



UNIVERSITÄT ZU LÜBECK

Information Systems

CS4130-KP06

Prof. Dr. Sylvia Melzer

SoSe2026





Building Information Systems with JavaScript

Information Systems

Building Information Systems with JavaScript

- From structured research data to interactive systems
- Bridging data transformation and user interaction
- Focus on implementation and system integration
- Based on Humanities use cases

From Data Model to Information System

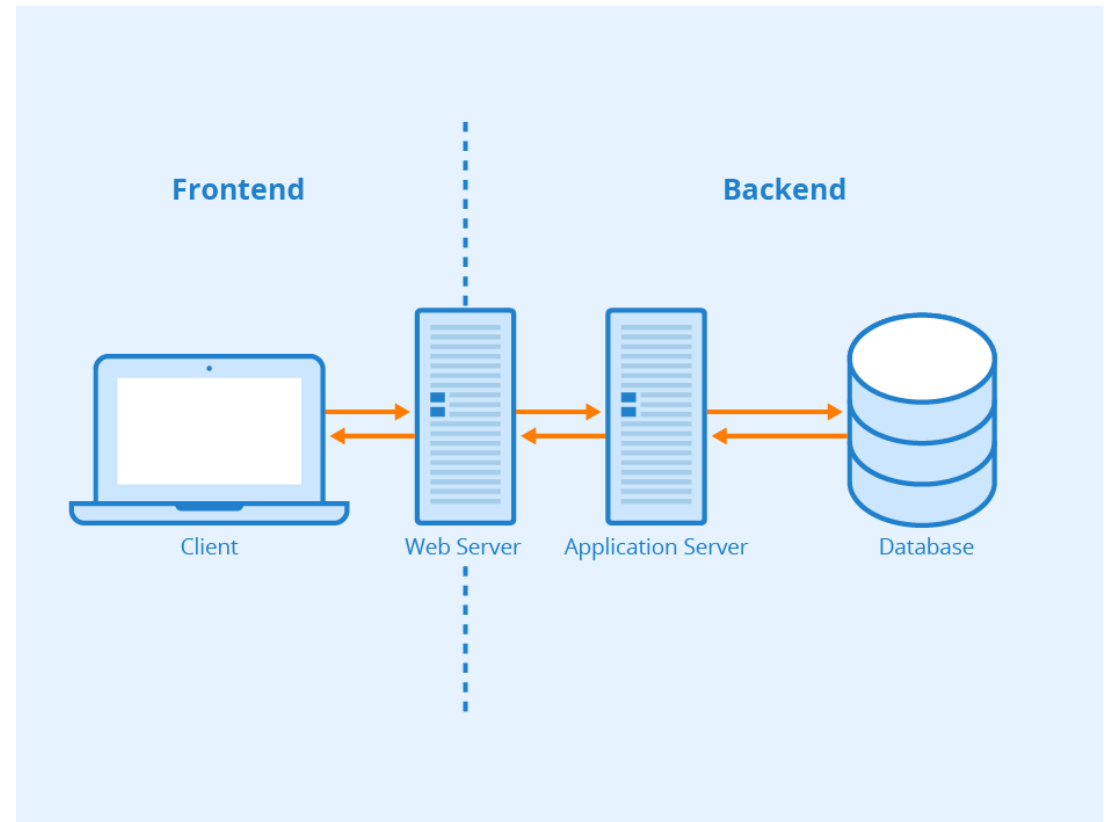
- Data has been modeled (e.g., TEI, JSON, CSV)
- Data has been transformed (e.g., DBoD: XSLT, Python)
- Data is stored in repositories (e.g., RDR)
- But: no direct interaction for users
- No querying, filtering, or exploration
- No interface for research questions

→ **The information system layer is missing**

From Data to System

- Data pipeline produces structured outputs (CSV/HML)
- These outputs are not yet usable systems
- A system requires interaction and access
- Users need filtering, querying, visualization
- Systems must respond dynamically to input
- Transformation \neq system

→ **A new layer must be introduced**



Why Building Information Systems?

- Research data alone is not directly usable
 - Large datasets require structured access
 - Researchers need filtering and exploration
 - Generic repository interfaces are often insufficient
 - Project-specific interfaces support research questions
 - Scalability becomes critical for growing datasets
- **Information systems bridge data and usage**

Technical View on an Information System

- A system consists of interconnected components
- Data layer: storage and structured datasets
- Logic layer: processing and querying
- Interface layer: user interaction
- Communication via APIs
- Real-time response to user input

→ **Focus today: implementation with JavaScript**

What You Will Learn

- Understand how information systems emerge from research data pipelines
- Learn how JavaScript connects repositories, APIs, and user interfaces
- Compare frontend-only and database-driven architectures
- Build interactive viewers for structured research data
- Understand the role of repositories such as Zenodo and RDR@UHH
- Evaluate scalability challenges in web-based systems
- Reflect on architectural decisions for research infrastructures

Architecture

- Browser acts as the user interface
 - Backend provides logic and data access
 - Database stores structured datasets
 - Communication via HTTP requests
 - Data exchanged in JSON format
 - Clear separation of concerns
- **Standard architecture of web-based systems**

Workflow Mapping

- EpiDoc → transformed into CSV (existing pipeline)
- CSV represents structured analytical data
- JavaScript backend exposes the data
- Frontend requests and visualizes the data
- Users interact via filtering and search
- System responds dynamically

Role of JavaScript

- JavaScript runs in the browser (frontend)
- JavaScript runs on the server (Node.js)
- Enables full-stack development
- Handles asynchronous data loading
- Processes and transforms JSON data
- Connects UI with backend logic

JavaScript - Features

- Influences from other programming languages
 - Notation similar to C/C++/Java
 - But little relation to Java
- Interpreted
- Dynamically typed
- Specific object-oriented concepts
- Interpreter integrated into web browsers
- Closely linked to HTML
 - Access to elements of the displayed document via the DOM API

Example: Web Application using JavaScript

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <title>JavaScript Example</title>
5 <script type="text/javascript">
6 var number = Math.floor(Math.random()*6)+1;
7 function check() {
8 var guessed = document.GuessForm.inputfield.value;
9 if(guessed==number){
10 alert("Congratulations! The dice value is "
11 + number + ".");
12 } else {
13 if(number<guessed) alert("My number is smaller!");
14 else alert("My number is bigger!");
15 I have thrown a dice.
16 Please guess the
17 number:
18 }
19 }
20 </script>
21 </head>
22 <body>
23 I have thrown a dice. Please guess the number:
24 <form name="GuessForm">
25 <input type="number" name="inputfield"
26 size="1" min="1" max="6"/>
27 <input type="button" value="Guess!" onClick="check()"/>
28 Guess!
29 </form>
30 </body>
31 </html>
```

I have thrown a dice. Please guess the number:

 Guess!

Embedding JavaScript in HTML

1. As a script section within an HTML document:

```
<script type="text/javascript">  
... // your javascript code goes here...  
</script>
```

2. Within HTML tags (to define handlers that are triggered by specific events):

- a. Defining the event as an attribute: The attribute value contains a sequence of instructions that is executed when the event occurs:

```
<input type="button" value="..." onClick="checkGuess()"/>
```

- b. Specifying a sequence of instructions in an anchor element (instead of a URL) using javascript:, which is executed when clicked:

```
<a href="javascript:checkGuess()">...</a>
```

Embedding JavaScript in HTML

3. In a separate file:

```
<script src="file.js" type="text/javascript"></script>
```

- The `<script>` element can be used multiple times
- JavaScript code executes immediately when the browser loads the document
 - Function definitions and event handlers do not execute immediately
- Multiple statements can be specified in HTML attributes, separated by semicolons (;), but this can be confusing
- Reduced loading time by placing JavaScript sections at the end of the document
- Separate file (exclusively) containing JavaScript code: file with the .js file extension

Data Flow in an Information System

- User performs an action (search/filter)
- Frontend sends request to API
- Backend processes the request
- Data is filtered or transformed
- Response is sent as JSON
- Frontend renders updated view


Frontend: What Does a Viewer Do?

- Displays structured data
- Allows filtering and searching
- Supports sorting and navigation
- Provides detail views
- Enables comparison of records
- Responds to user interaction
- Helps to analyze (research) data


Core Features of a Data Viewer

- Table-based representation of data
- Column-based filtering
- Row-level detail inspection
- Dynamic updates without reload
- Search across multiple fields
- Sorting and ordering

UNIVERSITÄT HAMBURG
CSMC Viewer CHAI Institute CSMC & UWA Cluster



Universität Hamburg
DER FORSCHUNG | DER LEHRE | DER BILDUNG



CENTRE FOR THE
STUDY OF
MANUSCRIPT
CULTURES

Epigraphic database of ancient Asia Minor

Franziska Weise; Universität Hamburg; [See more details](#).

Filters

Filter Categories Active: 0

Filter Categories – Map Active: 0

Select Visible Columns Visible: 7 / 19

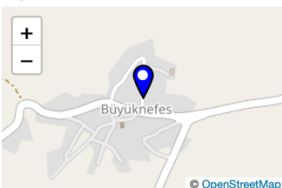
Full Text Search Inactive

EDAK00000002 ✕

Edition I.North Galatia 411

Findspot Büyüknefes

Findspot Map



Region Galatia

Place Tavium

Inscription Type dubia

Object Type architrave

Date commentary kaiserzeitlich (Mitchell); Anfang 4. Jh. n.Chr. (Ramsay, nach Kaisertitulatur)

Epoch

imperial

late antique / Byzantine

Text [--- im]perator VI co(n)[s]{ul} Γ

Table

Edition	Region	Place	Inscription Type	Object Type
<input type="text" value="Filter"/>	<input type="text" value="Filter"/>	<input type="text" value="Filter"/>	<input type="text" value="Filter"/>	<input type="text" value="Filter"/>
I.North Galatia 411	Galatia	Tavium	dubia	architrave
I.North Galatia 417	Galatia	Tavium	varia	architrave

Fetching Data (Concept)

- Frontend requests data via HTTP
- Uses Fetch API
- Asynchronous data loading
- No page reload required
- Data received as JSON
- Enables dynamic interfaces

Rendering Data

- Data must be transformed into HTML
 - Tables are common representations
 - Each row = one analytical unit
 - Columns represent attributes
 - Rendering updates dynamically
 - Based on user interaction
- **Data becomes visible and usable**

Backend & API

- Why an API?
- Separates frontend and backend logic
- Provides structured data access
- Enables reuse across applications
- Supports multiple clients
- Standardizes communication
- Improves scalability

Repositories as System Components

- Store and publish research data
- Provide metadata and identifiers
- Ensure FAIR compliance
- Limited interaction capabilities
- No advanced filtering or analysis
- No custom interfaces

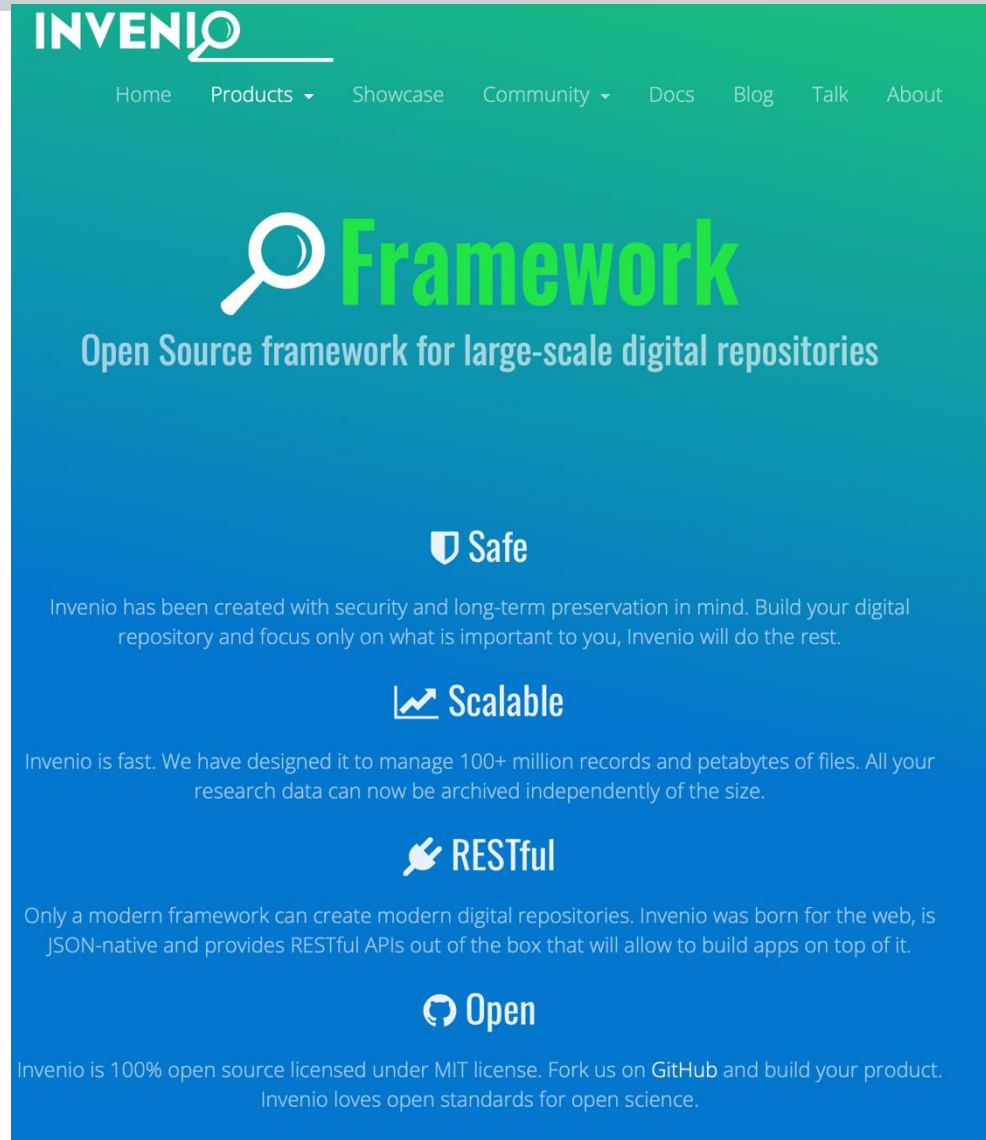
→ **Not full information systems**

Metadata Standards in RDRs

- DOI assignment and metadata standards are essential for research data archiving
- Key standards include DataCite, MARC, MODS, and Dublin Core
- These standards enable identification, description, and interoperability of data
- Different repositories implement these standards in different ways. Examples: InvenioRDM, RDR@UHH, and Zenodo
- Implementation depends on goals such as usability, interoperability, and community needs
- Comparing implementations reveals strengths and limitations for data management and preservation

Invenio Framework

- Open-source platform for building digital repositories
- Originally developed by CERN (Conseil européenne pour la recherche nucléaire)
- Provides a modular and extensible architecture
- Allows customization for different disciplines and use cases
- Supports access control and user management
- Includes file storage and search functionalities
- Designed for scalable and robust RDM systems
- is JSON-native and provides RESTful APIs out of the box that will allow to build apps on top of it



The screenshot shows the Invenio Framework website. At the top, the 'INVENIO' logo is displayed in white on a green background. Below the logo is a navigation menu with links for 'Home', 'Products', 'Showcase', 'Community', 'Docs', 'Blog', 'Talk', and 'About'. The main heading is 'Framework' in large green letters, with a magnifying glass icon to its left. Below this, the text reads 'Open Source framework for large-scale digital repositories'. The page features four key benefits, each with an icon and a brief description: 'Safe' (shield icon), 'Scalable' (line graph icon), 'RESTful' (hand cursor icon), and 'Open' (circular arrows icon).

Safe
Invenio has been created with security and long-term preservation in mind. Build your digital repository and focus only on what is important to you, Invenio will do the rest.


Scalable
Invenio is fast. We have designed it to manage 100+ million records and petabytes of files. All your research data can now be archived independently of the size.

RESTful
Only a modern framework can create modern digital repositories. Invenio was born for the web, is JSON-native and provides RESTful APIs out of the box that will allow to build apps on top of it.

Open
Invenio is 100% open source licensed under MIT license. Fork us on [GitHub](#) and build your product. Invenio loves open standards for open science.

InvenioRDM

- Open-source repository platform specifically designed for RDM
- Developed for institutions and organizations worldwide
- Highly flexible and extensible architecture
- Supports multiple metadata schemas (e.g., DataCite Metadata Schema)
- Enables customization for project- or domain-specific needs
- Ensures secure storage and long-term archiving of research data
- Promotes findability and interoperability across systems

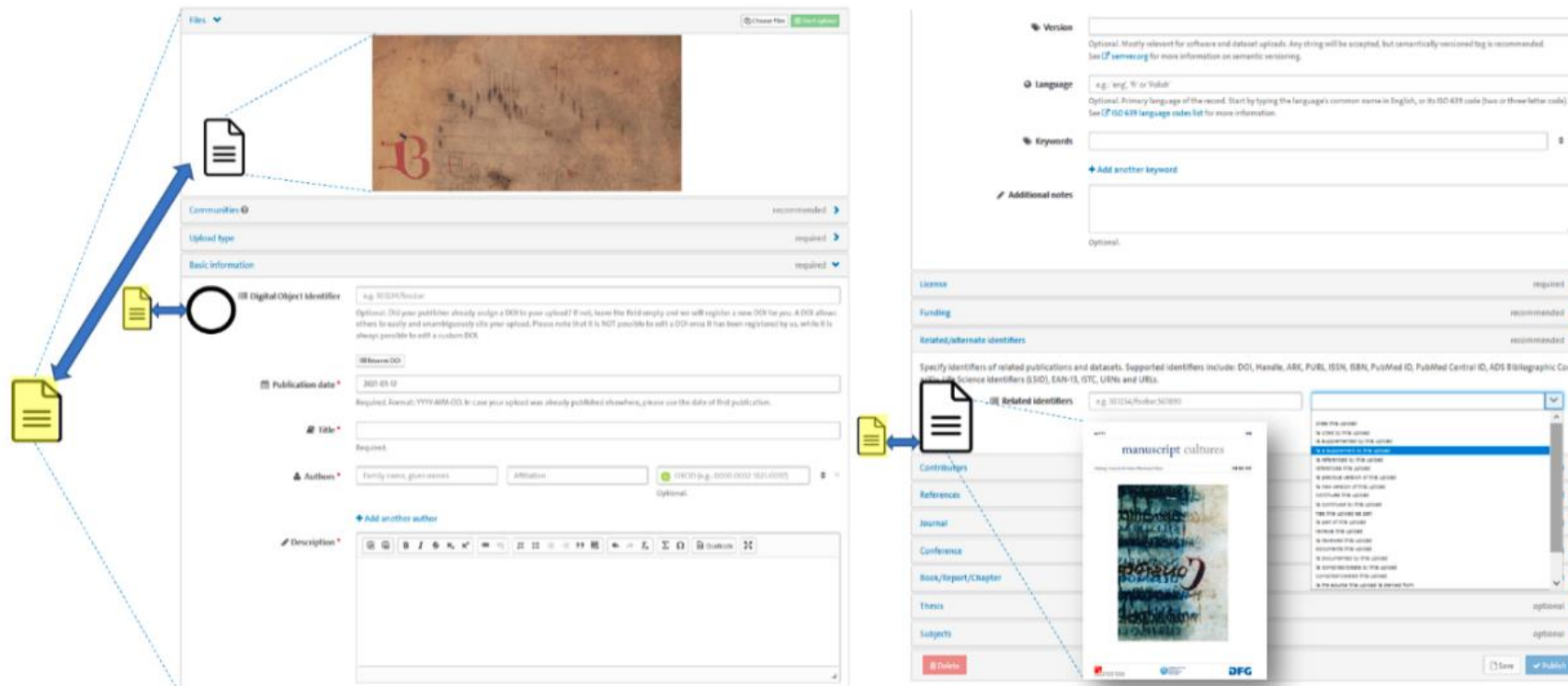


The screenshot shows the InvenioRDM website homepage. The header features the Invenio logo and navigation links: Home, Products, Showcase, Community, Docs, Blog, Talk, and About. The main content area has a dark blue background with the InvenioRDM logo (a magnifying glass over 'RDM') and the tagline 'The turn-key research data management repository'. Below this, a news item states 'July 23rd, 2025: InvenioRDM v13.0 released'. The page is organized into four sections: 'Status' (with a list icon), 'Roadmap' (with a road icon), 'Demo' (with a monitor icon), and 'Forum' (with a speech bubble icon). Each section includes a brief description of the content.

RDR@UHH

- Institutional repository for long-term storage and sharing of research data at the University of Hamburg
- Access restricted to UHH researchers for publishing; open access data available to all users
- Supports diverse data types (e.g., texts, measurements, audiovisual data, software) using DataCite metadata
- Enables linking of related datasets (e.g., references, supplements, part–whole relationships)
- Provides enhanced search across metadata and linked data, improving discoverability
- Offers export formats (e.g., BibTeX, JSON, Dublin Core, MARCXML) for interoperability and reuse
- Includes generic viewer and DBoD approach for flexible, project-specific data exploration (**NEW since 2026**)

RDR@UHH (Upload)



The screenshot displays the RDR@UHH upload interface. On the left, a 'Files' section shows a document icon and a thumbnail of a manuscript page. Below it, the 'Basic information' section includes fields for 'Digital Object Identifier' (with a white dot icon), 'Publication date', 'Title', 'Author', and 'Description'. A blue arrow points from a yellow document icon to the 'Digital Object Identifier' field. On the right, the 'Related identifiers' section shows a list of identifiers with a white document icon and a blue arrow pointing to the 'Related identifiers' field. A thumbnail of a book cover titled 'manuscript cultures' is also visible. The interface includes various tabs like 'Version', 'Language', 'Keywords', 'Additional notes', 'License', 'Funding', and 'Contributors'. A 'Publish' button is located at the bottom right.

Upload research data at the RDR. Yellow document symbol: RDR research data; white document symbol: research data, e.g., publications, measurements and other resources; white dot: DOI; blue arrow: linking of data.

RDR@UHH (Linking)

July 25, 2018 Dataset Open Access

Multispectral Imaging of Manuscript Atri, Archivio Capitolare, Museo della Basilica Cattedrale, C4-II, Frammento 17

Shevchuk, Ivan, & Janke, Andreas

Multispectral imaging of this fragment was performed during the workshop 'Liturgical Books and Music Manuscripts with Polyphonic Settings of the Mass in Medieval Europe' held in Hamburg, 3–4 November 2011, at the Sonderforschungsbereich 950 'Manuskriptkulturen in Asien, Afrika und Europa' (Centre for the Study of Manuscript Cultures), organised by Oliver Huck and Andreas Janke.

The results are published in:

- Sebastian Bosch, Claudia Colini, Oliver Huck, Andreas Janke, and Ivan Shevchuk, 'The Atri Fragment Revisited I: Multispectral Imaging and Ink Identification', *Manuscript Cultures*, 11 (2018), 140–156.
- Andreas Janke and Francesco Zimei, 'The Atri Fragment Revisited II: From the Manuscript's Context to the Tradition of the *Ballata del lo sa chi*', in Oliver Huck and Andreas Janke (eds.), *Liturgical Books and Music Manuscripts with Polyphonic Settings of the Mass in Medieval Europe* (Musica Manuscripta, 9), Olms: Hildesheim, 2020, 135–155.

The labels of the four TIF files (Figures 1–4) correspond to the 2018 publication.
(Published online September 8, 2020)

This project was carried out in the context of the SFB 950 'Manuskriptkulturen in Asien, Afrika und Europa' funded by the German Research Foundation (Deutsche Forschungsgemeinschaft, DFG) and within the scope of the Centre for the Study of Manuscript Cultures (CSMC).

Preview

Files (13.3 GB)

Name	Size	Download
00_preview_Atri-fragment.tif	2.6 MB	Download

Publication date: July 25, 2018
DOI: [10.25592/uuhfdm.1472](https://doi.org/10.25592/uuhfdm.1472)

Keywords(s): CSMC, SFB950, Manuscript, Fragment, Atri, Atri fragment, Archivio Capitolare, Museo della Basilica Cattedrale, 2011, 803, parchment, 16th century CE, musical notation, ars nova notation, Latin, Italian, Italy, Trento, Antonio Zacara da Teramo, Agostino 17, Damaged Writing, Artifact Profiling, Multispectral Imaging (MI)

Has part: https://www.manuscript-cultures.uni-hamburg.de/MC/articles/mc11_bosch_et_al.pdf

Supplement to: https://www.manuscript-cultures.uni-hamburg.de/MC/articles/mc11_bosch_et_al.pdf

License (for files): CC Creative Commons Attribution-NonCommercial-Share Alike 4.0 International

Versions

Version	Date
Version 1 10.25592/uuhfdm.1472	Jul 25, 2018

Share

Cite as
Shevchuk, Ivan, & Janke, Andreas. (2018). Multispectral imaging of Manuscript Atri, Archivio Capitolare, Museo della Basilica Cattedrale, C4-II, Frammento 17 [Data set]. <http://doi.org/10.25592/uuhfdm.1472>

Export

BBOTX CSL DataCite Dublin Core JSON JSON-LD MARCXML [DFG](#)

link to supplementing document



<https://doi.org/10.25592/uuhfdm.1472>

RDR@UHH (Search results)

RDR

Found 517 results. < 1 2 3 4 5 6 7 8 9 > Sort by: Best match asc.

1472 [July 25, 2018 \(v1\)](#) Dataset Open Access View

Multispectral Imaging of Manuscript Atri, Archivio Capitolare, Museo della Basilica Cattedrale, C4-II, Frammento 17

Shevchuk, Ivan, Janke, Andreas.

Multispectral imaging of this fragment was performed during the workshop 'Liturgical Books and Music Manuscripts with Polyphonic Settings of the Mass in Medieval Europe' held in Hamburg, 3–4 November 2017, at the Sonderforschungsbereich 950 'Manuskriptkulturen in Asien, Afrika'

Uploaded on September 8, 2020

1430 [July 25, 2018 \(v1\)](#) Dataset Open Access View

Scientific Analysis of Manuscript Atri, Archivio Capitolare, Museo della Basilica Cattedrale, C4-II, Frammento 17

Bosch, Sebastian, Colini, Claudia, Janke, Andreas.

Scientific analysis of this fragment was performed during the workshop 'Liturgical Books and Music Manuscripts with Polyphonic Settings of the Mass in Medieval Europe' held in Hamburg, 3–4 November 2017, at the Sonderforschungsbereich 950 'Manuskriptkulturen in Asien, Afrika'

Uploaded on August 20, 2020

[August 1, 2016 \(v1\)](#) Book Open Access View

The San Lorenzo Palimpsest. Florence, Archivio del Capitolo di San Lorenzo Ms. 2211. Introductory Study and Multispectral Images

Janke, Andreas; Nadas, John;

Google

About 226 results (0,63 seconds)

[www.fdr.uni-hamburg.de](#) › record ▼

Scientific Analysis of Manuscript Atri, Archivio Capitolare ...

25 Jul 2018 — Scientific Analysis of Manuscript Atri, Archivio Capitolare, Museo della Basilica Cattedrale, C4-II, Frammento 17. Bosch, Sebastian; Colini ...

[eudat7-devel.dkrz.de](#) › dataset ▼

Scientific Analysis of Manuscript Atri, Archivio Capitolare, Museo ...

... Atri, Archivio Capitolare, Museo della Basilica Cattedrale, C4-II, Frammento 17 ... Study of Manuscript Cultures), organised by Oliver Huck and Andreas Janke.

[eudat7-devel.dkrz.de](#) › dataset ▼

Multispectral Imaging of Manuscript Atri, Archivio Capitolare, Museo ...

... Archivio Capitolare, Museo della Basilica Cattedrale, C4-II, Frammento 17 ... 'Liturgical Books and Music Manuscripts with Polyphonic Settings of the Mass in ...

1430

Google has not found the record 1472.

Zenodo

- Open-access repository built on Invenio and hosted at CERN
- Developed within the OpenAIRE initiative funded by the European Commission
- Available to researchers worldwide for publishing data, software, and publications
- Automatically assigns DOIs to all uploads, improving citation and discoverability
- Supports linking between related datasets (similar to RDR@UHH)
- Offers a wide range of export formats (e.g., JSON, DataCite, Dublin Core, MARCXML, BibTeX, GeoJSON)
- Designed for broad dissemination and interoperability across international research communities

Why Repositories Are Not Enough

- Provide storage, not interaction
- Limited filtering and querying
- No custom analysis tools
- Generic interfaces only
- Not tailored to research questions
- No project-specific views

DataTable Approach

- Simple frontend-based table rendering
 - Uses libraries like DataTables
 - Data loaded entirely into browser
 - Supports filtering and sorting
 - Easy to implement
 - Works well for small datasets
- **Good for rapid prototyping**

Epigraphic database of ancient Asia Minor

Search all:

Edition	Region	Place	Inscription type	
Anderson, JHS 19 (1899) 116, Nr. 109	Galatia	Kinna	funerary inscription	ste
Anderson, JHS 19 (1899) 116, Nr. 110	Galatia	Kinna	funerary inscription	
Bosch, Ankara 290	Galatia	Ancyra	building inscription	ar
Bosch, Ankara 291	Galatia	Ancyra	building inscription	bl
Bosch, Ankara 292	Galatia	Ancyra	building inscription	
Bosch, Ankara 293	Galatia	Ancyra	building inscription	
Bosch, Ankara 305	Galatia	Ancyra	honorary inscription	
Bosch, Ankara 306	Galatia	Ancyra	building inscription	
Bosch, Ankara 54	Galatia	Ancyra	building inscription	ar

Anderson, JHS 19 (1899) 116, Nr. 109

Region: Galatia
Place: Kinna
Inscription type: funerary inscription
Object type: stele
Object description: Fragment
Date (epoch):
Date (century):
Date commentary:
Discovery:
Museum/Archive:

Text:

Μάνης Ἀππια-
νοῦ ἀνέστησεν
γ[υ]νηκί Ττινία
καὶ Τατεὶ Τισά[νδρου]
5 μνήμης
χάρ[ιν].

Apparatus criticus:

HTML

- For DataTables to be able to enhance an HTML table, the table must be valid, well formatted HTML, with a header (thead) and a single body (tbody)
- An optional footer (tfoot) can also be used

```
1 <table id="myTable" class="display">
2   <thead>
3     <tr>
4       <th>Column 1</th>
5       <th>Column 2</th>
6     </tr>
7   </thead>
8   <tbody>
9     <tr>
10      <td>Row 1 Data 1</td>
11      <td>Row 1 Data 2</td>
12    </tr>
13    <tr>
14      <td>Row 2 Data 1</td>
15      <td>Row 2 Data 2</td>
16    </tr>
17  </tbody>
18 </table>
```

HTML

DataTables (Javascript library)

- DataTables is a powerful Javascript library for adding interaction features to HTML tables, and while simplicity is a core design principle for the project as a whole
- You need only include two additional files in your page:

- The DataTables Javascript file
- The DataTables CSS file

```
1 <link rel="stylesheet" href="/DataTables/datatables.css" />
2
3 <script src="/DataTables/datatables.js"></script>
```

HTML

- DataTables has only one library dependency →jQuery
- JQuery: The most widely used JavaScript library, used on over 70% of websites
- DataTables uses jQuery as a utility library, allowing the venerable library to deal with many of the complexities of event handling and DOM manipulation

DataTables

With jQuery

```
1 | $(document).ready( function () {  
2 |     $('#myTable').DataTable();  
3 | } );
```

w/o jQuery

```
1 | let table = new DataTable('#myTable', {  
2 |     // options  
3 | });
```

HTML

```
<!DOCTYPE html>
<html lang="de">
<head>
  <meta charset="UTF-8">
  <title>DataTables Beispiel</title>
  <link rel="stylesheet" href="https://cdn.datatables.net/2.3.7/css/dataTables.dataTables.css">
  <script src="https://code.jquery.com/jquery-3.7.1.min.js"></script>
  <script src="https://cdn.datatables.net/2.3.7/js/dataTables.js"></script>
  <script defer src="app.js"></script>
</head>
<body>

<h2>Benutzerliste</h2>
<table id="example" class="display">
  <thead>
    <tr>
      <th>Name</th>
      <th>Position</th>
      <th>Ort</th>
      <th>Alter</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>Max Mustermann</td>
      <td>Entwickler</td>
      <td>Hamburg</td>
      <td>30</td>
    </tr>
  </tbody>
</table>
```



HTML

```
<table id="example" class="display">
  <thead>
    <tr>
      <th>Name</th>
      <th>Position</th>
      <th>Ort</th>
      <th>Alter</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>Max Mustermann</td>
      <td>Entwickler</td>
      <td>Hamburg</td>
      <td>30</td>
    </tr>
    <tr>
      <td>Anna Schmidt</td>
      <td>Designer</td>
      <td>Berlin</td>
      <td>27</td>
    </tr>
    <tr>
      <td>John Doe</td>
      <td>Manager</td>
      <td>München</td>
      <td>40</td>
    </tr>
  </tbody>
</table>
</body>
</html>
```

Benutzerliste

Name	Position	Ort	Alter
Max Mustermann	Entwickler	Hamburg	30
Anna Schmidt	Designer	Berlin	27
John Doe	Manager	München	40

JS

```
$(document).ready(function () {  
    $('#example').DataTable({  
        paging: true,  
        searching: true,  
        ordering: true,  
        pageLength: 5,  
        language: {  
            search: "Suche:",  
            lengthMenu: "Zeige _MENU_ Einträge",  
            info: "Zeige _START_ bis _END_ von _TOTAL_ Einträgen"  
        }  
    });  
});
```

Benutzerliste

Zeige EinträgeSuche:

Name	Position	Ort	Alter
Anna Schmidt	Designer	Berlin	27
John Doe	Manager	München	40
Max Mustermann	Entwickler	Hamburg	30

Zeige 1 bis 3 von 3 Einträgen

« < 1 > »

HTML + JS w/o jQuery

HTML

```
<table id="example">
  <thead>
    <tr>
      <th>Name</th>
      <th>Position</th>
      <th>Ort</th>
      <th>Alter</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>Max Mustermann</td>
      <td>Entwickler</td>
      <td>Hamburg</td>
      <td>30</td>
    </tr>
  </tbody>
</table>
```

JS

```
new DataTable('#example', {
  paging: true,
  searching: true,
  ordering: true
});
```

Implementation Process

- Step 1: Retrieve data via API
- Step 2: Transform into JSON
- Step 3: Process in backend
- Step 4: Serve via API
- Step 5: Render in viewer
- Step 6: Enable interaction

When Does Data Tables Stop Working?

- Small datasets → responsive interaction
 - Medium datasets → slower filtering
 - Large datasets → browser bottlenecks
 - Full dataset loaded into memory
 - Rendering becomes inefficient
 - User experience degrades
- **Need database-driven architectures**

From Frontend Tables to Distributed Systems

- Frontend-only viewers are easy to build
- But scalability becomes a challenge
- Large datasets exceed browser capabilities
- Offline access may be required
- Multiple users need synchronization
- Distributed architectures solve these problems

→ **Approach: PouchDB and CouchDB**

PouchDB

- PouchDB is an in-browser database that allows applications to save data locally, so that users can enjoy all the features of an app even when they're offline
- The data is synchronized between clients, so users can stay up-to-date wherever they go
- PouchDB also runs in Node.js and can be used as a direct interface to CouchDB-compatible servers
- The API works the same in every environment, so you can spend less time worrying about browser differences, and more time writing clean, consistent code
- PouchDB is a free open-source project, written in JavaScript

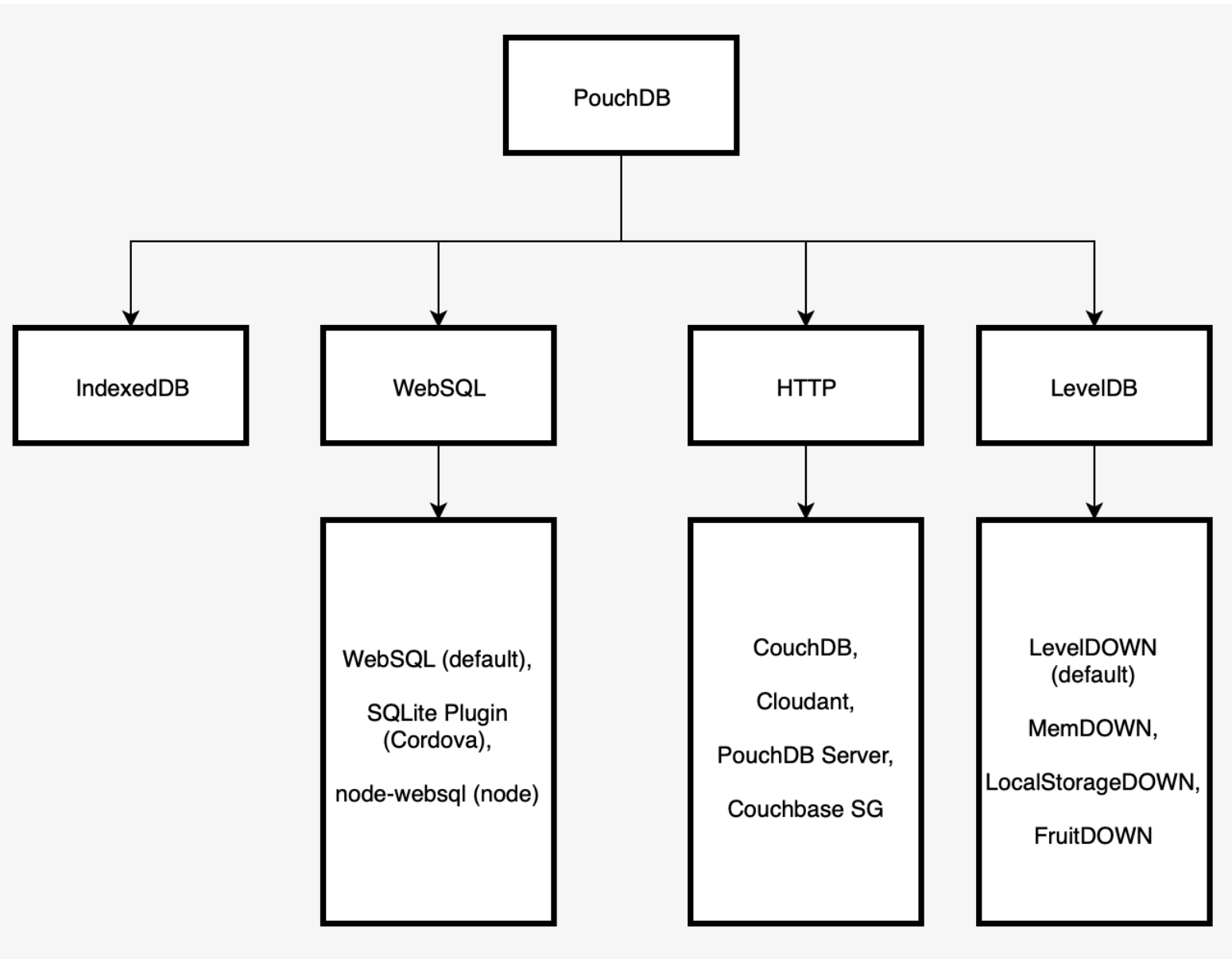
PouchDB

- PouchDB supports all modern browsers, using IndexedDB under the hood and falling back to WebSQL where IndexedDB isn't supported. It is fully tested and supported in:
 - Firefox 29+ (Including Firefox OS and Firefox for Android)
 - Chrome 30+
 - Safari 5+
 - Internet Explorer 10+
 - Opera 21+
 - Android 4.0+
 - iOS 7.1+
 - Windows Phone 8+

IndexedDB API

- IndexedDB is a low-level API for storing large amounts of structured data on the client side, including files and blobs
- This API uses indexes to enable high-performance searches of this data
- While Web Storage is useful for storing smaller amounts of data, it is less suitable for storing larger amounts of structured data → IndexedDB offers a solution for this
- IndexedDB is a transactional database system similar to an SQL-based relational database management system (RDBMS)
- Unlike SQL-based RDBMSs, which use fixed-column tables, IndexedDB is a JavaScript-based object-oriented database
- IndexedDB allows you to store and retrieve objects that are indexed by a key

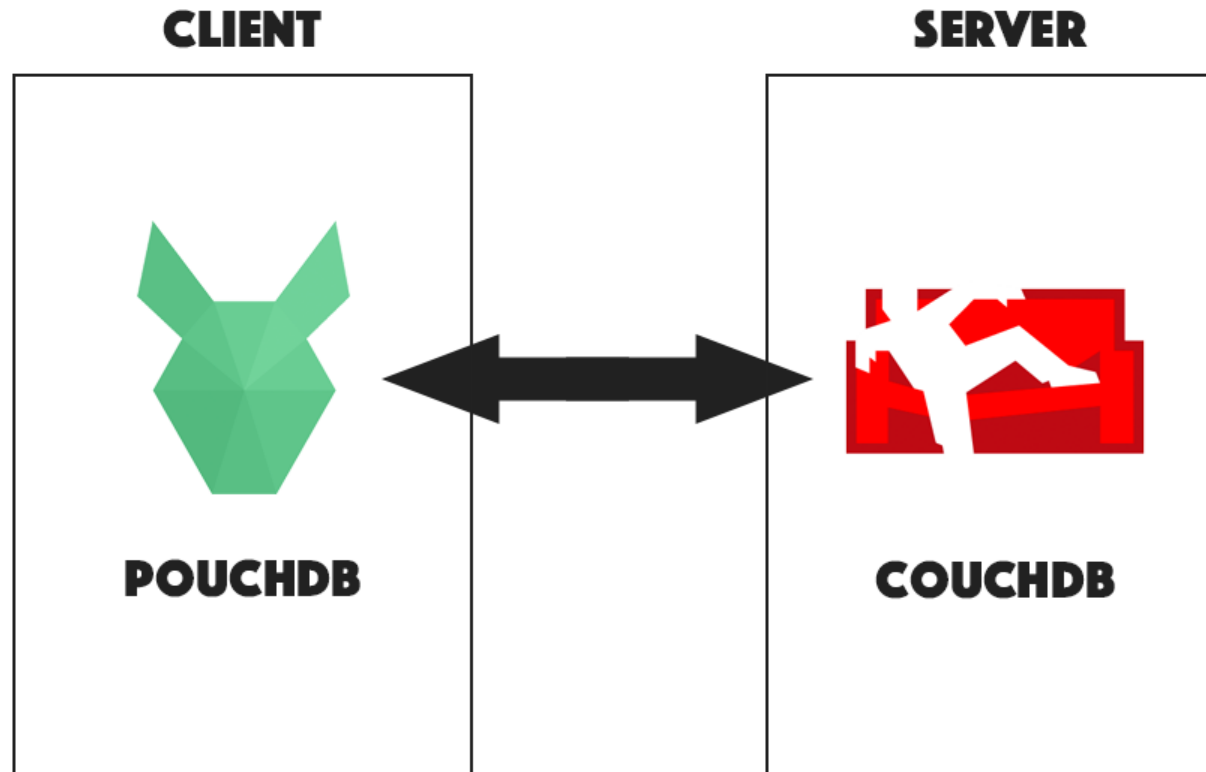
PouchDB



CouchDB

- Apache CouchDB is an open-source document-oriented NoSQL database, implemented in Erlang
- CouchDB uses multiple formats and protocols to store, transfer, and process its data
- It uses JSON to store data, JavaScript as its query language using MapReduce, and HTTP for an API.

A Database Built to Embrace Conflict

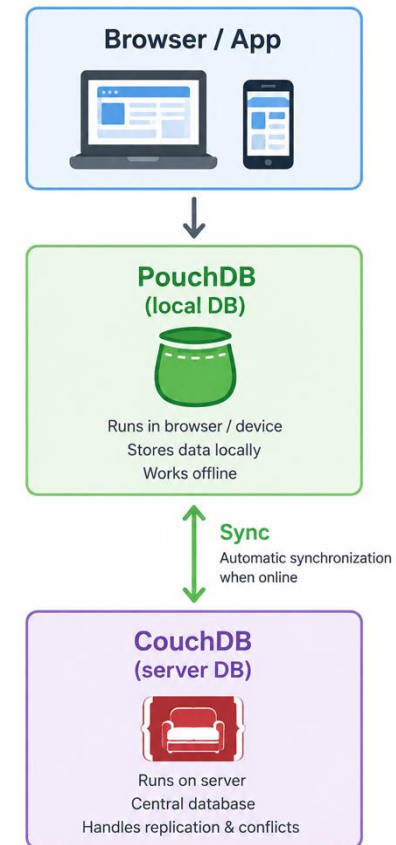


CouchDB – Aim

- Traditional databases try to avoid conflicts using locks and strict consistency
- CouchDB is designed to embrace conflicts and resolve them later
- Ideal for:
 - Offline-first apps
 - Distributed systems
 - Multi-device synchronization
 - Unreliable network environments

CouchDB + PouchDB Workflow

1. User works locally in browser (PouchDB)
2. Data stored offline
3. Internet reconnects
4. PouchDB syncs with CouchDB
5. Conflicts handled through revision system



CouchDB + PouchDB Workflow

Advantages

- Excellent offline support
- Seamless synchronization
- Fault-tolerant architecture
- Scales across distributed systems
- Easy JSON-based development

Challenges

- Conflict resolution complexity
- Eventual consistency may confuse developers
- Not ideal for:
 - Strong transactional systems
 - Complex relational queries

Build Viewer with PouchDB (HTML)

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4 <meta charset="UTF-8">
5 <meta name="viewport" content="width=device-width, initial-scale=1.0">
6 <link rel="stylesheet" type="text/css" href="https://cdn.datatables.net/1.11.5/css/jquery.dataTables.css">
7 <script type="text/javascript" charset="utf8" src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
8 <script type="text/javascript" charset="utf8" src="https://cdn.datatables.net/1.11.5/js/jquery.dataTables.js"></script>
9 <title>EDAK.CSV-Daten mit PouchDB darstellen</title>
10 <style>
11   tfoot {
12     display: table-header-group;
13   }
14 </style>
15 </head>
16 <body>
17   <h1>EDAK 2</h1>
18   <table id="example" class="display" style="width:50%; float:left;"></table>
19
20   <script src="/static/pouchdb.min.js"></script>
21   <script src="/static/app.js"></script>
22
23 </body>
24 </html>
```

Build Viewer with PouchDB (JavaScript)

```
1  const db = new PouchDB('csv_database');
2
3  // Funktion zum Löschen aller vorhandenen Daten in der PouchDB-Datenbank
4  function clearDatabase() {
5      return db.allDocs({ include_docs: true })
6          .then(result => {
7              const docs = result.rows.map(row => ({
8                  _id: row.doc._id,
9                  _rev: row.doc._rev,
10                 _deleted: true // Markiere Dokumente als gelöscht
11             }));
12             return db.bulkDocs(docs); // Lösche alle Dokumente in einem Bulk-Vorgang
13         });
14 }
```

Build Viewer with PouchDB (JavaScript)

```
16 // Funktion zum Abrufen von CSV-Daten aus PouchDB und Anzeige in der Tabelle
17 function displayCSVData() {
18     db.allDocs({ include_docs: true })
19         .then(result => {
20             const csvData = result.rows.map(row => row.doc);
21             processData(csvData); // Rufen Sie die Funktion processData auf, um die Daten anzuzeigen
22         })
23         .catch(error => {
24             console.error('Fehler beim Abrufen der CSV-Daten:', error);
25         });
26 }
```

```

28 // Funktion zum Importieren von CSV-Daten in PouchDB
29 function importCSVToPouchDB(csvData) {
30     return clearDatabase() // Löschen Sie zuerst vorhandene Daten
31     .then(() => {
32         // Importieren Sie die CSV-Daten wie zuvor
33         const lines = csvData.trim().split('\n');
34         headers = lines[0].split(';').map(header => header.trim());
35         const docs = lines.slice(1).map(line => {
36             const values = parseCSVLine(line);
37             const doc = {};
38             headers.forEach((header, index) => {
39                 doc[header] = values[index];
40             });
41             return doc;
42         });
43
44         // Dokumente in PouchDB-Datenbank speichern
45         return db.bulkDocs(docs);
46     })
47     .then(() => {
48         console.log('CSV-Daten erfolgreich in PouchDB importiert.');
```

displayCSVData(); // Nach dem Importieren die Daten verarbeiten und in der Tabelle anzeigen

```

50     })
51     .catch(error => {
52         console.error('Fehler beim Importieren der CSV-Daten in PouchDB:', error);
53     });
54 }

```



```
56 // Funktion zum Verarbeiten der Daten und Anzeigen in der Tabelle
57 function processData(dataSet) {
58     const tableData = dataSet.map(data => {
59         const rowData = {};
60         Object.keys(data).forEach(key => {
61             rowData[key] = data[key];
62         });
63         return rowData;
64     });
65
66     const columns = Object.keys(dataSet[0]).map(column => {
67         return { title: column, data: column };
68     });
69
70     var table = $('#example').DataTable({
71         data: tableData,
72         columns: [
73             { title: headers[0], data: columns[0].data },
74             { title: headers[1], data: columns[1].data },
75             { title: headers[2], data: columns[2].data },
76             { title: headers[3], data: columns[3].data },
77             { title: headers[4], data: columns[4].data },
```



```
76     { title: headers[3], data: columns[3].data },
77     { title: headers[4], data: columns[4].data },
78 ],
79 paging: true,
80 dom: 'Pfrtip',
81 initComplete: function () {
82     this.api().columns().every(function () {
83         var column = this;
84         var header = $(this.header());
85         var input = $('<input type="text" placeholder="Search" />')
86             .appendTo($(header))
87             .on('keyup change', function () {
88                 if (column.search() !== this.value) {
89                     column.search(this.value).draw();
90                 }
91             });
92     });
93 },
94 searching: true,
95 mark: true
96 });
97 }
```

Build Viewer with PouchDB (JavaScript)

```
138 // CSV-Daten aus der Datei edak.csv laden und in PouchDB importieren
139 fetch('/static/test.csv') // Stellen Sie sicher, dass der Pfad zur CSV-Datei korrekt ist
140   .then(response => response.text())
141   .then(importCSVToPouchDB)
142   .catch(error => {
143     | console.error('Fehler beim Laden der CSV-Datei:', error);
144   });
```

Build Viewer with PouchDB (JavaScript)

EDAK 2

Search:

Edition	Language	Region	Place	Inscription type
<input type="text" value="Search"/>	<input type="text" value="Search"/>	<input type="text" value="Search"/>	<input type="text" value="Search"/>	<input type="text" value="Search"/>
Anderson, JHS 19 (1899) 116, Nr. 109	Greek	Galatia	Kinna	funerary inscription
Anderson, JHS 19 (1899) 116, Nr. 110	Greek	Galatia	Kinna	funerary inscription
Bosch, Ankara 290	Greek	Galatia	Ancyra	building inscription
Bosch, Ankara 291	Greek	Galatia	Ancyra	building inscription
Bosch, Ankara 292	Greek	Galatia	Ancyra	building inscription
Bosch, Ankara 293	Greek	Galatia	Ancyra	building inscription
Bosch, Ankara 305	Latin	Galatia	Ancyra	honorary inscription
Bosch, Ankara 306	Greek	Galatia	Ancyra	building inscription
Bosch, Ankara 54	Latin	Galatia	Ancyra	building inscription
CIG 4045	Greek	Galatia	Ancyra	religious inscription

Showing 1 to 10 of 98 entries

Previous 1 2 3 4 5 ... 10 Next

Why a Generic Viewer?

- Many projects share similar requirements
- Rewriting viewers is inefficient
- Generic components improve reuse
- Configuration enables customization
- Same system, different datasets
- Reduces development effort

→ **Reusable infrastructure for research projects!!!**

Generic Viewer (JS)

- A Generic Viewer is a reusable JavaScript-based data viewer
 - Designed for structured research datasets (Input: CSV(s))
 - Works with project-specific configurations
 - Separates data, logic, and presentation
 - Enables filtering, searching, and browsing
- **A generic information system component**

Generic Viewer (index.html)

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <link rel="icon" href="/favicon.ico" type="image/x-icon; charset=binary" />
5      <title>Dev &ndash; GenView</title>
6  </head>
7  <body>
8      <h1>Development Server &ndash; Generic CSMC Viewer</h1>
9      <ul>
10         <li><a href="./viewer/">Demo</a> (Toy data, but shows all features)</li>
11         <li><a href="./docs/" target="_blank">Dokumentation (JS, Viewer) &nearr;</a></li>
12         <li><a href="./docs/csmcgen/" target="_blank">Dokumentation (Python, CSMCGen CLI) &nearr;</a></li>
13     </ul>
14     <ul>
15         <li><a href="https://git.chai.uni-hamburg.de/CSMC-Files/Generic-Viewer" target="_blank">Repository</li>
16     </ul>
17 </body>
18 </html>
```

Generic Viewer

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <link rel="icon" href="/favicon.ico" type="image/x-icon; charset=binary" />
5      <title>Dev &dash; GenView</title>
6  </head>
7  <body>
8      <h1>Development Server &dash; Generic CSMC Viewer</h1>
9      <ul>
10         <li><a href="/viewer/">Demo</a> (Toy data, but shows all features)</li>
11         <li><a href="/docs/" target="_blank">Dokumentation (JS, Viewer) &nearr;</a></li>
12         <li><a href="/docs/csmcgen/" target="_blank">Dokumentation (Python, CSMC Gen CLI) &nearr;</a></li>
13     </ul>
14     <ul>
15         <li><a href="https://git.chai.uni-hamburg.de/CSMC-Files/Generic-Viewer" target="_blank">Repository</li>
16     </ul>
17 </body>
18 </html>
```

Generic Viewer (viewer_config.js)

```
41  const VIEWER_CONFIG = {
42      // css (jQuery) selector of html element, the viewer populates in
43      "element" : "div#viewer",
44      // de-/active languages
45      "languages" : [
46          "en",
47          "de"
48      ],
49      // run the viewer in debug mode (uses less caching, etc.)
50      "debug" : true,
51      // the base url/ path (with trailing slash)
52      "base" : "./",
53      // location of assets file (relative to 'base')
54      "assets" : "static/assets.json",
55      // some information about the data shown by the viewer (will also be included in export)
56      "about" : {
57          // heading (applied to h1 of 'show=true') of the page
58          "heading" : "Demo of Generic CSMC Viewer",
59          // title of the page (applied if 'show=true')
60          "title" : "Demo CSMC Viewer",
61          // authors of the data
62          "authors" : "Magnus Bender",
```



More in the
exercise.

Viewer in RDR



HOME UPLOAD LOGIN

March 24, 2026

Dataset Open Access

Epigraphic database of ancient Asia Minor

Weise, Franziska; Harter-Uibopuu, Kaja


This is a beta version of the EDAK (Epigraphische Datenbank zum Antiken Kleinasien).

The project of the Department for Ancient History at the University of Hamburg aims to collect the widespread published Greek and Latin inscriptions of the regions of modern Turkey and to present them in a database with a short description and a commentary. Currently, the dataset contains inscriptions of ancient Lydia, Galatia, Paphlagonia, Phrygia, and Proseilemmene. TEI/EpiDoc is used as the standard for describing inscription data.

The beta version still has functional errors and only shows an excerpt from the entire data set of inscriptions to present the basic technical functions of the new search interface, which is developed and continuously optimized by Research Field F (Data Linking).

Preview

edak_beta_2025.2.csmc



CENTRE FOR THE STUDY OF MANUSCRIPT CULTURES

View Data or Extract Data for edak_beta_2025.2.csmc (on an external server)

Debug information

Files (36.8 MB)

Name	Size	
edak_beta_2025.2.csmc	36.8 MB	Preview Download

md5:b2055246265e8729611d7050d842aba5

ZENTRUM FÜR NACHHALTIGES FORSCHUNGSDATENMANAGEMENT



Publication date:

March 24, 2026

DOI:

DOI: 10.25592/uhhfdm.18503

Communities:

Centre for the Study of Manuscript Cultures UHH

License (for files):

Creative Commons Attribution 4.0 International

Versions

Version	Date
Version 2025.2	Mar 24, 2026

Cite all versions? You can cite all versions by using the DOI 10.25592/uhhfdm.18502. This DOI represents all versions, and will always resolve to the latest one.

Cite record as

Weise, Franziska, & Harter-Uibopuu, Kaja. (2026). Epigraphic database of ancient Asia Minor (Version 2025.2) [Data set]. <http://doi.org/10.25592/uhhfdm.18503>

Start typing a citation style...

Export

BibTeX CSL DataCite Dublin Core JSON JSON-LD MARCXML Mendeley

Architecture Evolution

Stage	Technology	Limitation
Static files	CSV/HTML	No interaction
DataTables	Frontend-only	Poor scalability
API + Viewer	JavaScript	Better interaction
PouchDB/CouchDB	Distributed DB	More complexity
Generic Viewer	Config-driven	Requires structured data

Conclusion

- Data transformation alone does not create an information system
- JavaScript enables interaction, querying, and visualization
- Repositories provide storage and interoperability
- Viewers connect users with research data
- Scalability requires architectural decisions
- PouchDB and CouchDB support distributed, offline-capable systems
- Generic viewers enable reusable research infrastructures

- **Research data becomes useful through interaction.**
- **Python prepares the data. JavaScript makes it usable.**