DiversityMobile – Mobile Data Retrieval Platform for Biodiversity Research Projects

Stefan Jablonski[1], Alexandra Kehl[2], Dieter Neubacher[3], Peter Poschlod[4], Gerhard Rambold[2], Tobias Schneider[1], Dagmar Triebel[3], Bernhard Volz[1], Markus Weiss[3]

[1]Applied Computer Science IV
[2] DNA Analytics and Ecoinformatics Laboratory
University of Bayreuth, Bayreuth, Germany
{stefan.jablonski, alexandra kehl, gerhard.rambold, tobias.schneider, bernhard.volz}@uni-bayreuth.de

[3]IT Center of the Bavarian Natural History Collections
Munich, Germany
{neubacher,triebel,weiss}@bsm.mwn.de

[4]Institute for Botany
University of Regensburg, Regensburg, Germany
peter.poschlod@biologie.uni-regensburg.de

Abstract: A majority of biodiversity research projects depend on field recording and ecology data. Therefore it is important to provide a seamless and transparent data flow from the field to the data storage systems and networks. Seamless in the sense, that data are available shortly after their gathering, transparent in the sense that the history of data operations may be traced backward. *DiversityMobile* (with the complementing applications of the Diversity Workbench frameworks) is a GUI software that provides the option of gathering biological and ecological research data in a structured way by using mobile devices for data retrieval.

1 Introduction

With the development of scientific data networks for biological and ecological research projects the need for a seamless and transparent flow of data became increasingly important within the last ten years. Especially the establishment of international web portals such as the Global Biodiversity Information Facility (GBIF, http://www.gbif.org), Species2000 (http://www.species2000.org) or the Encyclopedia of Life (http://www.eol.org/) required standards and guidelines set up by politics, and, in parallel, augmented pressure on scientists to provide their primary research data in appropriate interchangeable formats. Nevertheless, until now, biodiversity field data have been recorded manually with pen and paper, followed by subsequent transfer to a computer based data storage system.

During the process of data transcription and data acquisition several general problems arise and finally constrain the access to high-quality ecological data gained in field research projects:

- Transcribing handwritten data lists is an error-prone task. For instance, paper sheets can become unreadable under rough conditions.
- Typos can occur during the transcription of data into a digital representation.
- In many cases, more data are retrieved in the field than are required for a specific research project that forms the context of the data retrieval. In that case, the amount of data fed into a computer-based storage system, is often reduced to the minimal extent in order to reduce the required effort of time. The parts of the data which are not transferred to a digital form are lost in the sense that these data cannot be shared with other researchers with interest into other aspects.
- During the transcription process data is often manipulated by standardizing or cleansing operations, for instance, units are converted or single measurements are accumulated. Nevertheless, procedures of keeping track of such changes and logging why the changes were necessary are neglected in most cases.
- Multidimensional complex relationships (temporal, spatial) between biological or ecological entities often cannot be mapped accurately onto paper sheets in the field as background data is not accessible. For this reason, references to such data will have to be added retroactively to the user interface.

Based on these experiences, certain requirements are to be fulfilled to make an application suitable for successfully supporting scientists in data retrieving in the field:

(i) The central requirement for the application is that it should run on a mobile device, i.e. a device which is small enough to fit into a scientists pocket, does not weight too much and is appropriate to be used under rough conditions.
(ii) The tool should assist scientists during data retrieval and cost no additional time.
(iii) All entered research data are to be transferred into a central data storage system for management and further processing. Any operation enacted on the data during the transfer must be logged and made available for later review.
(iv) Research data which were collected during previous surveys must be made accessible, in order to establish multidimensional data relations directly in the field.
(v) Direct access to specific databases, for instance, taxonomic reference lists or thesauri of scientific terms, support the naming and characterization of objects by providing defined descriptors. Nevertheless, it should be possible as well to postpone this procedure, especially the identification of an item, until a later point in time.
(vi) Multimedia data such as images, video, audio recording or position data (for instance, GPS coordinates) should be integrated if the mobile device supports these capabilities.

If these requirements are met, the flow of data starting in the field with the mobile device and ending at large databases with biological data can be made more efficiently, i.e. faster and safer. The mobile application used for gathering data is an essential part of this chain.

A prototype application for mobile data retrieval has recently been set up and will be extended within a research project named "Setting up an **I**nformation Network on **B**iological Research Data gained in the **F**ield up to the Sustainable Storage in a Primary Data Repository (**I-B-F**)" [Tr09]. The IBF project is based on the concept and development of the *Diversity Workbench* framework (http://www.diversityworkbench.net) and is going to expand the platform with a client software *DiversityMobile* for smartphones or personal digital assistants (PDAs). The approach is to gather biological research data in the field by usage of a PDA with GPS functionality, a digital imagery option and a microphone. Several flexible user interfaces are set up to assist biodiversity scientists, ecologists on-site, and non-professional experts to gather and store complex biological data already in the field. The user interface of the PDA will be interchangeable and give access to taxonomic names, ecological descriptors, and general scientific term presets, and allows for selecting reference points in digitized topographic maps. The gathered data are transferred to a data repository at the IT Center of the Bavarian Natural History Collections (SNSB) via data synchronization between the involved databases and are redistributed to the end-users via various types of interfaces and web services. Besides this integrated infrastructure, the data repository hosts schemata and generic interfaces for the data exchange with smartphones and PDAs, wrappers (ABCD schema) and external applications for data analysis and presentation. The complete dataflow and virtual working environment is built up in cooperation between four research groups in biology and informatics and is documented on the internet (http://www.diversitymobile.net). In the course of the development new strategies of complex data access and structuring will be modeled and tested. This concerns especially more-dimensional interrelations between organisms in a temporal and spatial context.

In its current prototypical implementation *DiversityMobile* covers most of the requirements identified above. The context of the development project, the architecture of the mobile data retrieval platform and its practice within an ecological project is discussed below.

2 Related Work

The IBF project started building up a mechanism for mobile data retrieval using *DiversityMobile* and a platform for data transfer towards a repository database system. Currently, the project cooperates with two work groups which represent two major research communities gathering biodiversity monitoring and ecology data in the field. Organisms to be recorded include fungi, lichens, plants and gall-inducing insects.

As far we know about only few other projects, being focused on similar tasks and searching for solutions for data flow from the field by using mobile device technology. Preliminary conceptual studies on this subject are undertaken as part of the EU project EDIT (Workpackage 7) – All Taxa Biodiversity Inventories [KMH09].

Furthermore, the use of laptops and notebooks with access to local project-specific databases is tested by several groups of monitoring experts in the field (see FLORKART for bryophytes and lichens: [SF06]). The majority of biologists, however, are working without digitalizing their data in the field but store the data later in a local database system on their PC. Until now, mostly standard applications like spreadsheet MS Excel (Microsoft® Inc.) or Filemaker Pro (Filemaker® Inc.) were used as storage systems and are adapted to the specific requirements by including biological standard descriptors (see PILZOEK for habitat and ecology information on macrofungi [BDA07] and TurboVeg for vegetation data [HS01])). Other scientists rely on a centralized monitoring database with a specific web interface (e.g. Pilzkartierung online; see http://brd.pilzkartierung.de/). Some of the database systems in use are technically advanced, for instance by including GIS functionality and being based on a client/ server architecture and include collection, observation and occurrences data. Such systems used in Germany and adjacent countries are, for example, Recorder 6 D (http://netphyd.floraweb.de/?q=node/55), BioOffice (http://www.biooffice.at/index.php/ startseite.html) and *DiversityCollection* with its associated database components (http://www.diversityworkbench.net). Although these systems are customized for gaining and analysing observation data, they were not appropriate for smartphone-based data gathering.

A completely different aspect is the identification of organisms in the field with a smartphone. There are certain projects and initiatives for constructing tools for the usage of citizens and in educational institutions, for instance, as part of the EU project KeyToNature ([Ad09]; http://www.keytonature.eu/wiki/Main_Page). Another technically elaborated approach is that of the mobile application EcoPod ([Yu06];[Ma08]) supporting scientists and amateurs during the identification of single biological objects in the field. It is highly specialized with respect to finding a correct identification but does not contain the ability to include location data, images and custom fields. EcoPod instead supports taking "notes" which are then to be transcribed later into structured data. The development of a mobile version of *NaviKey* for interactive identification of organisms in the field is advised (http://www.navikey.net).

3 *DiversityMobile*

In the following, the architecture of the *DiversityMobile* prototype is explained. As *DiversityMobile* is currently under development not all features desired have been implemented yet. However, first versions of the mobile client and the synchronization framework are already running and show that the basic concepts work well. In the course of the IBF project, new requirements will be identified within the next few months. Hence, the concepts are designed in a way which allows the inclusion of new features easily.

3.1 Technical Setting for *DiversityMobile*

DiversityMobile is implemented as a *C#* application and targets the Microsoft *.NET Compact Edition* Framework. Currently many normal smartphones as well as many ruggedized mobile devices running Windows Mobile 5 or above support this runtime environment. This decision was based on the fact that the hardware platform of Windows Mobile is rather stable, i.e. not many differences exist in between the devices of different vendors with respect to memory space and processing power. Especially the latter is important since *DiversityMobile* uses in-memory database management systems such as the *Microsoft SQL Server Compact Edition* via standardized *.NET* interfaces. Thus, the most central requirement (i) of the introduction is fulfilled since besides normal smartphones also special ruggedized versions are available.

Many of the points mentioned further below would also apply to mobile devices capable of running a *JavaME* virtual machine, but there are grave arguments which object to an implementation of *DiversityMobile* in a first approach. Firstly, *MIDP 2.0* – the most common *JavaME* standard today, does not support *JDBC* and stores data over the Java Record Management System (*RMS*). As it is a central point in the IBF project to guarantee provenance of data and to store data in central databases such as *DiversityCollection*, it was regarded as convenient to use database technology for the purpose of temporary storage on the mobile device and for the synchronization of the data. Another reason for the choice of Windows Mobile over *JavaME* was that *DiversityMobile* is not a standalone application but part of the *Diversity Workbench* application framework which connects several database components for various domains of biological data. As most of the current existing applications were developed under the *.NET* framework the decision to use the *.NET Compact Edition* framework is also a result of considerations concerning project consistency.

Moreover, *DiversityMobile* can be easily ported to the normal *.NET* framework and therefore also runs in principle on laptops or desktops. This is an important point in respect of the relationship between *DiversityMobile* and *DiversityCollection* as part of the *Diversity Workbench.*

3.2 Architecture of *DiversityMobile*

The high-level architecture of *DiversityMobile* is shown in Figure 1. It consists of four tiers or layers which are (from top to bottom) Presentation, Controller, Data and Device Management and the Database Backend. The architecture is strongly oriented on the Model-View-Controller design pattern as introduced in [Re79] and [KP88].

The Presentation layer contains the graphical user interface (GUI) and is thus responsible for the interaction of *DiversityMobile* with the user in the field. Its layout as well as its content can be defined freely. However, in practice it does make sense to display only such values which the storage model of *DiversityMobile* is capable of saving and loading. Moreover, as the IBF project is connected with *DiversityCollection* it is crucial to support the database models used by *DiversityCollection* and supporting components of the *Diversity Workbench*.

The Controller is the core of the application. It connects the Presentation layer with the Data and Device Management layer. The main task of this layer is to interpret the input of the user. Other tasks range from the activation and de-activation of components and sub-systems depending on the current power management mode and application activity to manage the data transfer in between the other layers. It also communicates with the outside world (i.e. a server system which in fact hosts the institutional repository databases) via a special synchronization interface. Depending on the commands of the controller, the Device Management layer activates, de-activates or retrieves data from connected or built-in devices such as microphones, cameras or GPS chips (requirement (vi)).

The Data Management layer is used as an abstraction to the underlying data stores. Its main purpose is to link the gathered data against entities defined in the database model of *DiversityCollection*. To perform this, a special object-relational mapper (*ORM*) using the *ADO.NET* component of the *.NET* framework was developed. The *ORM* framework used within the project was implemented by the project group itself because at that time none was available that was running on the *.NET Compact* Framework and supporting composite primary keys. This special feature is needed because of the database structure of *DiversityCollection* in which as a well established repository many entries already exist. The *ORM* tool supports the construction of queries in analogy to the criteria framework introduced with *Hibernate* [BK06] and lazy loading of object collections.

As a backend at least two databases which contain information about the taxonomy and are able to store the field data in the database structure of *DiversityCollection* have to be included. Here, several taxonomy databases can be accessed (on- and offline, offline by upload from the *Diversity Workbench* module *DiversityTaxonNames*) and items can be linked against entries of these databases. To perform this, the Data Management Layer integrates a Taxonomy Wrapper. If an identification unit cannot be assigned to a specific taxon, it is also possible to enter "working names" which are placeholders in *DiversityCollection* to be filled in later with proper names from a taxonomic thesaurus for each data item. The working name may be substituted at any time in the data flow from the field to the central repository which defines a special requirement in conflict handling during the synchronization process (Chapter 3.4). Instead of assigning working names, scientists can also link voice recordings of a taxonomic term with the help of the Device Management component. This corresponds to requirement (v).

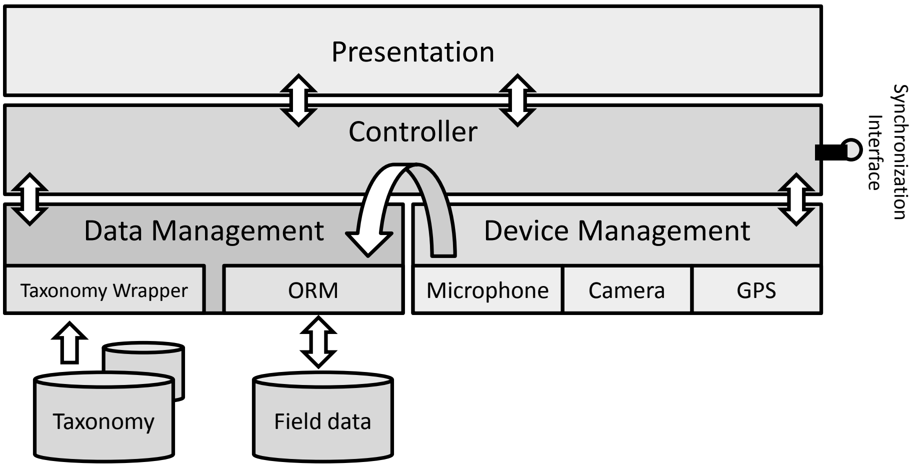

Fig. 1. Four-tier architecture of *DiversityMobile*

The advantage of the architecture of *DiversityMobile* is a decoupling of the components concerned with data management from the other components, especially the visual representation. This helps to adapt the development to the requirements of the different scientific interests of the research groups concerned. Whereas the data model (cf. Chapter 3.3) is rather stable, the various scopes and goals of biological projects often require the adaption of the user interface. In principle, also the data model can be easily modified or even exchanged in case this should be necessary.

3.3 Data Model

DiversityMobile per se does not require its users to follow a specific data model. However, in the context of the research project IBF we adapted the information model of *DiversityCollection* [WHT07]. Figure 2 shows the important (and simplified) parts of this data model in the style of an UML class diagram (please note that these classes are mapped into a mobile database by the *ORM* framework applied).

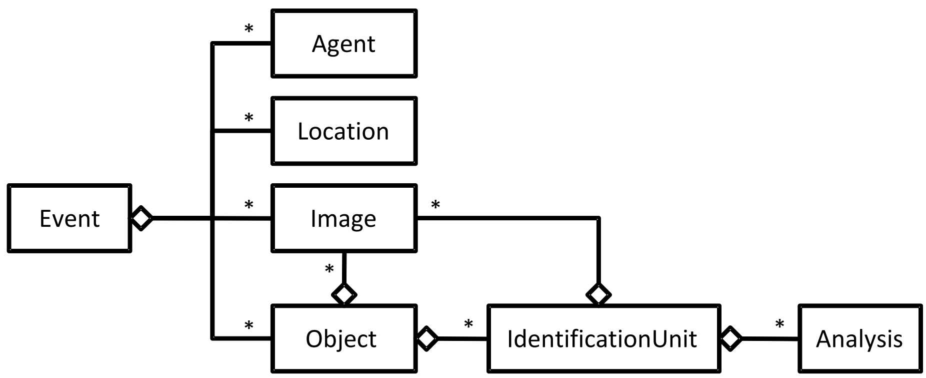

Fig. 2. Simplified version of the *DiversityCollection* data model as applied in the current *DiversityMobile* prototype

The *DiversityMobile* data model thus contains the entities Event, Agent, Location, Image, Object (Observation/Specimen), IdentificationUnit and Analysis. Event is used to collect one or more objects (observations/specimens) under a common denominator. Usually, an event carries a time period in which all observations were made and specimens were collected. It is further defined by location information (pure textual input or coordinates grabbed from a GPS device) and – if needed – by images which capture for instance the surrounding of an object (= observation/specimen). Figure 3 shows a screenshot of the current implementation of *DiversityMobile*. In the tree view of this screenshot an event is depicted with attached location information, here the altitude in meters. The image in the background showing the sampling plot can be attached to this specific event. Furthermore a description of the event may be added to the record e.g. for specifying the collection method or giving information about the habitat.

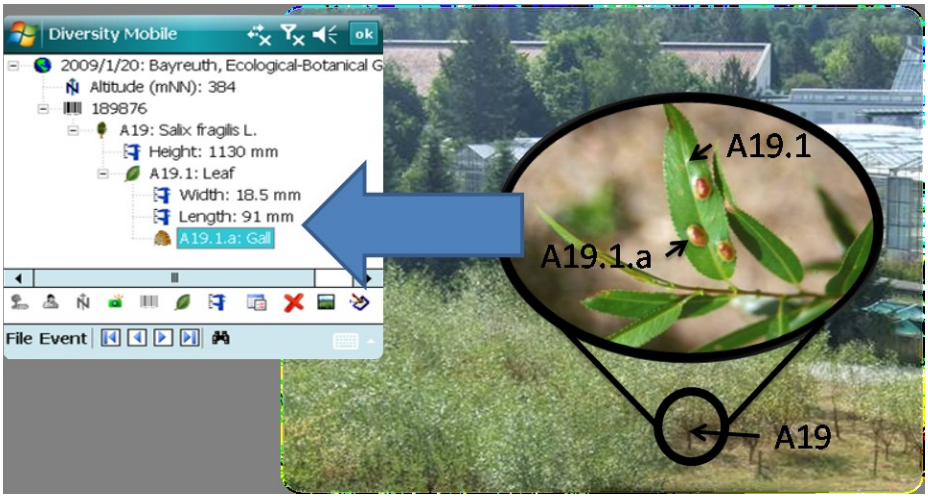

Fig. 3. Exemplary data retrieval of the scenario discussed in Chapter 4

One object might include one or more *IdentificationUnits* linked to each other. The object (observation/ specimen) contains information about a specific sample and can be interpreted in a wide range depending on the specific setting of the project. The main reasons for including this entity are benefits in the administration of the datasets in the central databases. To serve as a primary key an object number is assigned to each object which uniquely identifies each object (observation) and might finally be added by an accession number if a specimen or sample is collected and brought back to a natural history collection or laboratory. In Figure 3 the graphical representation of the object is depicted with an icon showing a barcode.

The object then carries information about "units" that can be identified with a name from a taxonomy. Examples for *IdentificationUnits* are by convention organismic groups such as plants, fungi or insects. To exclude typos and support the gathering person in the decision process, support is provided via the connection to taxonomic databases (Chapter 3.2). For certain projects also parts of the organisms (e.g. trees, bushes, branches, sprouts) can be described as a separate but dependent *IdentificationUnit*. As there is a recursive relationship between *IdentificationUnits* in the data model (cf. Figure 2) a whole plant can be described into detail with the help of a hierarchy of *IdentificationUnits* corresponding to different parts of the plant. Further *IdentificationUnit* types can be easily added by inserting corresponding values into the database. Images can be linked to *IdentificationUnits* for instance in order to visualize the current life stage of a gall or simply grasp an idea about a certain unit.

Different kinds of analyses can then be added to each *IdentificationUnit* depending on the type of the *IdentificationUnit*. Hence, a list of allowed analyses is stored as an attribute of the *IdentificationUnit*. For instance, it does indeed make sense to measure the height of a tree (*Salix fragilis* – for the use case, cf. Chapter 4) in some context – whereas it should not be possible to specify the number of leaves of a mushroom. Other kinds of analyses besides simple measurements can be performed and the result of the evaluation of these analyses can be attached to the *IdentificationUnit* of concern.

3.4 Data Synchronization

For data synchronization a specialized version of the *DaltOn* framework for data transformation [Ja08] is used. *DaltOn* uses an ontological description of a transformation source (device) and sink (server database). In addition, *DaltOn* can generate fine granular information for data provenance. Thus, requirements (iii) and (iv) are fulfilled.

The main difficulty during the implementation of the synchronization process was to manage different primary keys on the server and client side. *DiversityMobile* clients assign primary keys for items independently of any other client or the server. By doing so, every client can use the full range of the primary key type. As the primary key is a surrogate key which is administered only by the database management software, the primary keys cannot be changed during the process of synchronization. Hence, another system for the identification of data sets stored in different databases is needed. This issue is solved with the introduction of one synchronization database for each mobile client which is placed at the location of the server due to security reasons. The synchronization database identifies the data items on the client side with data items on the server side with the help of global unique identifiers (*GUID*) in the Microsoft implementation of the Universally Unique Identifier (*UUID*) standard. To perform this, in every table of the client and server database a column "ROWGUID" is included.

The synchronization database consists out of two tables: "SyncItem" and "FieldState". In "SyncItem" a data item in the client database is connected with a data item in the server database over the *GUID* by mapping the corresponding primary keys of both data items with the corresponding *GUID* crosswise. This is solved by inserting a pair of data items in the synchronization database containing the *GUID* as primary key and the primary keys of the data item of the client and the server database as foreign keys. During the process of synchronization it is checked, whether the *GUID* of a data item is already present in the synchronization database.

If not, the data item can be identified as "new" and corresponding entries in the synchronization database and the server database are created. Thus, this synchronization metadata contain information about which elements are stored on what client under what primary key. In case a new element is to be copied into the server database, a new value for the primary key on the server-side is requested and put into place. After that, all dependent objects are updated automatically by interpreting the mapping information of the *ORM* framework.

Otherwise, if there is already an entry in the synchronization database it is identified that there must be a corresponding entry in the server database and it needs to be checked if the data item on the server side stays in conflict with the data item on the client side.

To perform this, a hash value of all fields of a data item is stored after every successful synchronization. If a data item is already present in the database this hash value has to be recalculated. If the hash value has changed it is implied that the data item in the client and the server database contain different information. As a pair of entries in the synchronization database is connected with every data item, two hash values exist for each data item – one for the fields on the client side and one for the fields on the server side. If the hash value of the server side has not changed an update process is indicated. Otherwise a conflict occurred.

This happens, for instance, when the same data item is modified simultaneously on two or more devices including the server. To identify which fields in the data item contain the conflicting data the table "FieldState" is used. In this table for every field of a data item an entry is made and assigned with a hash value. Analogously to the identification on item level the hash values of a field are compared. If the hash values of two corresponding entries do not match a conflict is detected and localized. To resolve the conflict, the corresponding data item can be displayed on the synchronization interface (for the place of the synchronization interface in the architecture cf. Figure 4). As the current implementation runs on a local computer (laptop or desktop), we are using the normal screen of this computer for displaying and resolving conflicts rather than the screen of the mobile device which is limited with respect to screen size and resolution. However, it is also planned to implement the synchronization as a web service that can be accessed via a network connection (GPRS, UMTS, WLAN) from the mobile devices. This synchronization client is depicted between *DiversityMobile* and *DiversityCollection* in Figure 4. Thus, the synchronization and the *ORM* framework complement each other and can be seamlessly integrated.

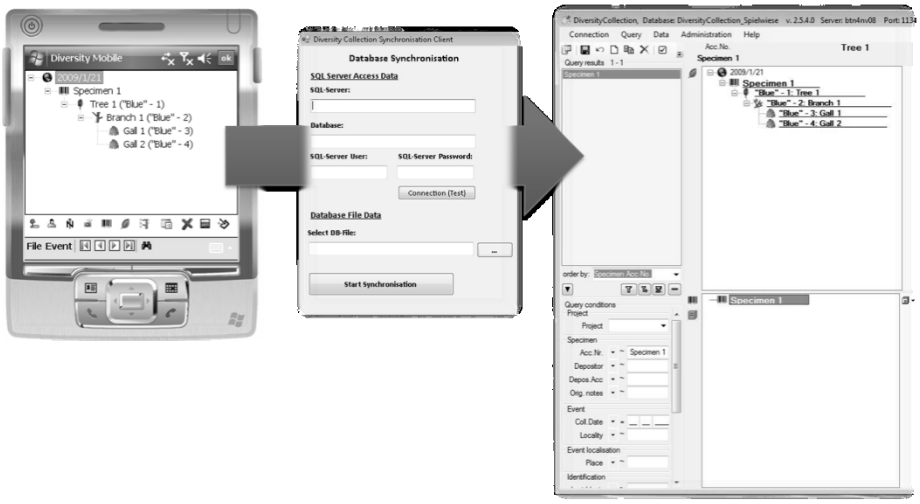

Fig. 4. Outline of the synchronization between *DiversityMobile* and *DiversityCollection*

3.5 Current Status of the Implementation

The first version of the *DiversityMobile* data retrieval application comprises the ability to add, modify or remove events, agents, locations, objects, *IdentificationUnit*s, analyses and images. Images can be acquired using the built-in camera (if present in the device of choice). However, the application also provides support for selecting images of an extension card such that external imaging devices can be used as well. Information about the location can be enriched with data from a GPS device (either internal or externally connected). Searching taxonomies has been implemented as a proof of concept but is not yet integrated into the current version. Also online access to a taxonomy database for instance via a GPRS or UMTS network is planned but has not yet been realized.

Data synchronization is complete with respect to the function of data transfer; conflicts are recognized but can only be resolved manually by overwriting the entire data item on the server- or client-side for the time being. A client for customized data synchronization that also supports a more complex conflict resolution (such as merging components of data items from both sides) will be implemented within the next months.

4 Case Study: Data Recording of Plant-Insect Interactions

The survey of specialized herbivores on their host plants and their temporal and spatial distribution is a suitable case to test and improve the capabilities (handling, expenditure of time, data structure, accessibility of data) of DiversityMobile. In this special case, plant galls (local restricted plant deformations on leaves or shoots which are induced by insects during oviposition sheltering one or more larvae) of three gall-inducing sawfly species (Hymenoptera, Tenthredinidae) are recorded on 40 plants on a cutting plantation of willow clones (concerning the study design see [KAR08]). The female insects eclose in early spring and search for oviposition sites. During the oviposition process, the gall growth is induced by a special fluid injected by the female in very young leaf tissue. These gall-inducing insects are very host specific and the galls can easily be identified due to their characteristic morphology. Plant galls are adequate for field studies, since they can be quantified easily. They indicate the successful gall induction (and oviposition) event, but even events as death or emergence of the inhabiting larvae can be recorded also after a longer time span.

The three gall-inducing sawfly species which will be investigated here (*Euura testaceipes* Brischke, *Phyllocolpa oblita* Serville, 1823 and *Pontania proxima* Serville) differ slightly in the time point of oviposition start in spring, and accordingly are expected to be found on different leaves (differing in time point of sprouting in spring). The aim of the study is to describe and quantify the temporal and consequential spatial separation of these herbivores on the plant during two consecutive vegetation periods. Therefore, multidimensional complex relationships need to be recorded (plant, shoot (several orders possible), leaf, gall) to display the spatial distribution of the galls on the individual plants and plant parts in detail along a time span of several months. During data acquisition, it is necessary to link each recorded item to the respective item on the higher level (i.e. leaf is located on a shoot, the gall is located on the leaf, etc.). Several measurements will be applied to most of the recorded items (i.e. height of the plant, length of the shoot, size of the leaf, position of the gall on the leaf etc.) also partly in a regular manner over months.

This survey will be used to test and improve the data recording via mobile device and the data flow from the field to the data repository. One major aim is to compare the expenditure of time for data acquisition, storage and management using *DiversityMobile* and *DiversityCollection* and the "classic" method using paper and pen, followed by data input in Microsoft® Excel (or comparable software).

It is expected that the process of data gathering in the field can be improved by using *DiversityMobile*, even if a certain time of practice should be allowed. Especially the direct or indirect failures which occur inevitably during the “normal” process of data handling (writing on paper, manual transfer to the PC, copying data to several applications, …) are expected to be reduced significantly which allows to fulfill requirement (ii).

5 Conclusion and Future Work

The architecture of *DiversityMobile* is described as a prototype mobile system used for entering, modifying or even deleting biological monitoring and ecological data already in the field. Its main architectural abstractions and the applied data model are shown as well as the mechanisms how the data transfer from mobile devices to a central database system is performed. Further, one of the biological research projects is introduced in which we are going to use, evaluate and extend the prototype. The use of *DiversityMobile* will make the flow of data from the field to the central database system more efficient (retroactive data entering becoming obsolete) and more transparent by recording data provenance information. Altogether, the requirements for a successful implementation as listed in the introduction will be fulfilled by the current development.

A big challenge is still the integration of taxonomy databases since they often consist of thousands of entries which are to be searched. Thus, special search strategies have to be designed in order to shorten the delay time whenever a taxonomy database is to be queried.

Another topic is the adaption of the Data Management layer and the graphical user interface with respect to the specific requirements for these layers from different biological projects. For that purpose, a client application will be set up that allows data producers to parameterize *DiversityMobile* according to their needs, load the parameterized version on the device and record data with it.

6 Acknowledgements

The project "Setting up an **I**nformation Network on **B**iological Research Data gained in the **F**ield" (IBF) is funded by the German Research Foundation (Deutsche Forschungsgemeinschaft, DFG) under the grant numbers INST 106535/1-1, INST 2850/1-1, INST 21946/1-1 and INST 747/1-1.

References

[Ad09] Addink, W.: ETI BioPortals and Mobile Identification software. – In Anonymous (ed.), e-Biosphere 09 Conference, 1-3 June 2009, London UK, Abstracts. – p. 79., 2009

[BK06] Bauer, C., King G.: Java Persistence with Hibernate. Manning Publications, 2006.

[BDA07] Bresinsky, A., Düring, C., Ahlmer, W.: Datenbank PILZOEK im Internet. 2. Update, 2007 – http://www.pilzoek.de. Retrieved 02.2009, from http://www.pilzoek.de.

[HS01] Hennekens, S. M., Schaminée J.H.J.: TURBOVEG, a comprehensive data base management system for vegetation data. Journal of Vegetation Science 12: p. 589-591, 2001

[Ja08] Jablonski, S., et al.: Architecture of the DaltOn Data Integration System for Scientific Applications. International Workshop on Applications of Workflows in Computational Science (AWCS) in conjunction with International Conference on Computational Science (ICCS). Krakow, Poland, 2008

[KAR08] Kehl, A., G. Aas, Rambold G.: Genotypical and multiple phenotypical traits discriminate *Salix × rubens* Schrank clearly from its parent species, Plant Systematics and Evolution 275: 10, 2008

[KP88] Krasner, G., Pope S.: A cookbook for using the Model–View–Controller interface paradigm in Smalltalk-80. Journal of Object Oriented Programming 1(3), p. 26-49, 1988

[KMH09] Kroupa, A., Monje, J.C., Häuser, C.L.: Field recording tools and techniques for all taxa biodiversity inventories + monitoring (ATBI + M) sites under the European Distributed Institute of Taxonomy (EDIT). – In Anonymous (ed.), e-Biosphere 09 Conference, 1–3 June 2009, London UK, Abstracts. - p. 140, 2009

[Ma08] Manoharan, A., et al.: Optimizations for the EcoPod field identification tool. BMC Bioinformatics 9(1): 150, 2008

[Re79] Reenskaug, T.: Thing–Model–View–Editor, an example from a planning system**:** Xerox Parc Technical Note, 1979

[SF06] Sievers, R., Frahm J.P.: FLORKART/M – System zur floristischen Kartierung von Moosen. 2006, Retrieved 02.2009, from http://www.bryologie.uni-bonn.de/FLORKARTM.pdf.

[Tr09] Triebel, D., et al.: Developing a sustainable working platform for gathering biological data in the field. – In Anonymous (ed.), e-Biosphere 09 Conference, 1–3 June 2009, London UK, Abstracts. – p. 142, 2009

[WHT07] Weiss, M., Hagedorn G., Triebel D.: DiversityCollection information model (version 2.05). 2007 Retrieved 2009-02-05, from http://www.diversityworkbench.net/Portal/CollectionModel_v2.05.

[Yu06] Yu, Y., et al.: EcoPod: A Mobile Tool for Community Based Biodiversity Collection Building, Proceedings of the 6th ACM/IEEE-CS joint conference on Digital libraries; Chapel Hill, NC, USA ACM Press; p. 244-253, 2006