

Active Documents for Distributed Document-Intense Collaborations in the Semantic Web



Christine Müller (c.mueller@jacobs-university.de)
Department of Computer Science, University of Auckland, NZ
Computer Science Department, Jacobs University Bremen, Germany

Active Documents are *interactive sessions* with an *environment*, which provides *views* on an collection of documents and supports *queries* to configure (or personalise) these perspectives.

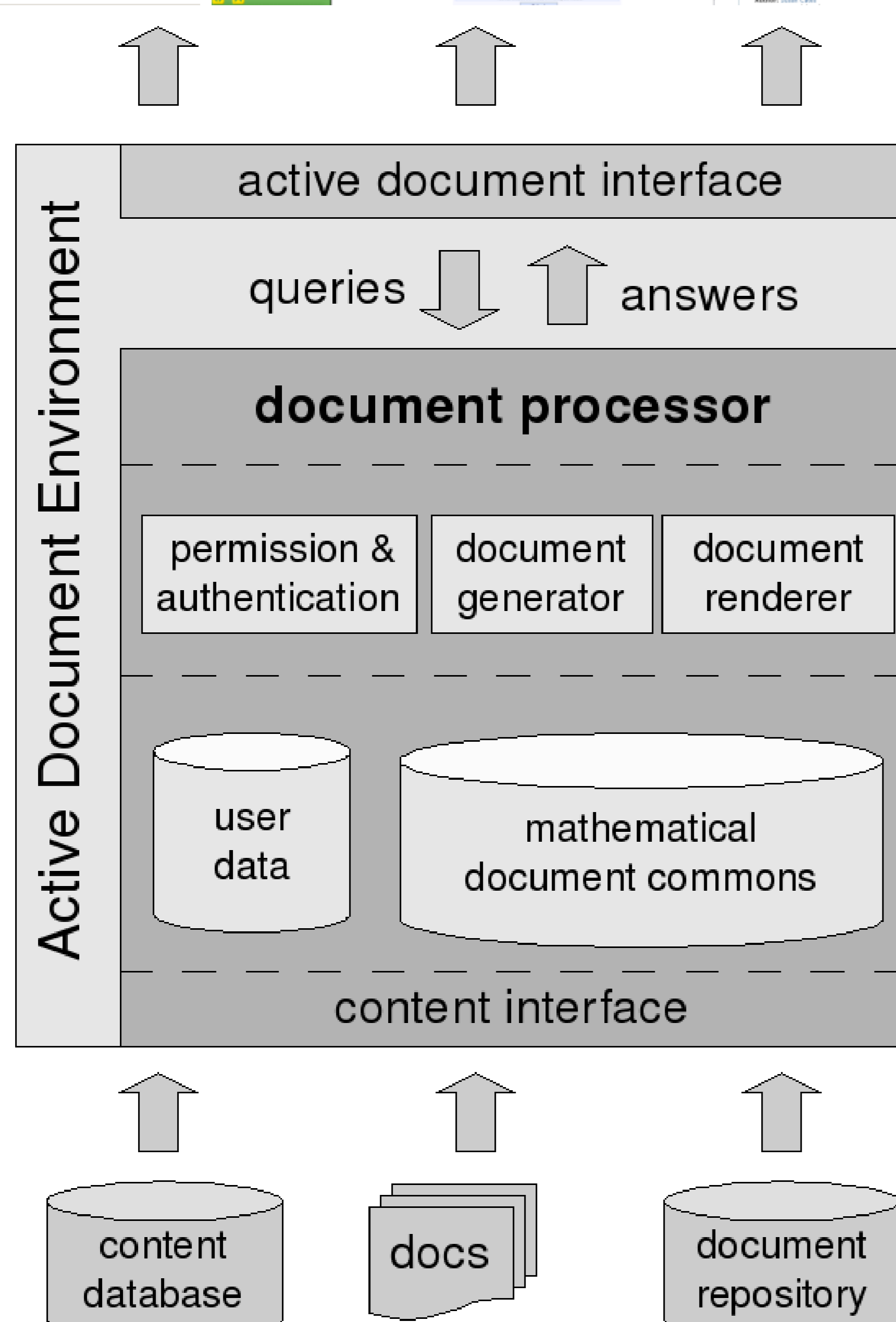
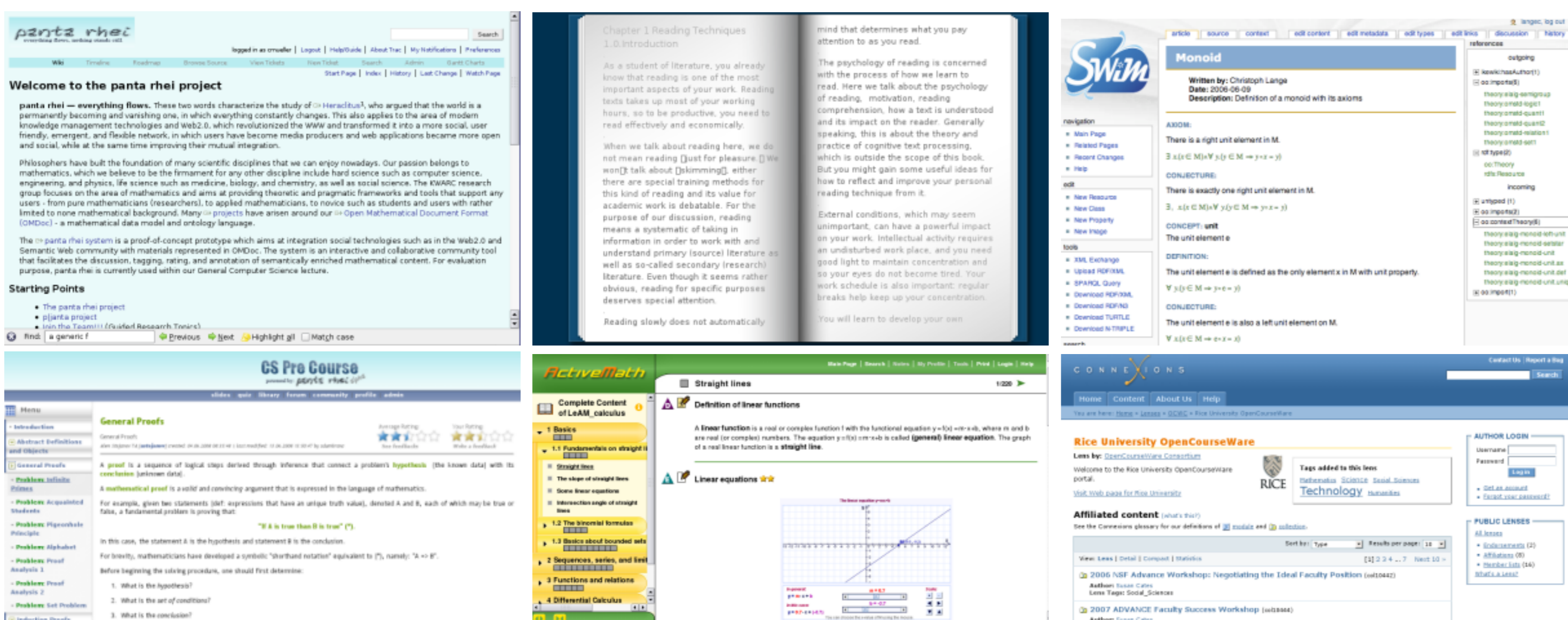
Case Study on Mathematics

- **Convenience:** Mathematical knowledge is *highly structured* and *precise*.
- **Extensibility:** The mathematical language is the foundation for several technical disciplines (e.g. software engineering)
- **Applicability:** Most mathematical results are encoded into documents. Mathematical research is a document-intensive collaborative activity.
- **Problem:** Most mathematical documents are *static* with a fixed level of abstraction, level of detail, or notation. There are no means to *query for* details or adapt the presentation.

Services of Active Documents

1. configurable *layouts* and *output formats*,
2. presentations with varying *level of detail* or *formality* (proportion of natural language and formulae),
3. *multilingual* presentations,
4. consistent use of mathematical *notations*,
5. *enrichment* of the documents with examples, definitions, etc.
6. *clarification* of new aspects and main ideas.

An Active Document Environment

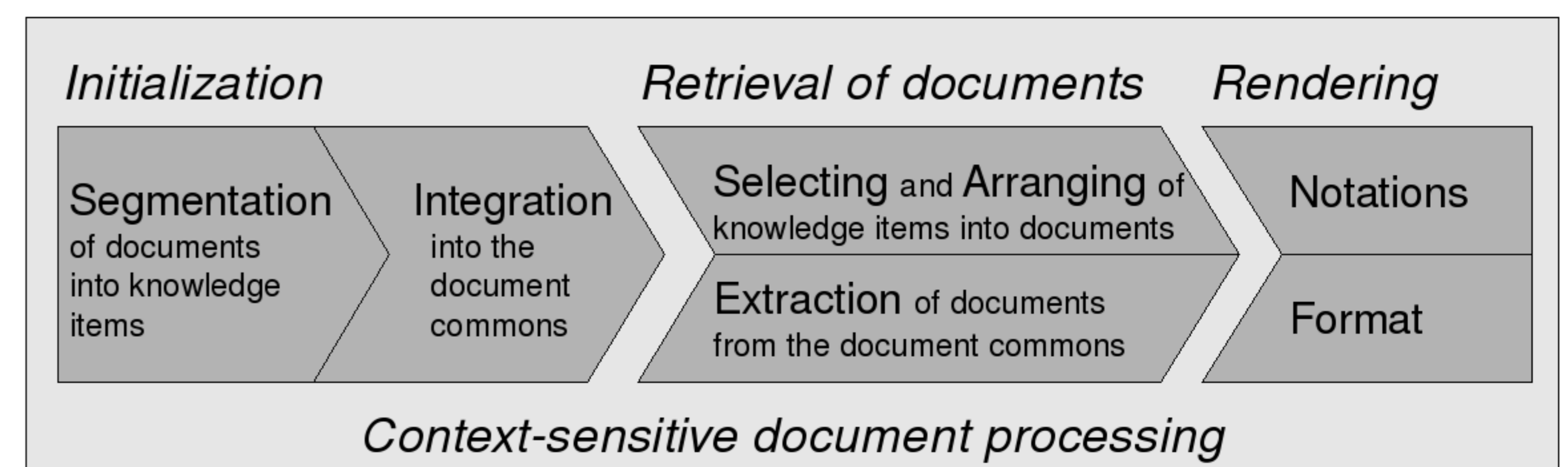


For details see <https://trac.kwarc.info/panta-rhei/wiki/janta>.

Document Processing Workflow

1. **Initialisation** of the Document Commons
 - (a) Retrieving document from a backend, e.g. file system
 - (b) Segmentation of document into *reusable, identifiable* knowledge items, preserves the original document structure
 - (c) Integration of items and initial structure into the document commons
2. **Retrieving the document** from the document commons
 - Extraction initial documents from the document commons
 - Creating new documents: Arranging and selecting items in various document layouts based on strategies or templates, e.g. a guided tour, a web page, a book, a slide presentation
3. **Rendering the document:** Setting the corresponding mathematical notations, converting the document into a suitable format

Machine-processable representation of documents	Representation of structures & strategies	Distinction between content and form
---	---	--------------------------------------



Representation of context annotations

Optional: Requests for document can provide **context parameters** that guide the creation and rendering of documents.

- Selecting preferable variants for the document, e.g. based on language
- Adapting to system settings, e.g. based on the font, font-size, line-height, and margins for the document display

Implementation Details

- XML documents in the *Open Mathematical Document Format*
- *Document Commons*: RDF Graph of interlinked knowledge items, specified by the OMDOC *schema* and its *ontologies*.
- Document Processor: RESTful web service, Java Servlet
- Rendering: XSLT transformation, e.g. OMDOC \Rightarrow XHTML, context-sensitive conversion from OPENMATH \Rightarrow MATHML
- Interface: PHP using CodeIgniter Framework

Trac: <https://trac.kwarc.info/panta-rhei>

Case Study in Lecture: <http://panta-rhei.kwarc.info>

We thank Josip Dzlonga for the initial client-server architecture, Dimitar Misev for implementing the notation conversion, and Normen Müller for his feedback.

Christine Müller is a PhD candidate at the Jacobs University. In collaboration with Cristian Calude, University of Auckland, NZ, she has conducted a case study on the understanding of proof. Her current research focuses on *active documents* that provide configurable views and queries on technical contents. She is coordinator of the *panta rhei* project.