl: (URL of the file embedded in a description page.)
- SourceTitle: (Describes title of a resource as mentioned in the source.)
- StartDate: (Start date of an event.)
- Stems from feed: (This property stores from which feed definition in the Wiki this feed stems.)
- SubjectIndexingComplete: (Saves value for publishing status when additional subject indexing is completed.)
- Surname1: (Last name of first author. Used in Wikipedia Citation template. <http://en.wikipedia.org/wiki/Template:Citation/core>)
- Surname2: (Last name of second author. Used in Wikipedia Citation template. <http://en.wikipedia.org/wiki/Template:Citation/core>)
- Surname3: (Last name of third author. Used in Wikipedia Citation template. <http://en.wikipedia.org/wiki/Template:Citation/core>)
- Surname4: (Last name of fourth author. Used in Wikipedia Citation template. <http://en.wikipedia.org/wiki/Template:Citation/core>)
- TechnicalComment: (Saves a technical explanation for a system article.)

- TextAdded: (Saves value for publishing status when article text was entered into a document.)
- TranslatorList: (An ordered list of translators similar to "BiboAuthorList")
- UpdateForReleaseNumber: (Saves number of a release in which a changed page should be included.)
- Use autogrow: (Use this property to annotate a property article if the property is of type Text and if you want the text area for this property in Automatic Semantic Forms to auto-grow instead of displaying a scrollbar.)
- Use class: (One can use this property to annotate a category or a property in order to tell the Automatic Semantic Forms Extension to use a specific CSS-class when creating a property's input field, respectively a categories table of input fields.)
- Use default input: (Use this property to annotate a category article if you want to customize which standard input fields should be shown in the form footer. If one of the categories of an instance has such an annotation, only the chosen default input fields are shown. Otherwise all are shown.)
- Use display template: (The Automatic Semantic Forms Extension creates silent annotations, i.e. annotations that are only displayed in the fact box and not in the rendered article. Use the Use display template to assign a template to a category so that the data which is entered for that category is displayed via a template.)
- Use feed item prefix: (The value of this property will be used as a prefix for titles of imported feed items.)
- Use input type: (Use this property to annotate a property article if you want to choose a different input type for that property in automatically created Semantic Forms.)
- Use page name template: (Use this property to provide your own custom page name template for the one-steep creation process. The value of this property is inherited from direct super categories if the category does not provide a value itself and if it does not set the value to false.)
- Use preload article: (Use this property in categories to automatically preload new instances with the given article. The value of this property is inherited from direct super categories if the category does not provide a value itself and if it does not set the value to false.)
- Validator: (Use the Validator property to annotate a property, if you want to validate a user's input into the property's form input field in an automatically created Semantic form with a regular expression.)
- ViafIdentifier: (Identifier from the Virtual International Authority File VIAF)
- VolumeTitle: (Saves title of journals or anthologies.)
- Was Added During Term Import: (Relates a wiki article to term import run, in which the article was generated. This property is set by the system and should not be changed by users.)
- Was Updated During Term Import: (Relates a wiki article to term import run, in which the article was generated. This property is set by the system and should not be changed by users.)

- WikipediaEnArticle: (The "Has type" property is a special property in Semantic MediaWiki with a built-in meaning: it assigns a datatype to a property.)
- Form:WikipediaEnArticle: (This is the page of Special property Imported from. It lists all pages using this special property. See [1] for further information.)
- ZoteroItemId: (Saves ID for imported Zotero bibliography items. Needed to identify duplicates.)

### 2.3. Dynamic Default Values for Properties

Every article in the "1914-1918-online" encyclopedia includes an appendix section which lists related content. The relation between the article and the related content is based on page properties that describe various kinds of semantic relations. One relation, for instance, is based on the fact, that some informative materials like an image or a description of a web-resource (ExternalLink) should be definitely shown with the article. Another property, for example, describes that the article shares the same subject classification with other articles.

To ease the process of uploading and saving new materials for the appendix section of an article, the wiki saves default values for several classification properties that are inherited from the "main" article. The values for these properties are generated based on the assumption that any "material" (image or web-description) that belongs to an article will also automatically share its subject classification. For example in the page source of Template:Work, the values of the property Property:HasMainThematicSections are dynamically set by a semantic query.

```

{{{#if:{{{HasMainThematicSections}}}}
.....{{#arraymap:{{{HasMainThematicSections}}};|*[[HasMainThematicSections::*| ]]] }}
.....{{#arraymap:{{{#arraymap:{{{ReferredByArticleName}}};|*{{#show:*?
HasMainThematicSections|sep=;}}};|*[[HasMainThematicSections::*| ]]] }}
}}

```

Since every image or every description of a web-resource belongs to an article it instantly inherits the subject classification from the article. If the classification of the image is changed manually the value overrides the dynamic classification.

Be aware that any changes to "dynamic" properties on the side of the article will only show up as semantic properties of the image or description of web-resource with a delay. This is due to the fact that the semantic database has to keep track with changes that are not immediately applied to an article by a background process that is usually started from time to time. Another way to test if changed properties are inherited correctly is to make a little change to the article of the image or web-resource description. This will refresh the database immediately.

## 2.4. Workflow

### 2.4.1. General Workflow

#### 2.4.1.1. Preparation

Documents by a large number of users, who are sending their scientific material to the editorial office, are usually not formatted in an accurate way. Therefore, these documents first need to get revised and if applicable confirmed. After that, a document has to be converted to a standardized format.

#### 2.4.1.2. Bibliography

There are many ways in which bibliographical data can be integrated into a scientific web project. As we are using the open source platform Zotero, we first need to add the selected bibliography to the Zotero group, which is linked to our project. After that, an API especially for Zotero import allows us to integrate the bibliography to a certain article.

#### 2.4.1.3. Compiling an Article and Metadata

There are two different forms for article types, which are Encyclopedic Entries and Handbook Articles. Additionally, internal as well as external links have to be inserted. Other metadata that might be added are images, locations and events.

### 2.4.2. Additional Workflow

After deciding what kind of article you want to upload - Handbook Article or Encyclopedic Entry – you may navigate to the respective form suitable for this article. After filling out the form and adding the text to the field, you can also add metadata like images, locations, external and internal links etc.

#### 2.4.2.1. Adding Geodata

There are two ways of adding geodata to the encyclopaedias' articles: editing geodata inside an article or external adding. The form for editing an article also contains a tab that is labelled "Additional Media, External Links & Geodata". By clicking on the "Geodata form" the user will be directed to a project form within which you can add/edit geodata to this related article. This form contains properties like "Wiki Article Name" (that is the Namespace of the URL), "DcTermsTitle" (that is the Title/Header of the Geo-page), "Description", "Related article(s)" (the user can add more related articles here) and the coordinates. Moreover, the user interface of the SMW provides an external access for adding/editing geodata. But still the user is directed to the same form, where the above-named properties can be filled out.

### 2.4.2.2. Adding Related Events

Within the workflow of this encyclopedia, Timeline Events have been added via CSV-import. We created a special page for importing CSV documents (Special:ImportCSV). Furthermore, there is also a form for adding/editing Events by hand. Relevant properties to be filled out are: "Wiki Article Name", "Start Date", "End Date", "Related Events" and "Related Parent Events". Additionally, the user can also link an event to certain geodata.

## 2.5. Generate DOI

An important module is to generate a DOI for each article. Currently it is a semi-automatic workflow in three steps.

### 2.5.1. Important details

- Bayerische Staatsbibliothek and GESIS signed a user contract
- To generate DOIs it is mandatory to know BSB's Login-Data for GESIS DOI-web portal
- Every Handbook / Encyclopedic Entry in 1914-1918-online already has a unique DOI stored in the Property HasDoi; after an article is published this DOI has to be announced on the GESIS DOI-web portal
- DOI-schema is: 10.5072/ie1418.XXXXX

### 2.5.2. Workflow

- Generate/Download csv-metadata from SMW
- use PHP-Script to convert CSV to XML
- upload XML to GESIS

#### 2.5.2.1. Conversion into XML

This is the code to generate XML-Files (each per article / row in csv)

```
<?php
$inputFilename = 'doi-1909-ilker-1.csv';
// Open csv to read
$inputFile = fopen($inputFilename, 'r');
// Get the headers of the file
$headers = fgetcsv($inputFile, 0, ",");
$count=0;
try{
// Loop through each row creating a <resource> node with the correct data
while (($row = fgetcsv($inputFile,0, ",")) !== FALSE)
{
// Create a new dom document with pretty formatting
$xmlDom = new DomDocument("1.0", "UTF-8");
$xmlDom->formatOutput = true;
$resource = $xmlDom->createElement('resource');
```

```

$resourceAttribute = $xmlDom->createAttribute('xmlns:xsi');
$resourceAttribute->value="http://www.w3.org/2001/XMLSchema-instance";
$resource->appendChild($resourceAttribute);
$xmlDom->appendChild($resource);
//Metadata-Typ = Text ==3
$resourceType = $xmlDom->createElement("resourceType", "3");
$resourceType = $resource->appendChild($resourceType);
$outerResourceTypesFree=$xmlDom->createElement("resourceTypesFree");
$innerResourceTypesFree=$xmlDom->createElement("resourceTypeFree");
$language=$xmlDom->createElement("language", "en");
$outerResourceTypesFree=$resource->appendChild($outerResourceTypesFree);
$innerResourceTypesFree=$outerResourceTypesFree-
>appendChild($innerResourceTypesFree);
$language=$innerResourceTypesFree->appendChild($language);
//append main XML-Tags
$resourceIdentifier=$xmlDom->createElement('resourceIdentifier');
$titles=$xmlDom->createElement('titles');
$creators=$xmlDom->createElement('creators');
$dataURLs=$xmlDom->createElement('dataURLs');
$doiProposal=$xmlDom->createElement('doiProposal');
$publicationDate=$xmlDom->createElement('publicationDate');
$glPlace=$xmlDom->createElement('glPlace', "Berlin");
$availability=$xmlDom->createElement('availability');
$rights=$xmlDom->createElement("rights");
$resourceLanguage=$xmlDom->createElement('resourceLanguage', "eng");
$classifications=$xmlDom->createElement('classifications');
$freeKeywords=$xmlDom->createElement('freeKeywords');
$descriptions=$xmlDom->createElement('descriptions');
$geographicCoverages=$xmlDom->createElement('geographicCoverages');
$temporalCoverages=$xmlDom->createElement('temporalCoverages');
$contributors=$xmlDom->createElement('contributors');
//XML-Tags unter Resource einhängen
$titles = $resource->appendChild($titles);
$creators = $resource->appendChild($creators);
$dataURLs = $resource->appendChild($dataURLs);
$doiProposal = $resource->appendChild($doiProposal);
$publicationDate = $resource->appendChild($publicationDate);
$glPlace = $resource->appendChild($glPlace);
$availability = $resource->appendChild($availability);
$rights=$resource->appendChild($rights);
$resourceLanguage = $resource->appendChild($resourceLanguage);
$classifications = $resource->appendChild($classifications);
$freeKeywords = $resource->appendChild($freeKeywords);
$descriptions = $resource->appendChild($descriptions);
$geographicCoverages = $resource->appendChild($geographicCoverages);
$temporalCoverages = $resource->appendChild($temporalCoverages);
$contributors = $resource->appendChild($contributors);

//echo $xmlDom->saveXML();
$count++;
foreach ($headers as $i => $header)
{

```

```

$row[$i]=trim($row[$i]);
//echo $header."<br>";
switch ($header){
  case "HasDoi":
    $identifier = $xmlDom->createElement("identifier", $row[$i]);
    $currentVersion = $xmlDom->createElement("currentVersion", "1");
    $identifier=$resourceIdentifier->appendChild($identifier);
    $currentVersion=$resourceIdentifier->appendChild($currentVersion);
    $doi=$xmlDom->createTextNode("10.5072/ie1418".$row[$i]);
    $doi=$doiProposal->appendChild($doi);
    break;
  case "HasDoiVersion":
    $currentVersion = $xmlDom->createElement("currentVersion", $row[$i]);
    $currentVersion=$resourceIdentifier->appendChild($currentVersion);
    break;
  case "DctermsTitle":
    echo $row[$i]."<br>";
    //in typeName einhängen
    $typeName = $xmlDom->createElement("typeName", $row[$i]);
    $typeName=$innerResourceTypesFree->appendChild($typeName);
    //in Titles einhängen
    $title=$xmlDom->createElement("title");
    $titleName=$xmlDom->createElement("titleName", $row[$i]);
    $language2=$xmlDom->createElement("language", "en");
    $title=$titles->appendChild($title);
    $language2=$title->appendChild($language2);
    $titleName=$title->appendChild($titleName);
    break;
  case "HasSummary":
    if($row[$i]){
      $description = $xmlDom->createElement("description");
      $languageDescription = $xmlDom->createElement("language", "en");
      $descriptionFreetext = $xmlDom->createElement("freetext", $row[$i]);
      //1 bedeutet Artikelzusammenfassung
      $type=$xmlDom->createElement("type", "1");
      $description=$descriptions->appendChild($description);
      $languageDescription=$description->appendChild($languageDescription);
      $descriptionFreetext=$description->appendChild($descriptionFreetext);
      $type=$description->appendChild($type);
    }
    break;
  case "Author1Firstname":
    $creator1=$xmlDom->createElement("creator");
    $person1=$xmlDom->createElement("person");
    $firstName1=$xmlDom->createElement("firstName", $row[$i]);
    $lastName1=$xmlDom->createElement("lastName", $row[$i+1]);
    $creator1=$creators->appendChild($creator1);
    $person1=$creator1->appendChild($person1);
    $firstName1=$person1->appendChild($firstName1);
    $lastName1=$person1->appendChild($lastName1);
    break;
  case "Author2Firstname":

```



```

if($row[$i]){
    //$creators2=$xmlDom->createElement("creators");
    $creator2=$xmlDom->createElement("creator");
    $person2=$xmlDom->createElement("person");
    $firstName2=$xmlDom->createElement("firstName", $row[$i]);
    $lastName2=$xmlDom->createElement("lastName", $row[$i+1]);
    //$creators2=$resource->appendChild($creators2);
    $creator2=$creators->appendChild($creator2);
    $person2=$creator2->appendChild($person2);
    $firstName2=$person2->appendChild($firstName2);
    $lastName2=$person2->appendChild($lastName2);
}
break;
case "Author1Affiliation":
if($row[$i]){
    $affiliation1=$xmlDom->createElement("affiliation");
    $affiliationName1=$xmlDom->createElement("affiliationName");
    $language3=$xmlDom->createElement("language", "en");
    $name1=$xmlDom->createElement("name", $row[$i]);
    $affiliation1=$person1->appendChild($affiliation1);
    $affiliationName1=$affiliation1->appendChild($affiliationName1);
    $language3=$affiliationName1->appendChild($language3);
    $name1=$affiliationName1->appendChild($name1);
}
break;
case "URL":
    $dataURL=$xmlDom->createElement("dataURL", $row[$i]);
    $dataURL=$dataURLs->appendChild($dataURL);
    break;
//Hier Publikationsdatum aus CSV und zugleich automatisch ORT=Berlin
case "DctermsIssued":
    //$date=$xmlDom->createElement("date", $row[$i]);
    //HIER FIX OKTOBER 2014 eintragen MUSS DANN WIEDER GEÄNDERT
WERDEN NACH DEM 8.10! <monthyear> <date>
    $date=$xmlDom->createElement("monthyear", "2014-10");
    $date=$publicationDate->appendChild($date);
    break;
//Verfügbarkeit Frei=1 Mit DcTermsLicenceText aus CSV Rights=Text aus
AvailabilityText
case "DctermsLicense":
if(!$row[$i]){
$row[$i]="http://creativecommons.org/licenses/by-nc-nd/3.0/de/deed.en_GB";
}
    $availabilityControlled=$xmlDom->createElement("availabilityControlled", "1");
    $availabilityFreeEN=$xmlDom->createElement("availabilityFree");
    $language4=$xmlDom->createElement("language", "en");
    $availabilityText=$xmlDom->createElement("availabilityText", $row[$i]. " -
Metadata is licensed under Creative Commons Licence CC0
(http://creativecommons.org/publicdomain/zero/1.0/legalcode CC0 1.0 Universal) " );
    //Deutsch
    $availabilityFreeDE=$xmlDom->createElement("availabilityFree");

```

```

        $textLicenceDE="Text ist lizenziert unter Creative Commons Licence
(http://creativecommons.org/licenses/by-nc-nd/3.0/de/deed.en_GB Attribution-
NonCommercial-NoDerivs 3.0 Germany-CC BY-NC-ND 3.0) - Metadaten unter Creative
Commons Licence CC0 (http://creativecommons.org/publicdomain/zero/1.0/legalcode CC0
1.0 Universal)";
        $languageDE=$xmlDom->createElement("language", "de");
        $availabilityTextDE=$xmlDom-
>createElement("availabilityText",$textLicenceDE);
        $availabilityControlled=$availability->appendChild($availabilityControlled);
        $availabilityFreeEN=$availability->appendChild($availabilityFreeEN);
        $language4=$availabilityFreeEN->appendChild($language4);
        $availabilityText=$availabilityFreeEN->appendChild($availabilityText);
        $availabilityFreeDE=$availability->appendChild($availabilityFreeDE);
        $languageDE=$availabilityFreeDE->appendChild($languageDE);
        $availabilityTextDE=$availabilityFreeDE->appendChild($availabilityTextDE);
        break;
    case "DctermsRights":
        if($row[$i]){
            $right=$xmlDom->createElement("right");
            $language5=$xmlDom->createElement("language", "en");
            $rightsText=$xmlDom->createElement("rightsText", $row[$i]);
            $right=$rights->appendChild($right);
            $language5=$right->appendChild($language5);
            $rightsText=$right->appendChild($rightsText);
        }
        break;
    case "HasGndIdentifier":
        if($row[$i]){
            $classificationGND=$xmlDom->createElement('classification');
            $classificationExternalGND=$xmlDom-
>createElement("classificationExternal");
            $languageGND=$xmlDom->createElement("language", "de");
            $schemaGND=$xmlDom->createElement("schema", "Gemeinsame Normdatei
(GND)");
            $termsGND=$xmlDom->createElement("terms");
            //HasGndIdentifier am Semicolon trennen
            $GndKeywords=explode(";", $row[$i]);
            foreach ($GndKeywords as $j => $myWord)
            {
                $termGND=$xmlDom->createElement("term", trim($GndKeywords[$j]));
                $termGND=$termsGND->appendChild($termGND);
            }
            $classificationGND=$classifications->appendChild($classificationGND);
            $classificationExternalGND=$classificationGND-
>appendChild($classificationExternalGND);
            $languageGND=$classificationExternalGND->appendChild($languageGND);
            $schemaGND=$classificationExternalGND->appendChild($schemaGND);
            $termsGND=$classificationExternalGND->appendChild($termsGND);
        }
        break;
    case "HasLcshIdentifier":
        if($row[$i]){

```

```

        $classificationLCSH=$xmlDom->createElement('classification');
        $classificationExternalLCSH=$xmlDom-
>createElement("classificationExternal");
        $languageLCSH=$xmlDom->createElement("language", "en");
        $schemaLCSH=$xmlDom->createElement("schema", "Library of Congress
Subject Headings (LCSH)");
        $termsLCSH=$xmlDom->createElement("terms");
        //am Semicolon trennen
        $Lcskeywords=explode(";", $row[$i]);
        foreach ($Lcskeywords as $j => $myWord)
        {
            $termLCSH=$xmlDom->createElement("term", trim($Lcskeywords[$j]));
            $termLCSH=$termsLCSH->appendChild($termLCSH);
        }
        $classificationLCSH=$classifications->appendChild($classificationLCSH);
        $classificationExternalLCSH=$classificationLCSH-
>appendChild($classificationExternalLCSH);
        $languageLCSH=$classificationExternalLCSH-
>appendChild($languageLCSH);
        $schemaLCSH=$classificationExternalLCSH->appendChild($schemaLCSH);
        $termsLCSH=$classificationExternalLCSH->appendChild($termsLCSH);
    }
    break;
    /*case "HasRameauIdentifier":
        if($row[$i]){
            $classificationExternalRAMEAU=$xmlDom-
>createElement("classificationExternal");
            $languageRAMEAU=$xmlDom->createElement("language", "fr");
            $schemaRAMEAU=$xmlDom->createElement("schema", "Répertoire
d'autorité-matière encyclopédique et alphabétique unifié (Rameau)");
            $termsRAMEAU=$xmlDom->createElement("terms");
            //am Semicolon trennen
            $myKeywords=explode(";", $row[$i]);
            foreach ($myKeywords as $j => $myWord)
            {
                $termRAMEAU=$xmlDom->createElement("term", trim($myKeywords[$j]));
                $termRAMEAU=$termsRAMEAU->appendChild($termRAMEAU);
            }
            $classificationExternalRAMEAU=$classification-
>appendChild($classificationExternalRAMEAU);
            $languageRAMEAU=$classificationExternalRAMEAU-
>appendChild($languageRAMEAU);
            $schemaRAMEAU=$classificationExternalRAMEAU-
>appendChild($schemaRAMEAU);
            $termsRAMEAU=$classificationExternalRAMEAU-
>appendChild($termsRAMEAU);
        }
        break;*/
    case "HasTopics":
        if($row[$i]){
            $freeKeyword=$xmlDom->createElement("freeKeyword");
            $languageTopics=$xmlDom->createElement("language", "en");

```

```

        $languageTopics=$freeKeyword->appendChild($languageTopics);
        $keywords=$xmlDom->createElement("keywords");
        //HASTOPICS am Strichpunkt trennen
        $myKeywords=explode(";", $row[$i]);
        foreach ($myKeywords as $j => $myWord)
        {
            $keywordTopics=$xmlDom->createElement("keyword",
trim($myKeywords[$j]));
            $keywordTopics=$keywords->appendChild($keywordTopics);
        }
        $freeKeyword=$freeKeywords->appendChild($freeKeyword);
        $keywords=$freeKeyword->appendChild($keywords);
    }
    break;
case "HasKeywords":
    if($row[$i]){
        if($keywords)
        {
            //HasKeywords am Strichpunkt trennen
            $myKeywords=explode(";", $row[$i]);
            foreach ($myKeywords as $j => $myWord)
            {
                $keywordKW=$xmlDom->createElement("keyword",
trim($myKeywords[$j]));
                $keywordKW=$keywords->appendChild($keywordKW);
            }
        }
        else {
            $freeKeyword=$xmlDom->createElement("freeKeyword");
            $languageKW=$xmlDom->createElement("language", "en");
            $languageKW=$freeKeyword->appendChild($languageKW);
            $keywords=$xmlDom->createElement("keywords");
            //HASKEYWORDS am Strichpunkt trennen
            $myKeywords=explode(";", $row[$i]);
            foreach ($myKeywords as $j => $myWord)
            {
                $keywordKW=$xmlDom-
>createElement("keyword", trim($myKeywords[$j]));
                $keywordKW=$keywords-
>appendChild($keywordKW);
            }
            $freeKeyword=$freeKeywords->appendChild($freeKeyword);
            $keywords=$freeKeyword->appendChild($keywords);
        }
    }

    break;
case "DdcGeographicAreaCode":
    $geographicCoverage=$xmlDom->createElement("geographicCoverage");
    $languageGeo=$xmlDom->createElement("language", "en");
    $freetextGeo=$xmlDom->createElement("freetext", $row[$i]);

```

```

        $geographicCoverage=$geographicCoverages-
>appendChild($geographicCoverage);
        $languageGeo=$geographicCoverage->appendChild($languageGeo);
        $freetextGeo=$geographicCoverage->appendChild($freetextGeo);
        break;
    }
}
//TemporalCoverages
$temporalCoverage=$xmlDom->createElement("temporalCoverage");
$languageTime=$xmlDom->createElement("language", "en");
$temporalCoverageFormal=$xmlDom->createElement("temporalCoverageFormal");
$startDate=$xmlDom->createElement("startDate");
$startYear=$xmlDom->createElement("year", "1914");
$endDate=$xmlDom->createElement("endDate");
$endYear=$xmlDom->createElement("year", "1918");
$temporalCoverage=$temporalCoverages->appendChild($temporalCoverage);
$languageTime=$temporalCoverage->appendChild($languageTime);
$temporalCoverageFormal=$temporalCoverage->appendChild($temporalCoverageFormal);
$startDate=$temporalCoverageFormal->appendChild($startDate);
$startYear=$startDate->appendChild($startYear);
$endDate=$temporalCoverageFormal->appendChild($endDate);
$endYear=$endDate->appendChild($endYear);
//Noch die Herausgeber ans Ende anfügen ID=5==Herausgeber!
    $mycontributors = array
    (
        array ("Ute", "Daniel", "5"),
        array ("Peter", "Gatrell", "5"),
        array ("Oliver", "Janz", "5"),
        array ("Heather", "Jones", "5"),
        array ("Jennifer", "Keene", "5"),
        array ("Alan", "Kramer", "5"),
        array ("Bill", "Nasson", "5"),
        array ("Nicolas", "Apostolopoulos", "9"),
        array ("Oliver", "Janz", "9"),
        array ("Gregor", "Horstkemper", "9")
    );
    foreach ($mycontributors as $k => $mycontributor)
    {
        $contributor=$xmlDom->createElement("contributor");
        $contributorPerson=$xmlDom->createElement("person");
        $contributorFirstName=$xmlDom->createElement("firstName", $mycontributor[0]);
        $contributorLastName=$xmlDom->createElement("lastName", $mycontributor[1]);
        $contributorType=$xmlDom->createElement("contributorType", $mycontributor[2]);
        $contributor=$contributors->appendChild($contributor);
        $contributorPerson=$contributor->appendChild($contributorPerson);
        $contributorFirstName=$contributorPerson->appendChild($contributorFirstName);
        $contributorLastName=$contributorPerson->appendChild($contributorLastName);
        $contributorType=$contributorPerson->appendChild($contributorType);
    }
$outputFile = fopen("xml/output_". $count. ".xml", 'w');
    fwrite ($outputFile, $xmlDom->saveXML());
    fclose($outputFile);

```

```

}
}
catch (Exception $e)
    {echo "fehler: ". $e->getMessage();}
//echo $xmlDom->saveXML();
?>

```

## 2.6. Data Structure

The data structure of the “1914-1918-online” encyclopedia is based on a semantic syntax that can be compared to projects that use Semantic MediaWiki software. Its content is divided into three stages of types that are interlinked, namely EntityType, EntitySubType and AdditionalSubType.

### 2.6.1. Metadata

Metadata provides additional information on each article, which may not be of obvious use for the user but may be essential for some encyclopedia features such a visibility in library and other catalogues, linking to related articles and such. Inside this encyclopedia there is a huge convolute of metadata composing background information for each article or entities that is indispensable for creating a complete structured encyclopedia. Some examples of metadata relevant to this encyclopedia are author keywords, keydates, geographic coordinates, bibliographic details etc. In addition, each article is assigned to a certain taxonomical rank and a specific ontology which are both based on the decisions of the editorial office. With the help of this kind of classification and assignments, articles can be found more easily and are interlinked with each other in a meaningful way within this online encyclopedia. Therefore, a whole network of articles, persons, themes and regions can be generated that shows the user amongst others the globality of the Great War. Furthermore, Metadata also enables the user to find relevant information using the search function, e.g. if you search for the person Carl von Clausewitz, you will get all articles this person is assigned to. The same works for all other metadata that exists within this encyclopedia.

#### 2.6.1.1. EntityType

Saves value for the main entity type described by the wiki-article. Equals the main template-name and the category-name of the article. Properties and templates are described as "Wikipage".

There are some important templates that are assigned to Classifications that are part of the encyclopedia: e.g. Image, Person, Organization, Concept, Space, Work, Code, ProjectDocument and ProjectItem.

#### 2.6.1.2. EntitySubType

Defines sub-type and category value for the main content classes (ProjectArticle,ProjectPage, ProjectTask, ProjectPerson,BibliographyEntry etc.).

### 2.6.1.3. AdditionalSubType

Defines specific sub-types for the main content classes, e.g. Thematic Survey Article, MainPage, ProjectAuthor, ProjectEditor. This classification type can be linked to Article Types, Authors, Editors etc.

## 2.6.2. Keyword Vocabulary

### 2.6.2.1. GND subject headings

GND (Gemeinsame Norm Datei) is a controlled keyword vocabulary that is used in all research libraries in Germany. These keywords guarantee correct intellectual semantic indexing and therefore enable you to find relevant literature about the same topics in various German-speaking libraries. We provide links to the most important library-catalogues on every article of this encyclopedia. Searching with GND keywords is the most reliable search method, because it unites synonyms in one expression (e.g. "First World War" and "Great War") and includes books that have metaphoric titles such as "In Europa gehen die Lichter aus" ("The Lamps went out in Europe") or "All Quiet on the Western Front". At the same time, the use of controlled vocabularies facilitates automatic or manual cross-linking from other online resources to relevant articles in this encyclopedia.

### 2.6.2.2. Library of Congress

LC Subject Headings (Library of Congress Subject Headings, or LCSH) is a controlled keyword vocabulary for English speaking libraries that is also applied internationally. Library of Congress Subject Headings have a different structure than the German GND, often combining fixed blocks of semantically linked subject headings such as "World War, 1914-1918--Great Britain" or "World War, 1914-1918--Medical care--France". Searching with LCSH has the same advantages as described above for GND keywords, uniting synonyms and including metaphoric titles. We also provide links to the most important English speaking and international library-catalogues using LCSH (e.g. Library of Congress or WorldCat, which unites catalogues of various libraries containing approximately 2 billion bibliographical entries in 470 languages).

### 2.6.2.3. Rameau subject

Rameau (Répertoire d'autorité-matière encyclopédique et alphabétique unifié) is a controlled keyword vocabulary for French speaking libraries that is also applied in the francophone world. The structure of Rameau keywords is very similar to that of LCSH and features the same advantages as described above. We provide links to the BnF (Bibliothèque nationale de France) and use Rameau in addition to GND and LCSH in our automated searches in catalogues such as WorldCat in order to supply you with comprehensive international literature on the topic you have been reading about in our encyclopedia. Additionally, we guide you to the actual catalogues so that you can not only view the bibliographical information,

but also actually access the books, either electronically - if available - or order them directly at your nearest library.

## 2.7. Basic Interfaces & Extensions

### 2.7.1. Zotero

Zotero is a free and open source reference management software and it was possible to build an API in order to import all data that is collected for this encyclopedia automatically by using time stamps etc. Zotero offers the possibility to use and cumulate bibliographical data in a collaborative way, which is why this software is an advantage for a decentralized project like 1914-1918-online with a large number of participants.

### 2.7.2. Geodata

One of the main extensions within the encyclopedia is the implementation of a form for saving and editing geodata that is related to one or more article(s). By using this form an admin user can generate a new page with geographical information and articles that relate to this "page" - which is assigned to the template "location". This geographical vantage point opens new navigational routes through the encyclopedia and introduces you to articles you may otherwise not have come across. Within the encyclopedia, the locations are not only connected to articles, but there also might be interlinkages to Timeline Events.

### 2.7.3. Timeline Events

There are 190 Timeline Events that are connected to articles or other metadata within the encyclopedia. It is important to connect event data to a specific article and not the other way around. Data for the Timeline Events may be imported via csv import or manually. It is possible to create some kind of sub-events that are linked directly to major events. As we are using SMW, it is also possible to establish interlinkages to other metadata, such as geo-coordinates. Within the backend, a special semantic form has been created in order to make it possible for admin users to fill in Timeline Events by hand. But it is also possible to batch upload a list of events via the special template upload CSV.

### 2.7.4. PDF Print Preview

A basic extension but important development has been the development of a pdf print preview for article.



### 3. User Documentation

General information on the encyclopedia structure and background may be found on our website at:

[http://www.1914-1918-online.net/03\\_encyclopedia/index.html](http://www.1914-1918-online.net/03_encyclopedia/index.html)

as well as in our *Author Guidelines*:

[http://www.1914-1918-online.net/05\\_information\\_for\\_authors/01\\_Project\\_Information\\_2015\\_Authors.pdf](http://www.1914-1918-online.net/05_information_for_authors/01_Project_Information_2015_Authors.pdf)

More detailed information for readers may be found in the encyclopedia introduction:

<http://encyclopedia.1914-1918-online.net/project/about/>

and the FAQ section of the encyclopedia:

<http://encyclopedia.1914-1918-online.net/project/faq/> .