



xBook, a Framework for Common Scientific Databases

Daniel Kaltenthaler¹(✉), Johannes-Y. Lohrer¹, Peer Kröger¹,
and Henriette Obermaier²

¹ Institut für Informatik, Ludwig-Maximilians-Universität München,
Munich, Germany

{kaltenthaler,lohner,kroeger}@dbs.ifi.lmu.de

² Bavarian State Collection for Anthropology and Palaeoanatomy, Munich, Germany
henriette.obermaier@palaeo.vetmed.uni-muenchen.de

Abstract. In this paper, we introduce xBOOK, an open-source framework that is designed for creating databases for heterogeneous scientific disciplines. Gathering and recording data is an essential part of scientific work. However, the challenges of the workflow are similar in the different fields. We want to support scientists in their work by offering a flexible and scalable framework that is intended to provide common functionalities to the inherited applications.

Keywords: Data analysis · Database · Features · Framework
Launcher · Synchronization · xBook

1 Introduction

Collecting and sharing data is a frequent task in many scientific areas. Although each discipline has its own workflows and practices, the basic requirements are similar in many cases. Very often the gathered data has to be stored in a relational database, but an application to enter, search, exchange, analyze, and manage data, is also required to fulfill the scientific tasks. A common database framework solution, which is independent of the disciplinary field of work, could provide a general infrastructure to scientists that provides the widely used features.

Therefore, we implemented the xBOOK framework which provides built-in methods that follow the scientific workflow. Gathering and storing data to the database is enabled by dynamic input masks that can be composed by a wide range of predefined input fields. Custom input fields can also be integrated. The synchronization process, including a conflict management solution, allows the collaboration with colleagues, sharing of data with other users, and the possibility to work offline and synchronize data later. This process considers a flexible user and group rights management. The built-in Analysis Tool provides possibilities to execute analysis requests without special programming knowledge. Additionally, xBOOK also provides further features for data overview, filtering

and search options, automatic application update processes, export and import possibilities, etc. Domain-specific features can also be implemented and added to any database. All features are integrated in a flexible, user-friendly graphical user interface that can be re-used and customized for each database application.

In this paper, we give an overview of the xBOOK framework and its features with four real examples of inherited applications from the archaeo-related domain.

2 Features

xBOOK is a generic open-source framework including the common and basic features for a database for scientific disciplines. Our main priority is to integrate – if possible – all features into the xBOOK framework to provide the available development to all “Books” (instances of xBOOK). Because each Book is built upon the xBOOK framework, all other Books also profit from this new feature. Each Book provides these common features: [1]

Collaboration Methods: For a database framework, which aims for the collaboration of different areas and scientists, it is important to offer different aspects that enable collaboration. The basis for the solution in xBOOK is the implementation of a synchronization which allows to share, to back-up, and to collaborate on specific data sets [2–4]. A synchronization method is especially required in sciences that gather their data directly in field work. Working on a global database is not always possible due to the lack of reliable Internet connections.

In xBOOK, data sets are organized as projects which each can be synchronized and shared. The users get displayed the list of projects in the synchronization panel that exist in the local database and the list of projects that exist on the server, as can be seen in Fig. 1 (left). From there, the projects to be synchronized can be chosen.

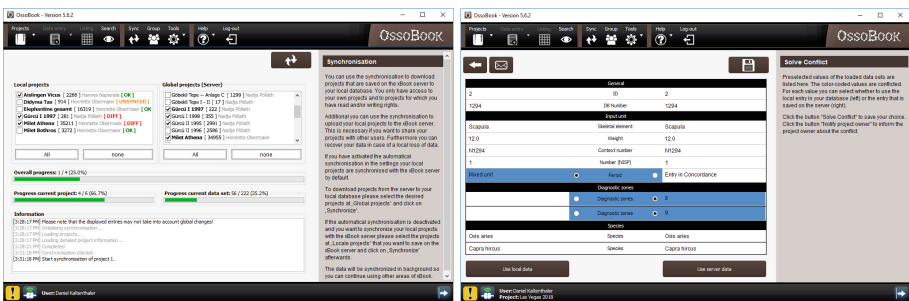


Fig. 1. Screenshots of the synchronization panel (left) and the conflict management panel (right) in Ossobook.

The synchronization uses timestamps on the global server to keep track of changes. Entries, that were edited on the clients, are marked with an indicator to keep track of entries that have to be synchronized. During the synchronization first all locally edited entries are sent to the server. Then, new or edited entries are synchronized back to the client in order of the timestamp of the entries on the server. A conflict might occur during the synchronization process if the timestamp of the synchronized entry is lower (older) than the timestamp on the server. Therefore, the user was not working on the most recent version of the entry as it has been edited on the server since the last execution of the synchronization. This conflict has to be solved by the user. The user can select whether to use the values that are saved on the client or the values that are saved in the local database in the conflict management screen, as can be seen in Fig. 1 (right). When the conflict of the entry is solved, the entry is updated on the server.

Additionally, a rights management is necessary to manage the access permissions to the data sets. The project owner authorizes other users to access or edit specific parts of the data. Since often the same work group of scientists (e.g. companies or within an institute) is working together, xBOOK also provides the definition of groups. This is necessary to avoid having to define and manage the rights for each single scientist of this group individually. The project owner can gain permissions to this group and the group owner can delegate the available rights within this group.

Analysis Methods: Analyzing data is usually the most important step of scientific work, regardless of the field of work. However, scientists may be specialized on other methods and may lack how to handle certain analysis tools or software which are necessary to create complex analyses. In addition, the export of data to spreadsheet or CSV files to use it in external applications can be aggravating, time-consuming, and error-prone.

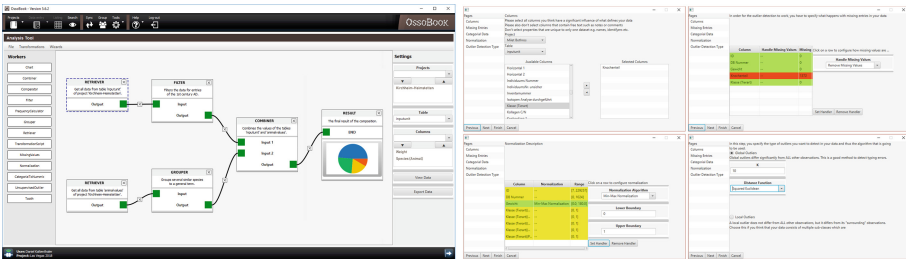


Fig. 2. Screenshots of the ANALYSIS TOOL which is embedded to OSSOBOOK (left) and the Outlier Detection Wizard [7] (right).

Therefore, we developed the ANALYSIS TOOL [5,6] and embedded it into xBOOK (cf. Fig. 2 left) to provide analysis methods for the scientific data,

without the requirement to first export and then import the data into another application. It is a flexible tool that provides a visual querying the data of the database scientists are used to work with. The tool is connected to the base application over a simple interface which defines the columns that can be used for the analyses. To composite an analysis the users can use independent modules called ‘Workers’ which each fulfill a specific task, e.g. retrieve, filter, sort, merge, or visualize data. Workers have inputs and outputs which can be connected to create the desired analyses.

By connecting the data output of one Worker with the data input of another Worker, a directed graph or multigraph is formed. The Workers in this graph are calculated, starting from the front Workers. The composition of the analysis can conclude in either an (exportable) table of data or a graphical representation in a diagram. This visual approach of the ANALYSIS TOOL enables the querying, analysis, and statistical evaluation of data, e.g. an outlier detection [7] (cf. Fig. 2 right), in the familiar user environment without the user having in-depth programming skills.

Graphical User Interface: xBOOK provides a flexible graphical user interface that consists of a basic, graphical structure which is used by each Book by default. These are necessary because all Books use the same logical structures, for example the login screen, the synchronization process, the settings, etc. However, these consist of dynamic elements that can individually be adjusted for each instance of xBOOK by override the abstract methods.

This way it is possible that each Book can have different input fields in the input mask, like indicated in Fig. 3. The xBOOK framework provides reusable input fields that contain the logic for handling and displaying the data. Each Book can integrate them into their input mask by defining their individual settings adjusted to the database, like the table or column where to store the data. Additionally, individual input fields can be implemented, the input masks are not limited to the predefined input fields. So each Book can have a complete different, individual structure of the input mask with diverse input fields that all consider the same program logic.

Similar to the input fields, it is also possible to customize the other elements of the graphical representation, for example to define the information displayed in the project overview screen, the general style of the Book (like the logo, the name of the Book, colors, etc.), the location where to save settings, the individual output for the export, etc. The navigation and sub navigation is also extendible to allow to integrate individual navigation elements for custom features.

This flexible structure has the advantage that all Books can reuse features and functions that have already been implemented before, either for the xBOOK framework or individual for another Book. This saves time and resources, so that necessary elements do not have to be implemented again and again for different applications.

xBOOK Launcher: The xBOOK LAUNCHER (cf. Fig. 4) is the central application for the users to manage all Books. Since there are multiple instances of xBOOK we needed a stand-alone application that is independent of the Books, but allows to add each available Book.

This is done by entering the Book-specific URL in the xBOOK LAUNCHER, then the Book will automatically be installed and can be executed. Existing Books are checked for updates and – if they are not up-to-date – be updated automatically to the current application version. Further update processes, to update the database scheme to the current version and to update the value tables used in the input fields of the Book, are integrated to xBOOK.

Further Features: Furthermore, the xBOOK framework provides several features that are required in scientific work. The stored data of the database is meaningfully displayed in a sortable data listing, considering that the values are human-readable. The framework provides an export method that saves the data in a CSV- or spreadsheet (XLS/XLSX) file on the local system. Both, the listing and the export, can be filtered by the user. The available projects can be searched for specific entries. A dynamic data importer is in development.

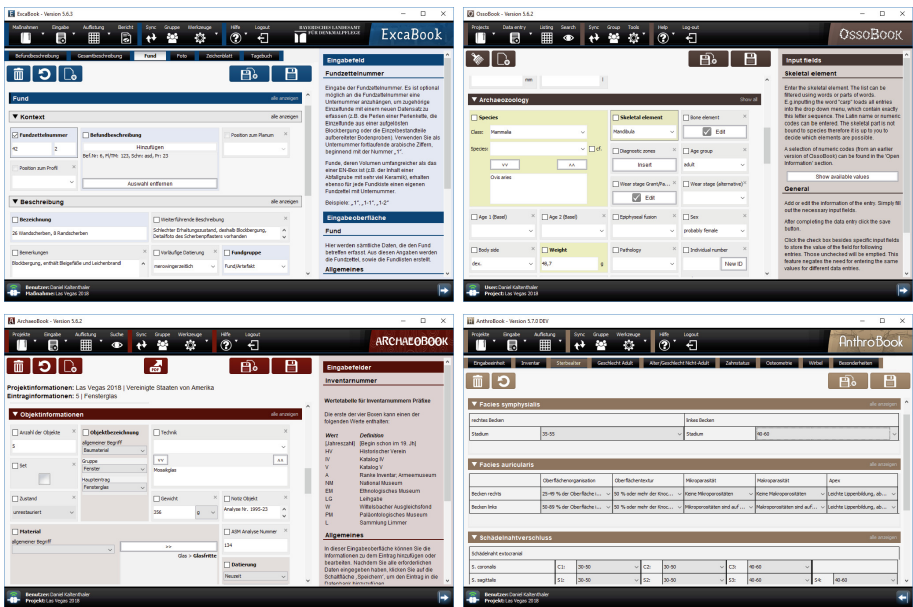


Fig. 3. Screenshots of the input masks of EXCABOOK (top-left), OSSOBOOK (top-right), ARCHAEOBOOK (bottom-left), and ANTHROBOOK (bottom-right).

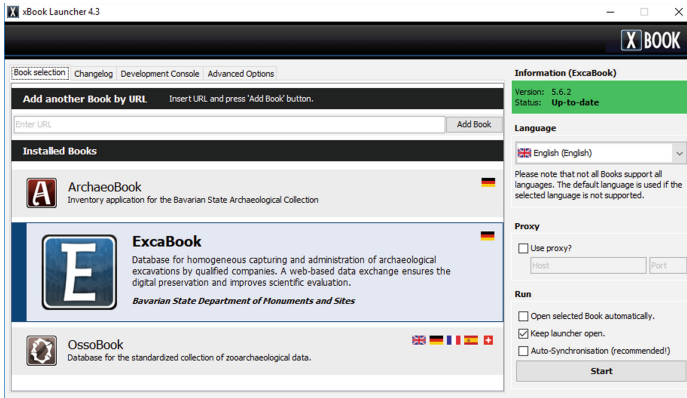


Fig. 4. Screenshot of the xBOOK LAUNCHER.

3 Use Case: Databases in the Archaeo-Related Disciplines

In this section, we introduce four different databases from the archaeo-related domains that are based on the xBOOK framework. We first briefly describe each of the applications, then we describe the advantages that xBOOK provides them.

Archaeo-Related Databases: Examples of xBook Instances

Below, we introduce the available, archaeo-related instances of xBOOK.

EXCABOOK [8] is an archaeological database for the documentation of excavations. It is used by the *Bavarian State Department of Monuments and Sites*¹ for the homogeneous capturing and administration of archaeological excavations by qualified companies. The state department creates new projects for the excavations and give the excavation companies access to the project. After the excavation is completed, they enter the excavation data into the database and transmit the data back to the state office by using the built-in synchronization method. Then, the state office continues internally administrating the data.

OSSOBOOK [9] is a scientific database for the standardized collection of zooarchaeological data, for long-term archival storage, for data exchange, and serves as a basis for scientific analyses. It is currently used by approximately 350 users including scientists, Ph.D. students and students in institutes of universities, museums, and scientific collections with zooarchaeology as field of work as well as freelance zooarchaeologists. The synchronization method enables collaboration and the sharing of data. OSSOBOOK is operated by the *Bavarian State Collection for Anthropology and Palaeoanatomy Munich*² and the *Institute of Palaeoanatomy, Domestication Research and History of Veterinary Medicine*³ of the *Ludwig-Maximilians-Universität München*.

¹ www.blfd.bayern.de.

² www.sapm.mwn.de.

³ www.palaeo.vetmed.uni-muenchen.de.

ARCHAEOBOOK [10] is an inventory database for the archaeological findings of the *Bavarian State Archaeological Collection*⁴. The information and characteristics of the objects in the collection and the museum are saved in the database. It is also used to track the current location of the objects within the collection and to log loans of findings.

ANTHROBOOK [11] is a scientific database that is currently developed for the standardized collection of anthropological data, operated by the *Bavarian State Collection for Anthropology and Palaeoanatomy Munich*². Similar to OSSOBOOK, ANTHROBOOK is planned for long-term archival storage, for data exchange, and as a basis for scientific analyses.

Further instances of xBOOK for the archaeo-related sciences includes the inventory databases ANTHRODEPOT, PALAODEPOT, and INBOOK.

Benefits for xBook Instances

The mentioned archaeo-related databases each focus on a different scenario: For administrating and retrieving homogeneous data (EXCABOOK), for scientific usage (OSSOBOOK and ANTHROBOOK), and for inventory reasons (ARCHAEOBOOK). However, they all benefit from the development of the xBOOK framework. xBOOK provides the features that were introduced in Sect. 2 that can be used by each database without having to implement an own solution.

The built-in synchronization method is flexible in use, each of the database use it for another purpose. While EXCABOOK uses the synchronization to send and retrieve data to/from the excavation companies, OSSOBOOK and ANTHROBOOK requires the possibility to store data offline directly in the field and to synchronize the data later to share it with colleagues or other scientists, and ARCHAEOBOOK use the method for house-internal collaboration.

Since xBOOK already provides the most basic structures, the developers of the databases could focus on the integration of the necessary input fields, without having to concern about how to save and load the data from the database and how to design the graphical representation. They benefit from a common graphical user interface and program logic which enables a fast development of a well-functioning application. Of course, customizations of the existing elements and the addition of new elements are still possible. Also the structure of the input masks are flexible for different uses. While OSSOBOOK and ARCHAEOBOOK each have one single input mask, EXCABOOK uses several different input masks that are cross-linked among each other and ANTHROBOOK manages one entry in several separates input masks.

Additionally, the workflow of a database application is comparable in the different archaeo-related disciplines – so the requirements for a database application are also similar. For example, all archaeo-related disciplines need a data export for long-term archiving or for standardized analysis tools. Since xBOOK provides an export feature the developers do not have to implement a new export method for each database. The same applies to the graphical user interface, user management methods, settings, etc.

⁴ www.archaeologie-bayern.de.

Creating all these features individually for each single database application would take time and therefore could be expensive even if single features are small. Also bugs and errors have only to be fixed once in the framework instead of in each single application. Especially financially weak sciences benefit from saving money by using commonly developed frameworks like xBOOK.

4 Availability

The xBOOK LAUNCHER is available from <http://xbook.vetmed.uni-muenchen.de>. From there, OSSOBOOK, ARCHAEOBOOK, and EXCABOOK are publicly available.

References

1. Kaltenthaler, D., Lohrer, J.-Y.: The Historic Development of the Zooarchaeological Database OssoBook and the xBook Framework for Scientific Databases. ArXiv e-prints: [arXiv:1801.08052](https://arxiv.org/abs/1801.08052) (2018)
2. Lohrer, J.-Y., Kaltenthaler, D., Kröger, P., van der Meijden, C., Obermaier, H.: A generic framework for synchronized distributed data management in archaeological related disciplines. In: 10th IEEE International Conference on e-Science, eScience 2014, São Paulo, Brazil, 20–24 October 2014, pp. 5–12 (2014)
3. Kaltenthaler, D., Lohrer, J.-Y., Kröger, P., van der Meijden, C., Obermaier, H.: Synchronized data management and its integration into a graphical user interface for archaeological related disciplines. In: Design, User Experience, and Usability: Users and Interactions—4th International Conference, DUXU 2015, Held as Part of HCI International 2015, Proceedings of Part II, Los Angeles, CA, USA, 2-7 August 2015, pp. 317–329 (2015)
4. Lohrer, J.-Y., Kaltenthaler, D., Kröger, P., van der Meijden, C., Obermaier, H.: A generic framework for synchronized distributed data management in archaeological related disciplines. *Future Gener. Comput. Syst.* **56**, 558–570 (2016)
5. Lohrer, J.-Y., Kaltenthaler, D., Kröger, P.: Leveraging data analysis for domain experts: an embeddable framework for basic data science tasks. In: 7th International Conference on Internet Technologies & Society 2016, Melbourne, Australia, pp. 51–58 (2016)
6. Kaltenthaler, D., Lohrer, J.-Y., Kröger, P.: Supporting domain experts understanding their data: a visual framework for assembling high-level analysis processes. In: 11th International Conference on Interfaces and Human Computer Interaction 2017, Lisbon, Portugal, pp. 217–221 (2017)
7. Blau, J.: Interactive outlier detection based on archaeological data. Bachelor thesis, Ludwig-Maximilians-Universität München, Germany (2018, to appear)
8. Kaltenthaler, D., Lohrer, J.-Y., Kröger, P.; van der Meijden, Jantos, S., Rahm, A., Sassen, I., Wanke, T., Wanninger, R., Haberstroh, J., Sommer, S.: ExcaBook v5.6.2. [Computer Software], Munich, Germany (2018)
9. Kaltenthaler, D., Lohrer, J.-Y., Kröger, P.; van der Meijden, C., Granado, E., Lamprecht, J., Nücke, F., Obermaier, H., Stopp, B., Baly, I., Callou, C., Gourichon, L., Peters, J., Pöllath, N., Schiebler, J.: OssoBook v5.6.2. [Computer Software], Munich, Germany; Basel, Switzerland (2018)

10. Kaltenthaler, D., Lohrer, J.-Y., Kröger, P; van der Meijden, Harrington, C., Claßen, E., Gebhard, R., Marzinzik, S., Schwarzberg, H.: ArchaeoBook v5.6.2. [Computer Software], Munich, Germany (2018)
11. Kaltenthaler, D., Lohrer, J.-Y., Kröger, P; van der Meijden, Mösch, A., Sizova, T., Harbeck, M., Toncala, A.: AnthroBook Development Version [Computer Software], Munich, Germany (2018)