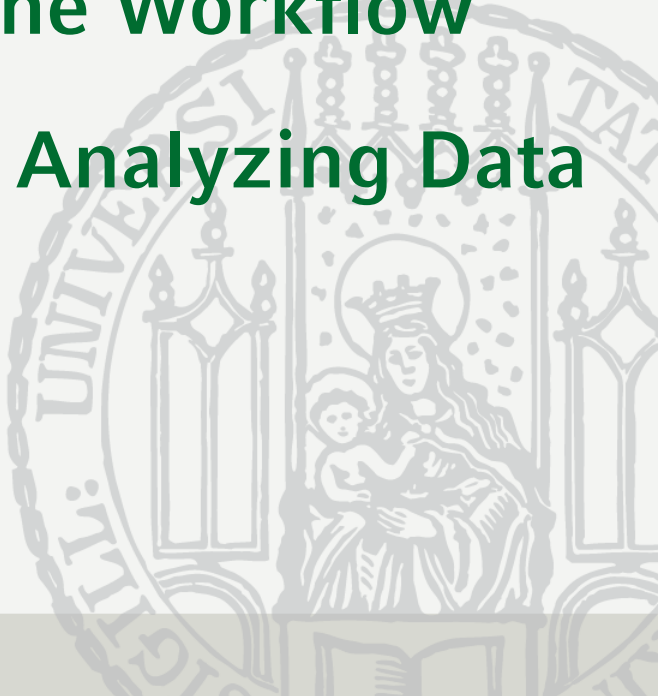
LMU LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

Ludwig-Maximilians-Universität München – Institut für Informatik –
Lehrstuhl für Datenbanksysteme und Data Mining

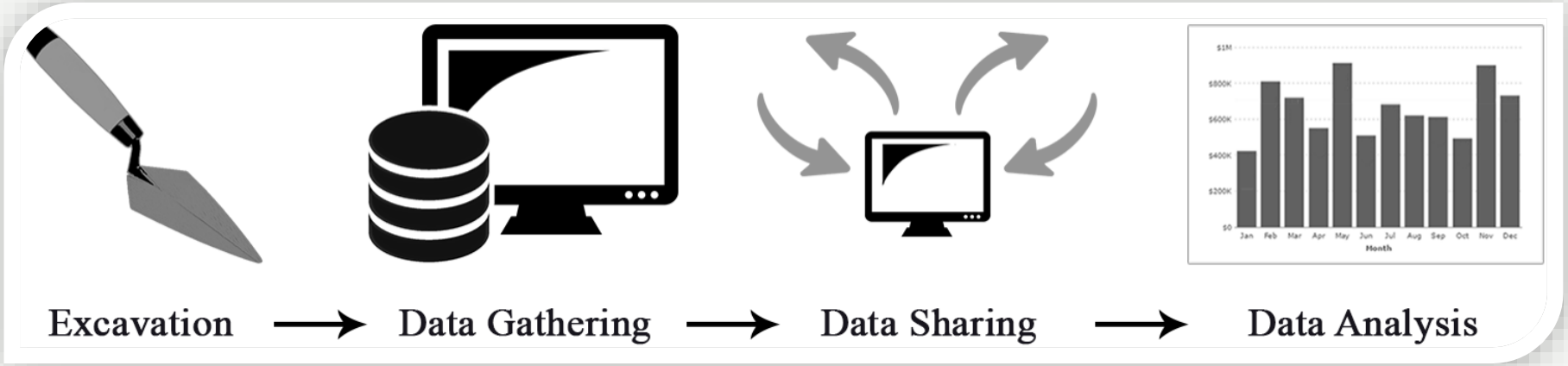
Presentation: Daniel Kaltenthaler

A Framework for Supporting the Workflow for Archaeo-related Sciences: Managing, Synchronizing and Analyzing Data

Daniel Kaltenthaler, Johannes-Y. Lohrer,
Peer Kröger, Henriette Obermaier



Workflow of archaeo-related work



1. **Data Gathering:**
   Saving of archaeo-related data digitally in a database.
2. **Data Sharing:**
   Collaborating with colleagues and sharing data with other users.
3. **Data Analysis:**
   Executing analyses on the available data.

Data Gathering (I)

Development and Implementation of the xBook Framework

Background:

- Workflow is similar in all disciplines
- Collected data is different in each special field.

Challenge:

- Provision of a generic database solution for all disciplines
- As customizable as possible to allow required information about the specific data to be gathered

→ **xBook**, a generic open-source framework including the common and basic features for a database for archaeo-related disciplines

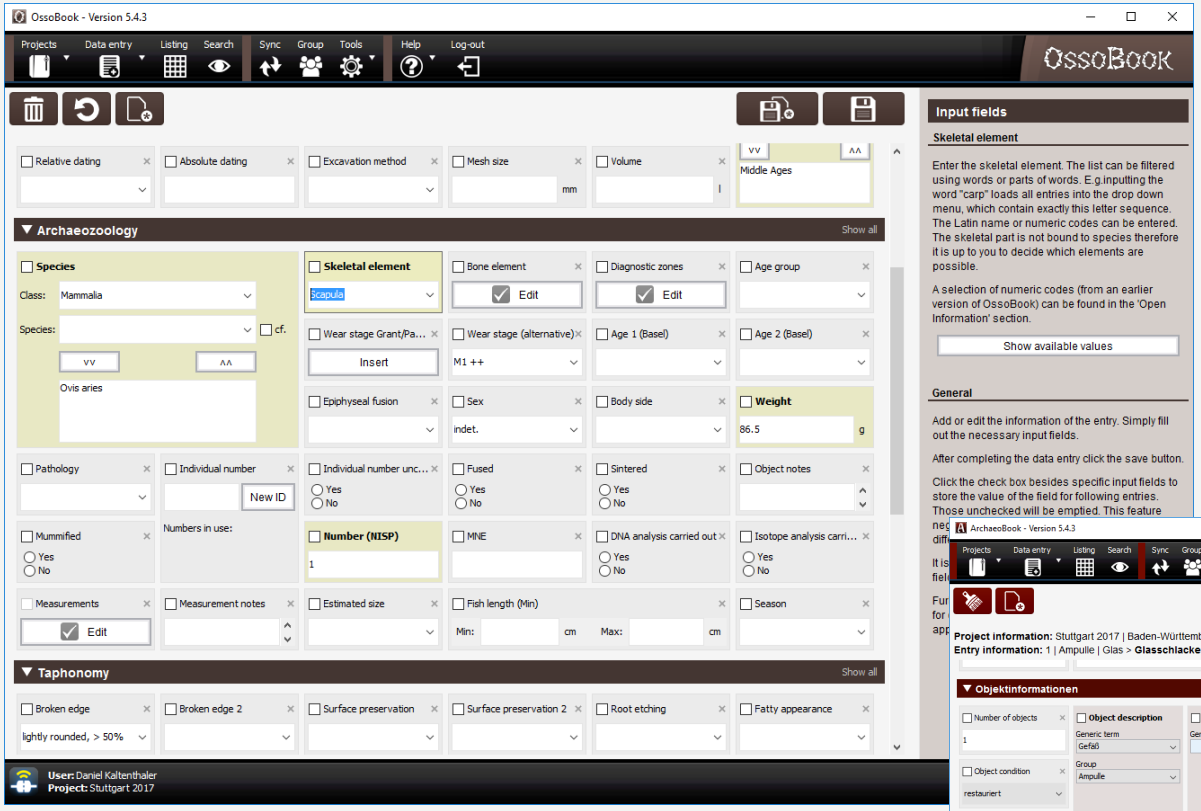
Data Gathering (II)

xBook Framework

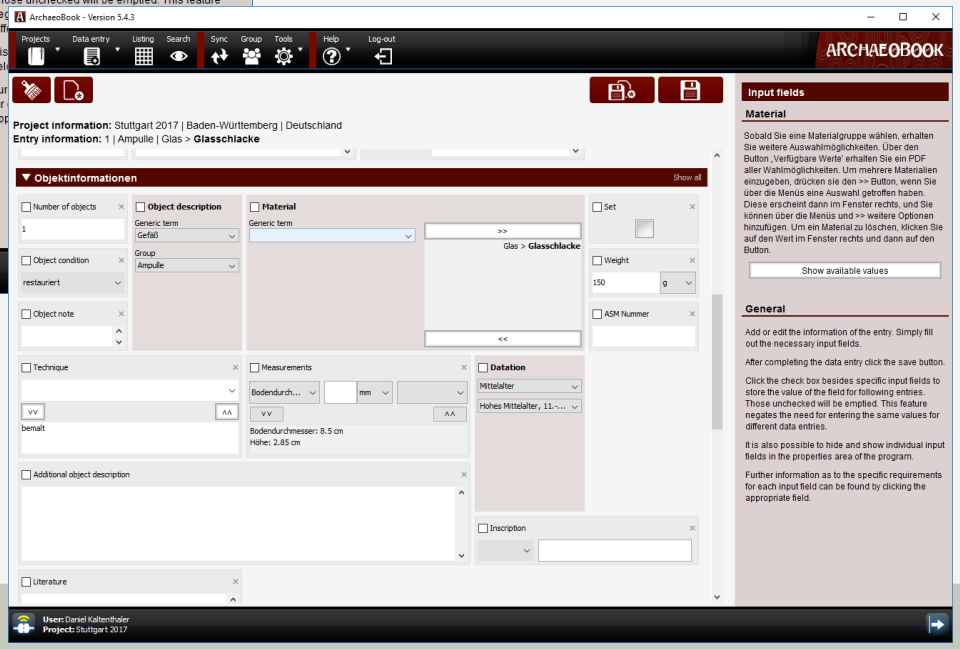
- Registration for users (**User Management**)
- Gathered data is separated to single data groupings (**Projects**)
- Dynamic and flexible **Data Entry Mask** that provides reusable input fields; custom input fields can be added to each Book
- All data can be displayed as a table form (**Data Listing**)
- **Data Search** to filter available data with specific terms or values
- Entered data can be exported to Excel Sheets / CSV (**Exporter**)
- **Project Right Management** to share data with specific users.
- **Update Function** to update program version / database scheme
- Wrapped into a **common GUI** which structure can be reused

LMU LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

Data Gathering (III)



OssoBook



ArchaeoBook

Data Sharing (I)

Synchronization of Data

Background:

- Collaborating/sharing data with others required, or storing data
- Working offline must be possible:
  – data must be synchronizable to the global server at any time
  – updated/added entries must be synchronized to the local db
- Existing synchronization methods do not fulfil all requirements

Challenge:

- Creation of a Synchronization process

Data Sharing (II)

Synchronization: Concept

- Uses timestamps to keep track of the last time the entry was updated in the global database
- Status is saved for each entry in the local database:
  - *synchronized*,
    if the entry was not updated locally
  - *updated*,
    if the entry was updated locally
  - *conflicted*,
    if a conflict was detected
    (when entry is updated on two different local databases without synchronizing the entry in between)

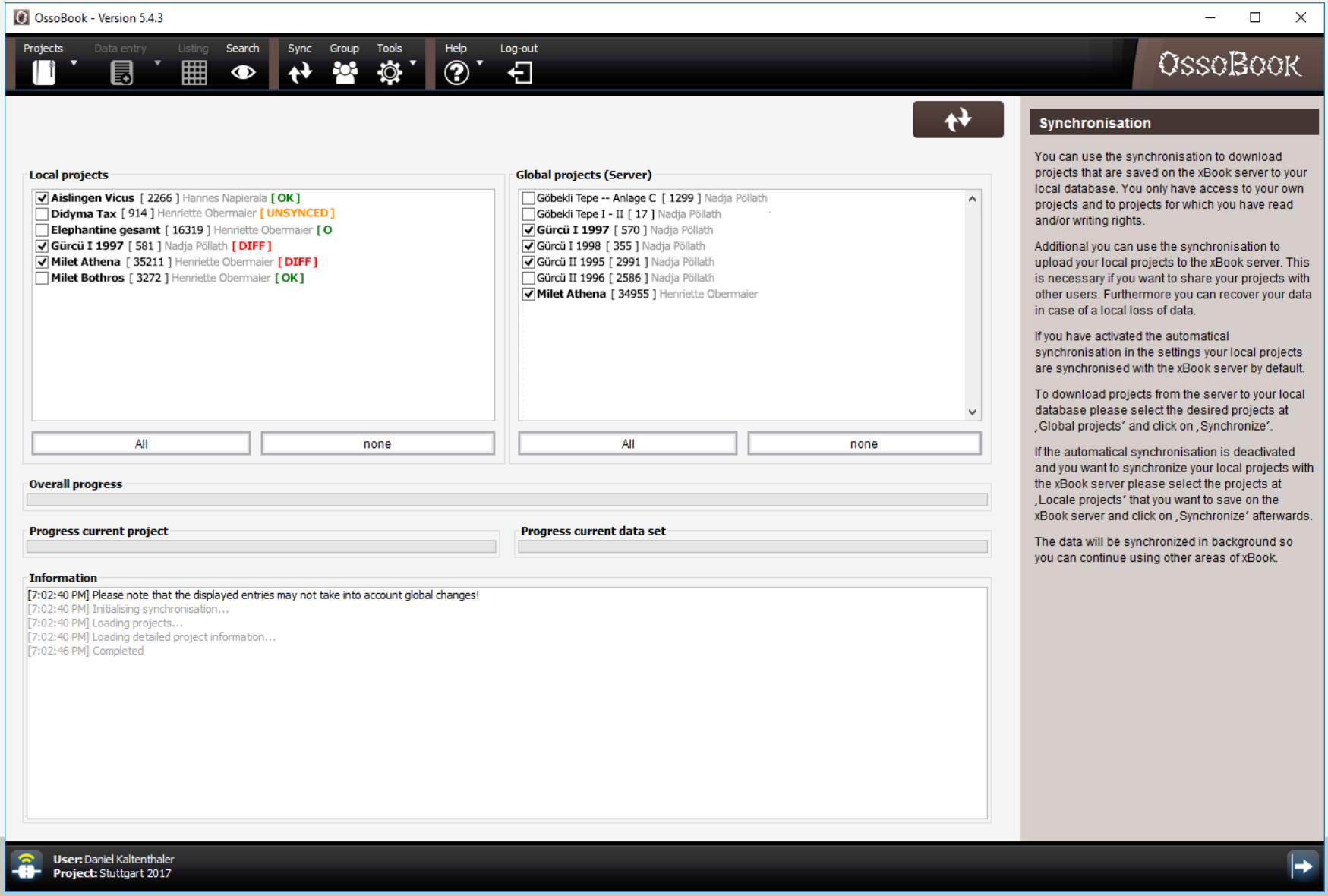
Data Sharing (III)

Synchronization: Implementation

- Each update of an entry updates sets the status to *updated*
- Start of synchronization process: each input unit is checked which entries are not synchronized
- The single unsynchronized data sets are
  – sent to the server
  – updated on the server
  – the status is set to *synchronized* locally
  (if no conflicts occur)
- A trigger on the server sets the timestamps to the current time, every time an entry is uploaded or updated
- If there is no unsynchronized entry left the entries are synchronized back to the local database

LMU LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

Data Sharing (IV)



Data Analysis (I)

Embeddable Analysis Tool

Background:

- Use of extern analysis tools is still common practise
- This method is aggravating, time consuming, error-prone

Challenge:

- Providing a dynamic, flexible, and powerful tool that allows specific queries from (archaeological) databases without having any prior knowledge of programming
- Must be embeddable to existing Java applications

→ Development of the **Analysis Tool**

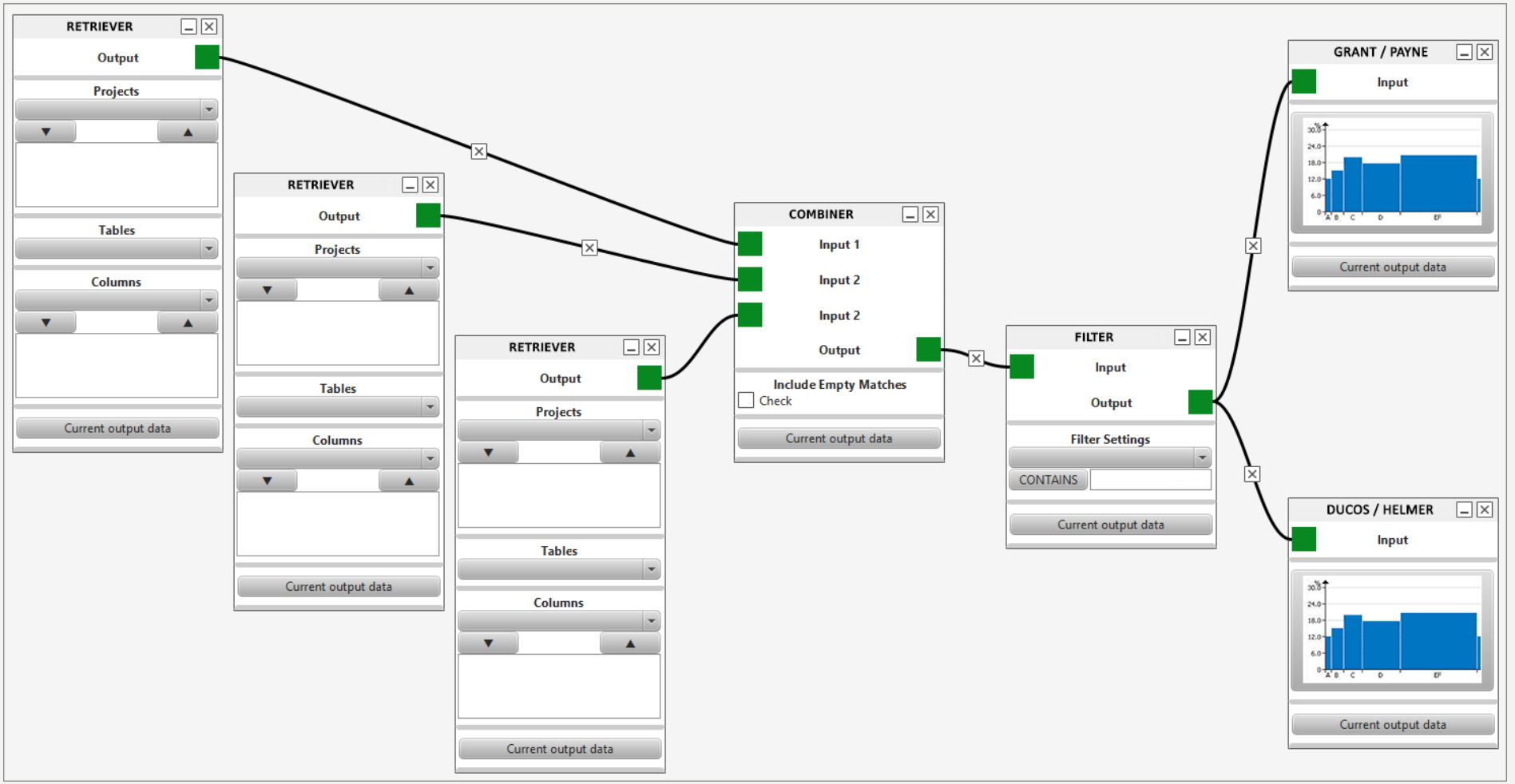
Data Analysis (II)

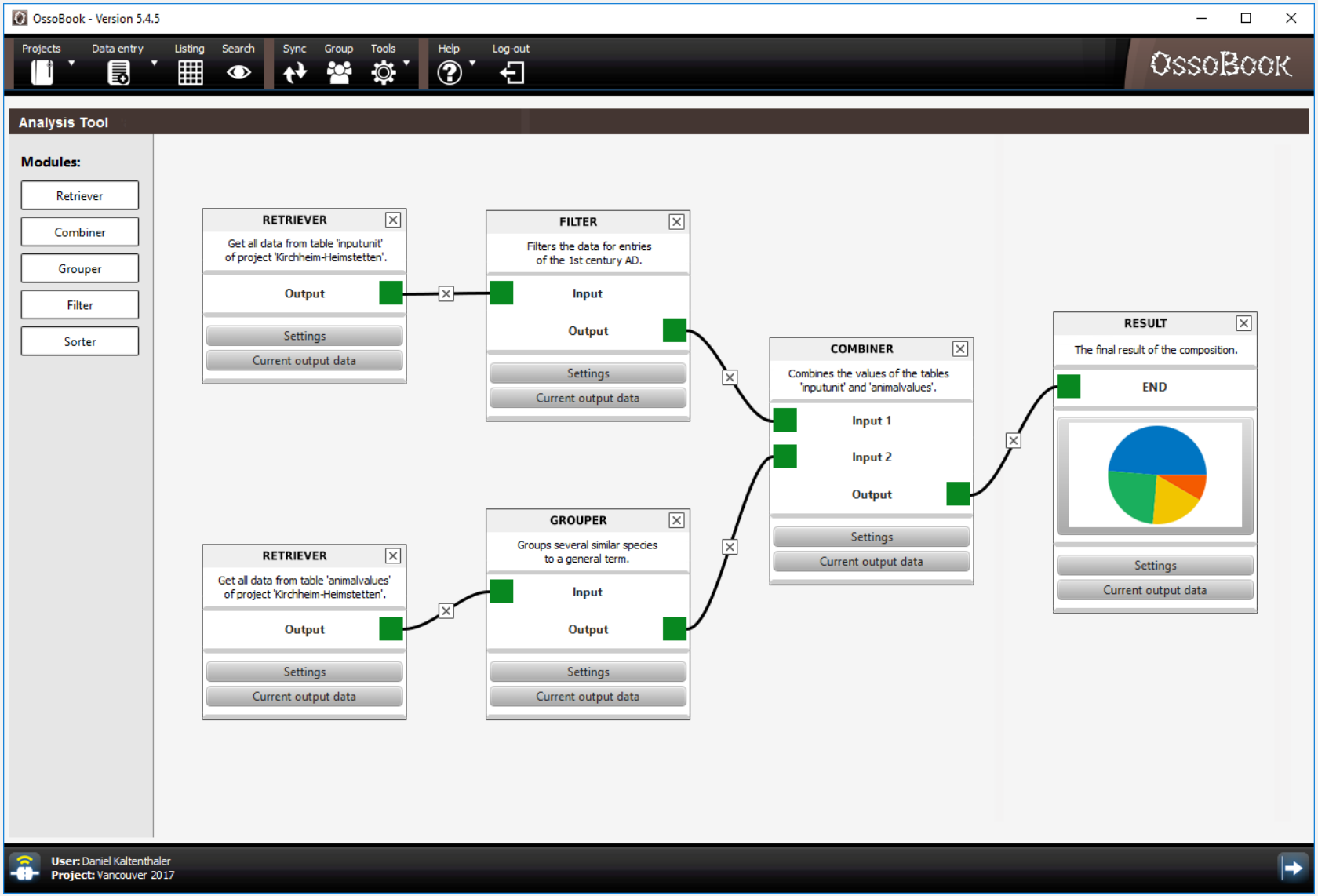
Analysis Tool: Concept

- Some functions have to be implemented by the application (e.g. the Analysis Tool cannot know how the data is stored, or how the connection to the data is realized)
- Provides a set of standard components to handle the data
- API to allow new specific components to be registered to an application
- Components have inputs/output – can be connected by the user
- Once a composition is completed, the result can be calculated and be viewed in tables or diagrams
- Generic communication between the components (connected components are not known beforehand)

Data Analysis (III)

Example: Simplified composition of the age distribution of animals





Example: Percentage of the most important livestock in settlements of the Heimstettener Gruppe (Heimstetten, Germany) in the 1st century AD, based on the number of bones

Example: Percentage of the most important livestock in settlements of the Heimstettener Gruppe (Heimstetten, Germany) in the 1st century AD, based on the number of bones

Discussion

Synchronization:

- Improvements of the performance necessary, e.g.
  – using compressing methods for the transmitted data
  – using special data type to decrease the size of the data
- Solution to be able to delete the file sets in the database

Analysis Tool:

- Currently an early stage of development!
- Creation of further Workers would extend the possibilities (especially Workers for predefined calculations)
- Frequent used combinations of single Workers may be integrated as own Workers
- Number of provided diagrams for the graphical representation
- Possibility to save, export, and share a composition

Servus!

Thank you for your attention!

Questions…?