



# OpenAtlas

A web based tool to acquire and manage historical and archaeological data

Alexander Watzinger  
Bernhard Koschiček-Krombholz

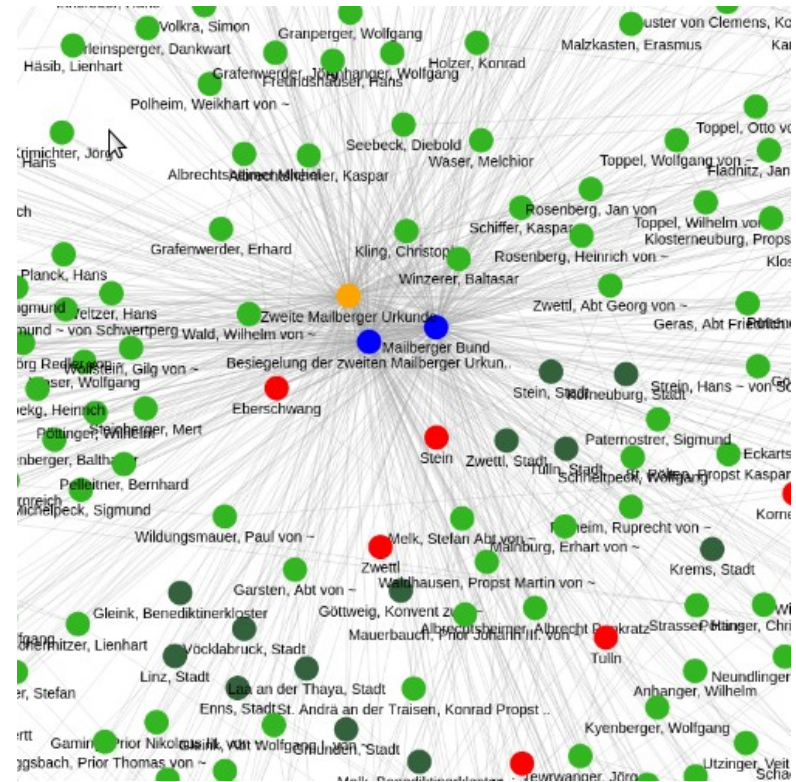
CC-BY 4.0



# OpenAtlas

<https://openatlas.eu>

- Open source, browser based database software
- Acquire, edit and manage research data
- Historical, archaeological and prosopographic projects
- Developed by a small core team



<https://demo.openatlas.eu/overview/network/>

# OpenAtlas Team



**Stefan Eichert**  
Idea, Concept and Data Modeling



**Alexander Watzinger**  
Lead Developer and Concept



**Bernhard  
Koschiček-Krombholz**  
Software Development, API



**Christoph Hoffmann**  
Frontend Development



**Nina Brundke**  
Bioarchaeological Expertise



**Jan Belik**  
Logo Design and Design Consulting

And many contributors: <https://openatlas.eu/team>

# Projects

## A small selection

**THANADOS**

The Anthropological and Archaeological Database of Sepultures  
PI: Stefan Eichert, Nina Brundke  
2019 - 2021

**MAMEMS**

Mount Athos in Medieval Eastern Mediterranean Society  
PI: Zachary Chitwood  
2020 - 2025

**Indigo**

INventory and DIsseminate Graffiti along the dOnaukanal  
PI: Geert Verhoeven, Norbert Pfeifer  
2021 - 2023

**MEDCON**

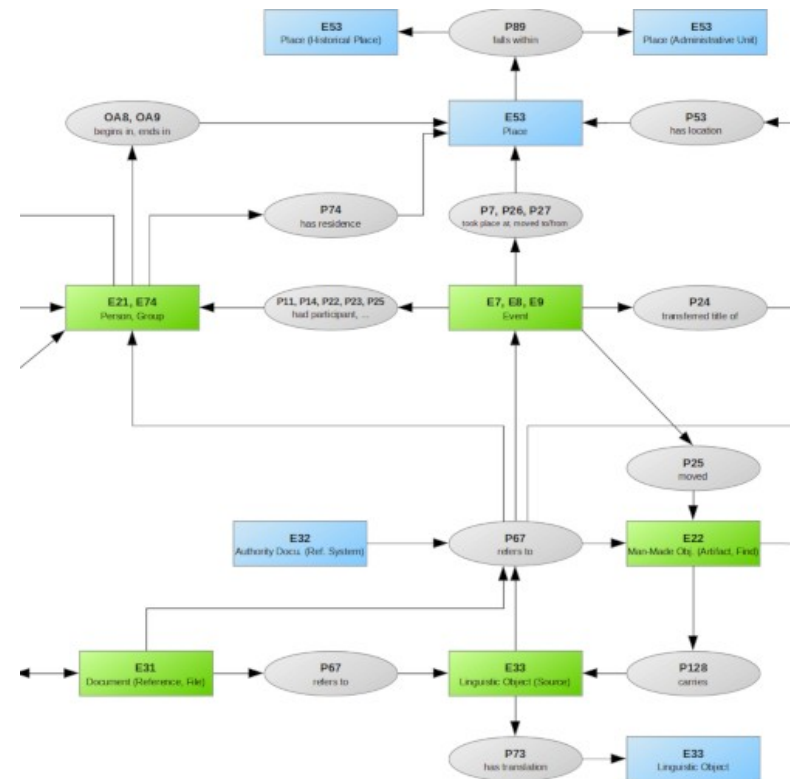
Mapping Medieval Conflict  
PI: Johannes Preiser-Kapeller  
2014 - 2017

<https://openatlas.eu/projects>

# Model

## CIDOC Conceptual Reference Model

- International standard (ISO)
- Developed by CIDOC CRM Special Interest Group
- Specifies classes for entities like actor, source, event, place and rules how to link them
- Stored in an object oriented network



<https://demo.openatlas.eu/overview/model>

# Project Data

## Data enrichment with types and links

- Standard types
- Custom types
- Value types
- Linked open data
  - Wikidata
  - GeoNames
  - Custom, e.g. Viaf, GND

The screenshot shows the 'Standard types' tab in the OpenAtlas interface. The top navigation bar includes 'Standard types', 'Custom types', 'Value types', and 'Places'. Below the navigation, a list of types is displayed: Actor actor relation, Actor function, Artifact, Bibliography, Edition, and Event. A search bar with the placeholder 'Type to search' and a '+' button is visible. The search results are grouped into categories with counts: Building activity (0), Change of Property (2, 69), Confirmation of Property (12), Conflict (5), Consecration of a church (0), Extreme event (0), Gathering (0), Mentioned (23, 5), Movement of people or goods (1, 8), Recognition of Title (0), and Wedding (0). On the right side, there is a 'Selection: single' dropdown, 'Classes: Acquisition, Activity, Move', 'Untyped entities: show', and a 'Description' section with the text: 'Categories for the type of events like Change of property, Conflict, Movement, Attendance etc.'

<https://demo-dev.openatlas.eu/types>

# API

## Application Programming Interface

- Interface to external systems
- Abstraction layer for standardized data exchange
- Can be used for
  - Presentation sites
  - GIS applications
  - Analytic applications, e.g. network visualization
  - Complex research queries

# API

## OpenAtlas API

- Web based REST-like API
- Uses the OpenAPI specification 3
- Different output formats, e.g.
  - Linked Places format (Pelagios)
  - GeoJSON
  - RDF
  - XML

```
▼{
  @context: "https://raw.githubusercontent.com/LinkedPasts/linked-places/master/linkedplaces-context-v1.1.jsonld",
  ▼features: [
    ▼{
      @id: "https://demo.openatlas.eu/entity/15888",
      crmClass: "crm:E18 Physical Thing",
      depictions: null,
      ▼description: [
        ▼{
          value: "Prague is a nice city"
        }
      ],
      ▼geometry: {
        ▼coordinates: [
          14.42876,
          50.08884
        ],
        description: "Prague (3067696), imported from GeoNames",
        title: "Prague",
        type: "Point"
      },
      ▼links: [
        ▼{
          identifier: "https://www.geonames.org/3067696",
          referenceSystem: "GeoNames",
          type: "closeMatch"
        }
      ],
      ▼names: [
        ▼{
          alias: "Praha"
        }
      ],
      ▼properties: {
        title: "Prague"
      },
      ▼relations: [
        ▼{
          label: "City",
          relationDescription: null,
          relationSystemClass: "type",
          relationTo: "https://demo.openatlas.eu/api/0.2/entity/752",
          relationType: "crm:P2 has type",
          type: null,
          when: { }
        },
        ▼{
          label: "Location of Prague",
          relationDescription: null,
          relationSystemClass: "object_location",
          relationTo: "https://demo.openatlas.eu/api/0.2/entity/15889",
          relationType: "crm:P53 has former or current location",
          when: { }
        }
      ]
    }
  ]
}
```



# API

## Links

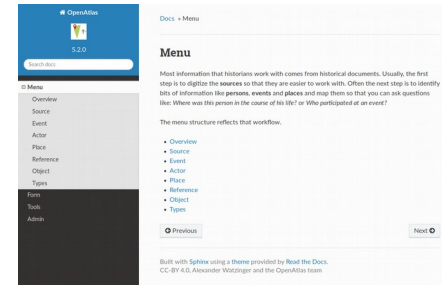
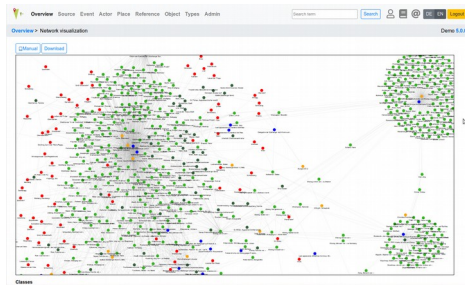
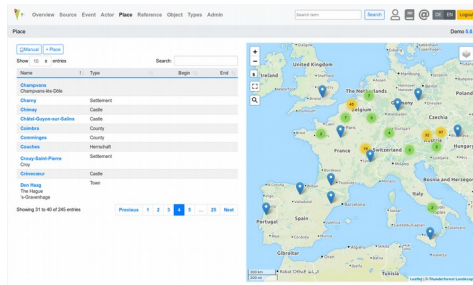
- OpenAPI
  - <https://swagger.io/specification>
- Linked Places format
  - <https://github.com/LinkedPasts/linked-places-format>
- OpenAtlas API documentation
  - <https://app.swaggerhub.com/apis/ctot-nondef/OpenAtlas/0.2>

# Development

- Active development, a new version about every month
- Technologies: Python3, Flask, PostgreSQL + PostGIS
- High coding standards
  - Bug free
  - Tested
  - Current software
- Close cooperation with the projects teams
- Transparent workflow, e.g. public
  - Code
  - Roadmap
  - Issues

# Demonstration

## Live demonstration of OpenAtlas



- OpenAtlas website ([openatlas.eu](https://openatlas.eu))
- OpenAtlas user interface ([demo.openatlas.eu](https://demo.openatlas.eu))
- OpenAtlas manual ([link](#))
- OpenAtlas issue tracker and Wiki ([link](#))
- OpenAtlas Discovery ([link](#))

# Thank you for listening



<https://openatlas.eu>