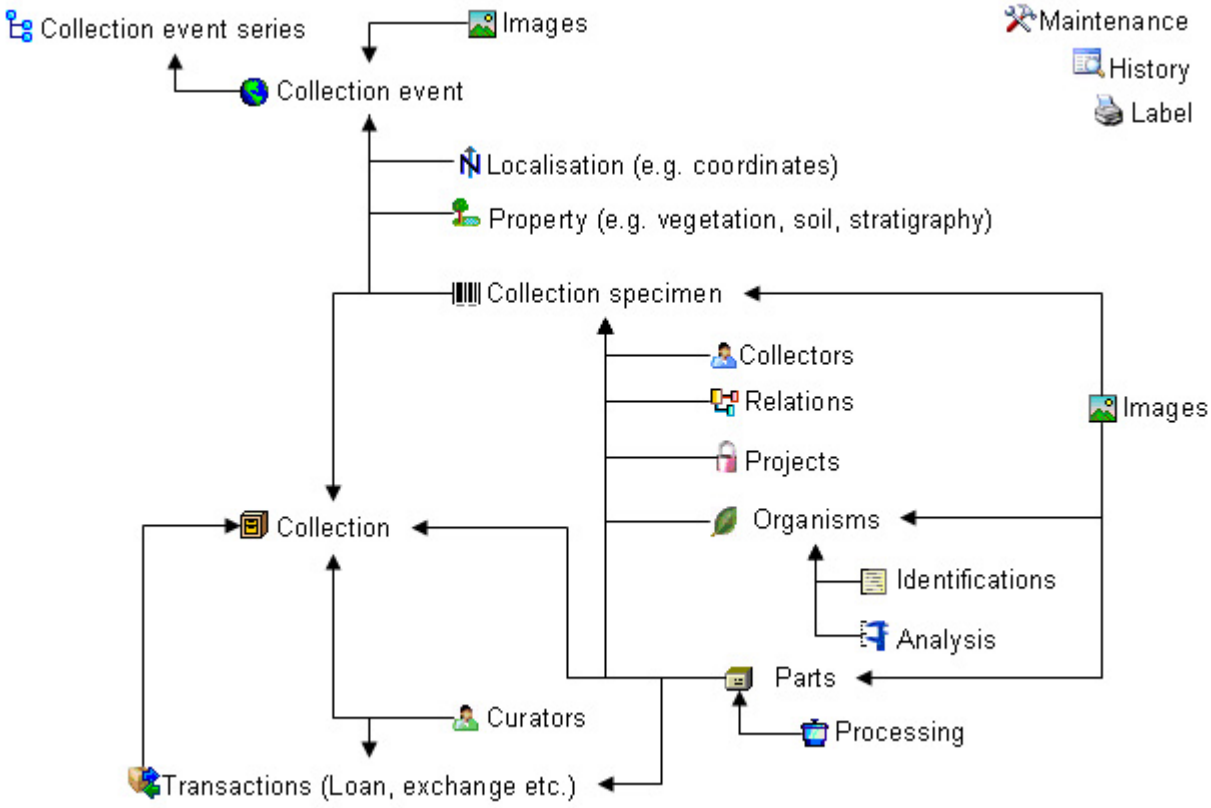
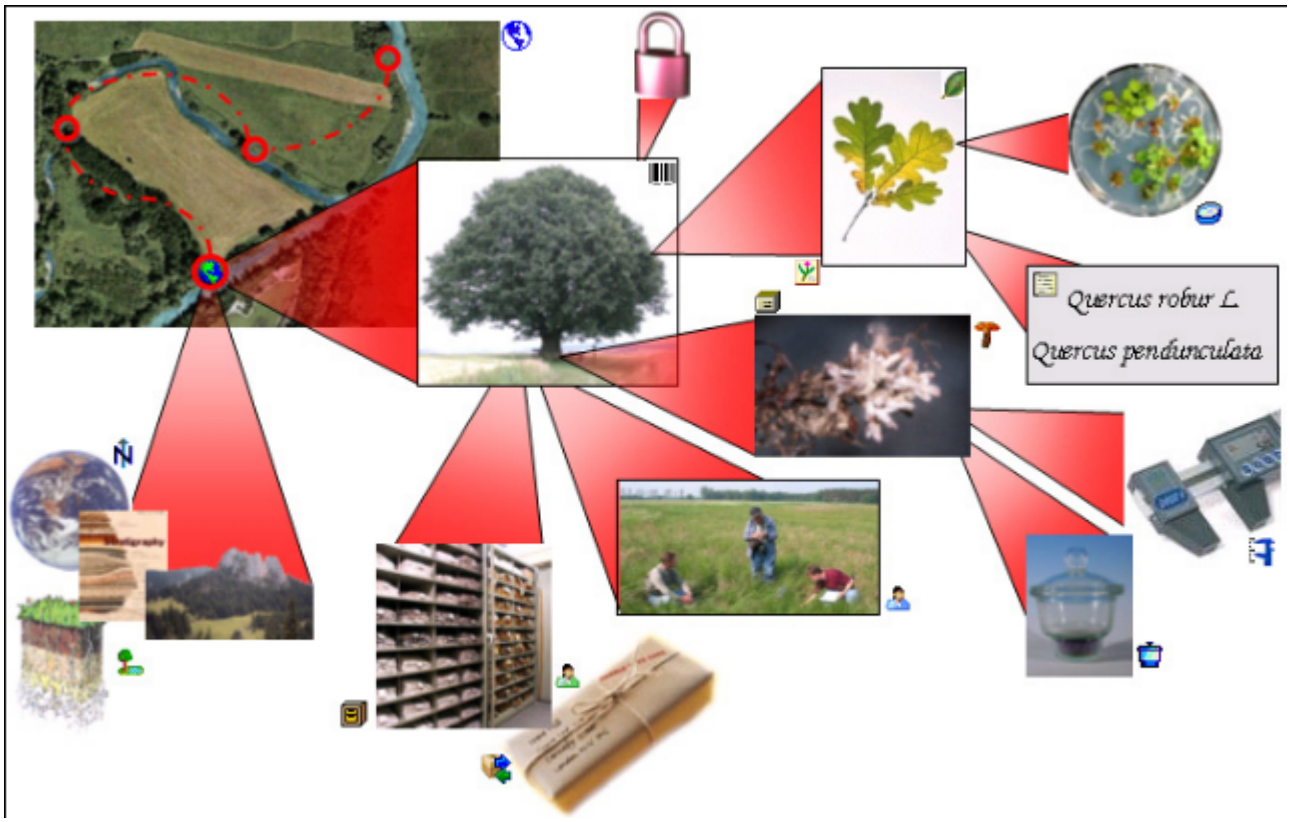
DiversityCollection

DiversityCollection (Version 2) is part of the database framework Diversity Workbench. Within this framework the application DiversityCollection is confined to the management of specimens in scientific collections. In this context it is designed to documente any action concerning the collection, storage, exchange and treatment of specimens in a collection and is also appropriate to store observation data. DiversityCollection is distinguished from other collection management systems by its focus on biological relations between organisms collected together as one or more specimens (e.g. host, parasite and hyperparasite, symbionts etc.). Any module within the Diversity Workbench is focused on a specific data domain. DiversityCollection keeps only data connected with the handling of collection specimens and observations. Data of other realms like e.g. taxonomy are handled in separate modules. For an overview of the available modules see Diversity Workbench. DiversityCollection might also be used as a stand-alone application.

The image below gives you an overview of the main parts, relations and functions of DiversityCollection



The image below gives an overview for some typical data depicted in DiversityCollection together with the symbols used throughout the program. A typical specimen may have been collected at a collection event during an expedition. The site of the collection event may be localized and characterized. The colletors collected twigs of the plant and fungi from the roots. They store the samples as herbarium sheets and specimen in a collection. The curator sent some of the samples to another collection. The samples were collected as part of a project. Certain parts were cultivated, analysed and processed. The organisms on a herbarium sheet or specimen were identified several times. Images were taken for the event series, the collection event, the Collection specimen as well as organisms and part of this specimen.



DiversityCollection 2 is based on Microsoft SQL-Server 2005 and the .Net Framework, Version 2.0.

Copyright: Diversity Workbench 1999 - 2008.

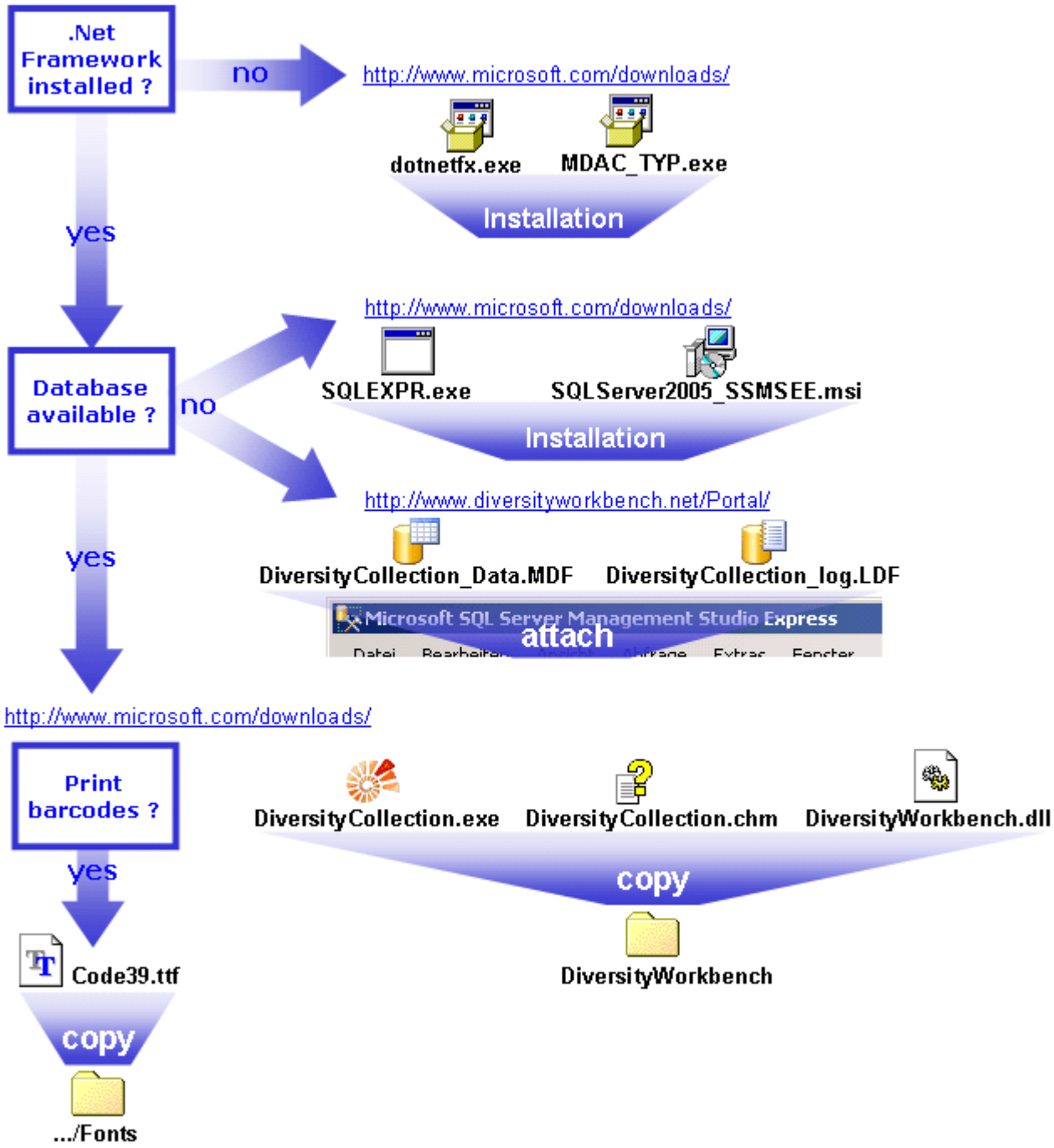
Author: Markus Weiss.

Licence: GNU General Public License

Installation

To run DiversityCollection, you need the database and the client. All parts are free and can be downloaded from http://www.microsoft.com/downloads/ and http://www.diversityworkbench.net/Portal/.

The image below gives an overview of the installations and files needed.

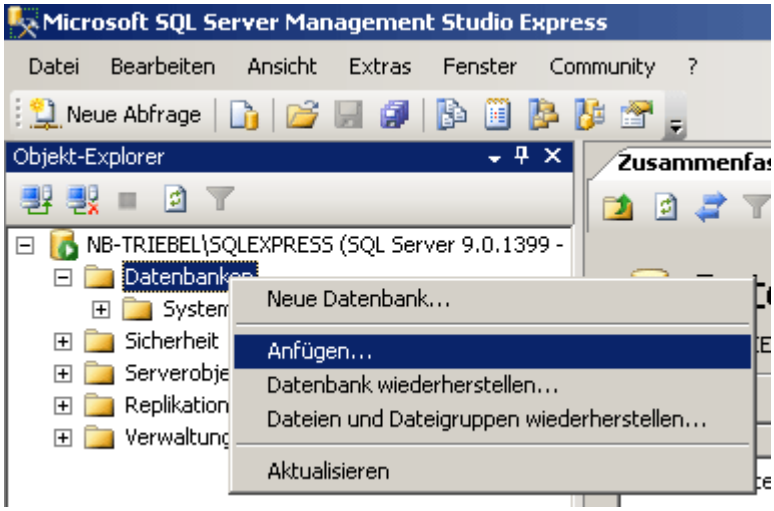


Database

DiversityCollection uses Microsoft SQL-Server 2005 as database engine. If you do not have a database server with DiversityCollection already available, you have to install the database engine first. Download the free version of Microsoft SQL Server Express

(SQLEXPR.exe resp. the version of you preferred language, e.g. SQLEXPR_DE.exe) and Microsoft SQL Server Managment Studio Express (SQLServer2005_SSMSEE.msi)from http://www.microsoft.com/downloads/. Start the program and follow the installation instructions.

Download the database files DiversityCollection_Data.MDF and DiversityCollection_log.LDF from http://www.diversityworkbench.net/Portal/ and copy them into your database directory. You have to attach the database to the database engine using the Microsoft SQL Server Managment Studio Express. In the **Object-Explorer** select databases and choose attach from the context menu as shown below.



A form will open where you can choose the file DiversityCollection_Data.MDF and attach it to the database engine.

Client

The client is based on the .Net framework from Microsoft. If not already present, you have to install the following components first:

Microsoft .NET Framework Version 2.0. Download (dotnetfx.exe), start the program and follow the installation instructions.

Microsoft Data Access Components (MDAC) 2.8. Download (MDAC_TYP.exe), start the program and follow the installation instructions.

(see http://www.microsoft.com/downloads/ for the latest versions)

Download the files for DiversityCollection from http://www.diversityworkbench.net/Portal/

Copy all files (DiversityCollection.exe , DiversityCollection.chm , DiversityWorkbench.dll) into your DiversityWorkbench directory. To print barcodes the font for Code 39(Code39.ttf) must be copied to the folder C:\WINNT\Fonts.

After the installations make shure to get the latest updates from http://v4.windowsupdate.microsoft.com/.

Menu

Overview of the menu in DiversityCollection

Connection

Database	Choose one of the databases available on the server. Only those databases will be listed to which the user has access permission.
Transfer previous settings	Transfer the settings of a previous version.
Quit	Quit the application and stop all processes started by the application.

Query

Show query	Show or hide the query list
Predefined queries	Under this menu entry all predefined queries will be listed
Scan mode	Change to the scan mode to open the dataset of a specimen by scanning the barcode

Data

Import	
Specimen scans ...	Import scans of specimen labels
Import list ...	Import tab separated lists
Export	
Export list ...	Export a tabulator separated file with the data of the specimen
XML ...	Export date as a XML-file according to ABCD

Administration

Analysis ...	Administration of the analysis methods used for the collection specimen
Collections ...	Administration used for the collections.
Maintenance ...	Maintenance of database entries especially if connected to other modules.
Material categories ...	Administration of the material categories that should be visible when creating a new part for a specimen.
Processing ...	Administration of the processing procedures applied in the collection.
Queries ...	Creating and editing predefined queries.
Taxonomic groups ...	Administration of the taxonomic groups that should be visible for creating a new identification unit.
Transactions ...	Administration of the transactions, e.g. loans, exchange etc.
Expired loans ...	Administration of expired loans. This menu entry will appear when there are expired loans in collections where the current user is a curator.
User ...	Administration of the user and their permissions in the database.
Curators ...	Administration of the curators, responsible for the administration of the transactions.

Help

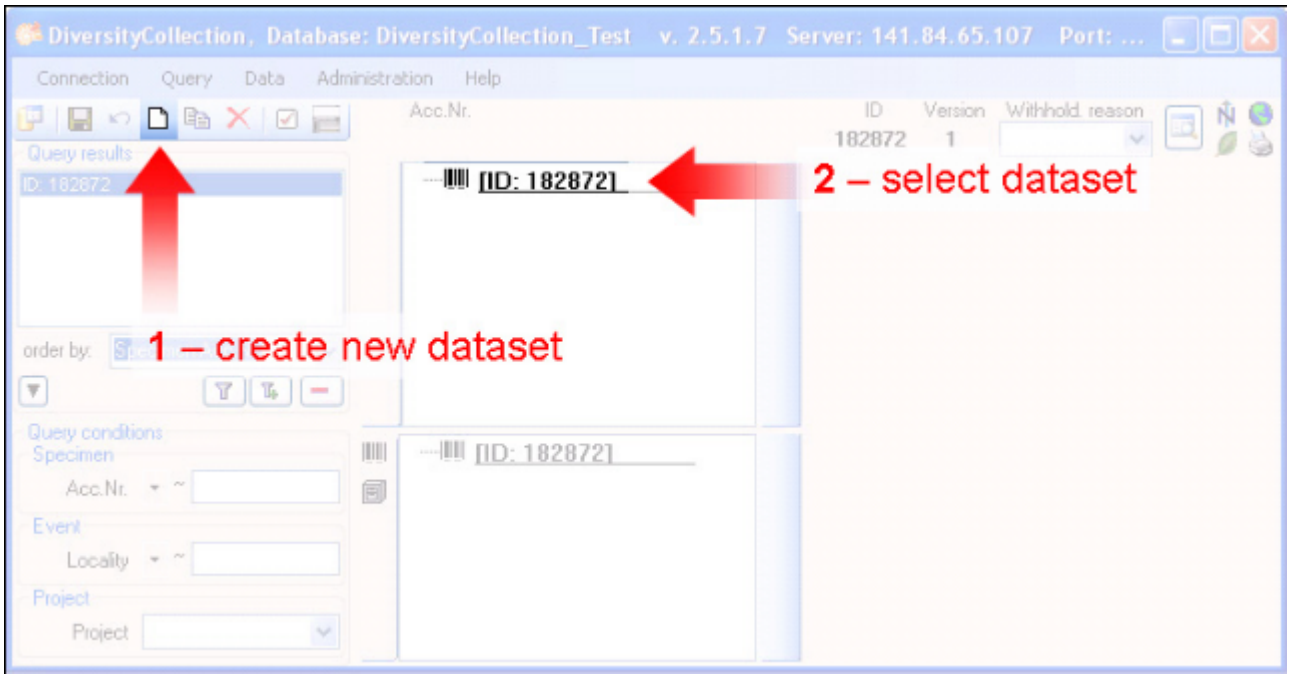
Manual	Opens the online manual
Info	Show the version and corresponding information
Feedback ...	Opens a form for sending feedback

Manual

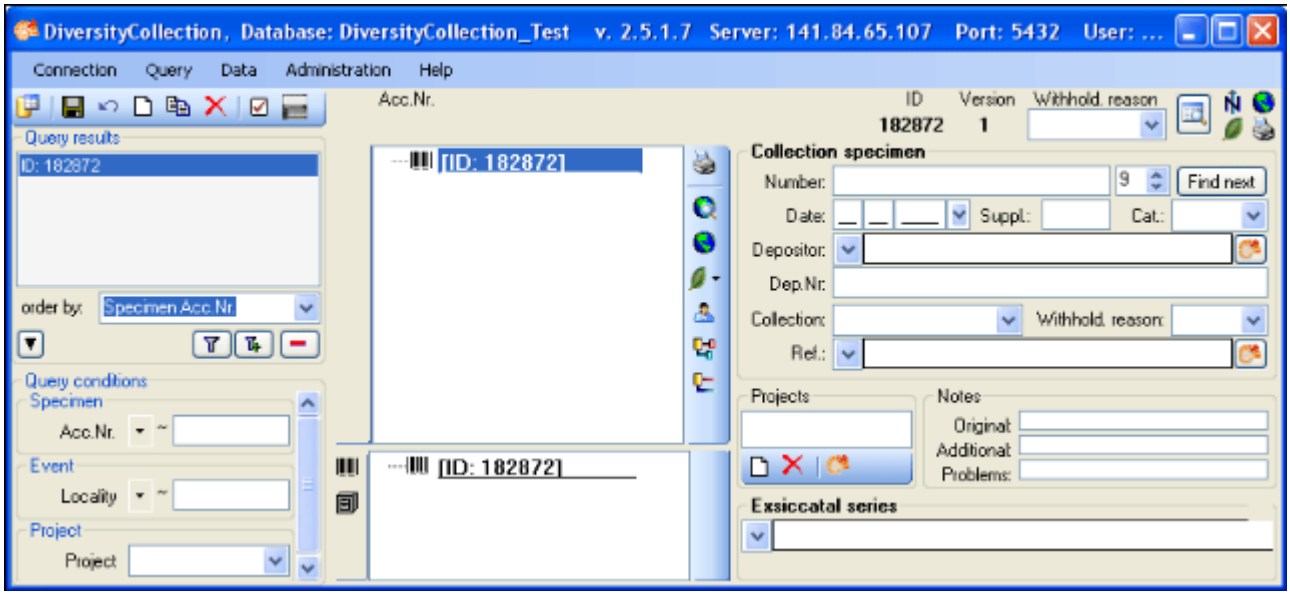
The online manual DiversityCollection.chm must be placed in your application folder, together with the application DiversityCollection.exe and the library DiversityWorkbench.dll.To get information to any topic in the application DiversityCollection and open this manual, just click on the field you need information about and press F1.

Tutorial - first steps

This tutorial will guide you through the first basic steps to enter a single dataset in DiversityCollection. After the installation, make sure, you have access to the database. If you have the rights to enter any data, click on the button in the upper left panel (see point **1** in image below). If there are accession numbers in the database, a dialog will appear if you want to take the next free number. In the end you will find a new entry in the specimenlist on the left and the trees for the specimen (see point **2** in image below) and the storage as shown below.



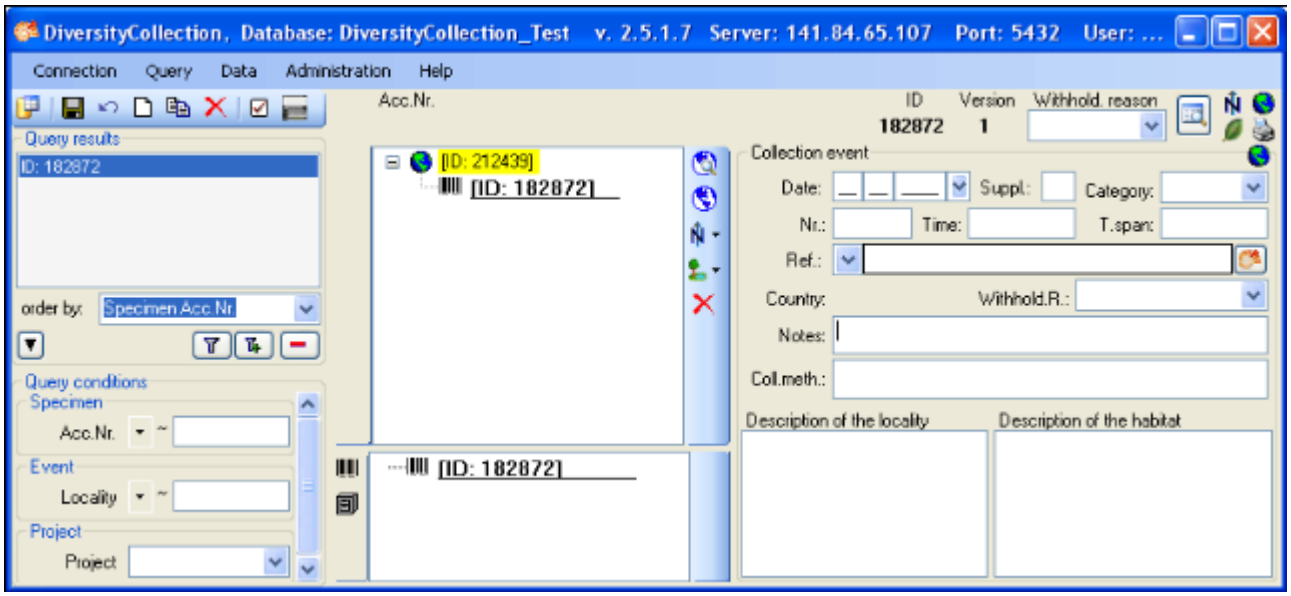
If you did not specify an accession number the entries will look like in the image above, otherwise the accession number will be shown. To enter your data for the specimen, click on the entry in the upper tree as shown below. This will open the fields where you can enter the details for the specimen in the area right from the tree.



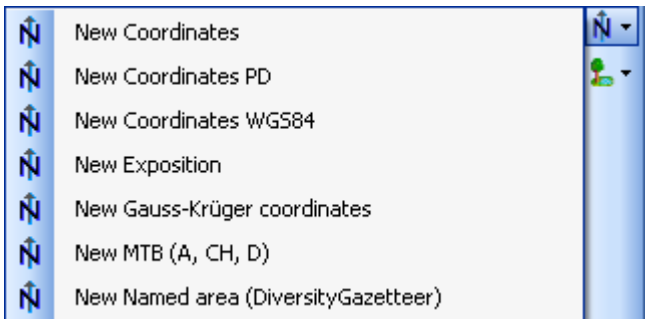
In the panel right of the tree, several buttons will appear where you can enter additional informations to this specimen. To enter information about the collection event (when and where the specimen was collected) click on the button. This will add an entry for the collection event in the upper tree as shown below. Select this entry in the tree, to open the fields for the collection event. To see the projected contents of the fields, simply place you mouse in the field. A explanation will appear like for the field **Description of the locality**:

Locality description of the locality, exactly as written on the original label (i.e. without corrections during data entry)

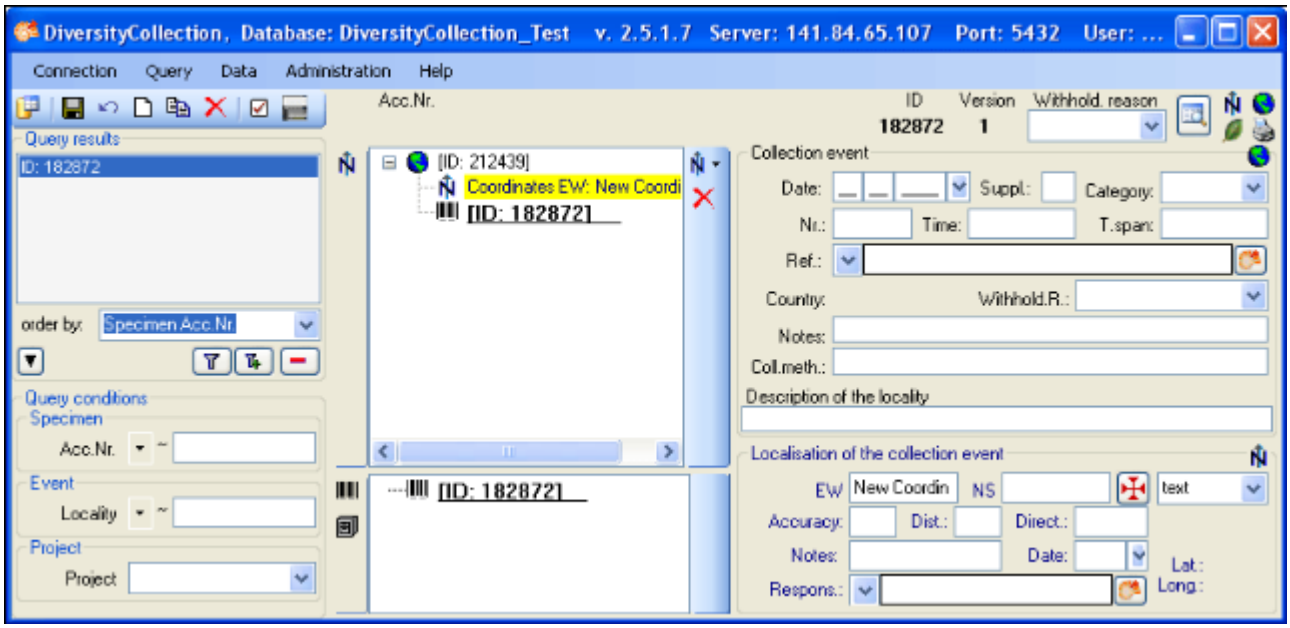
See the event part for further details.



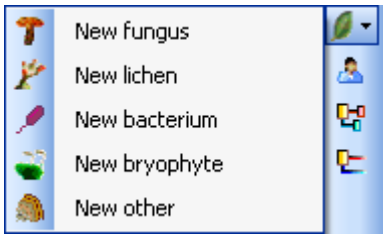
To enter coordinates, named places etc. select an option from the list.



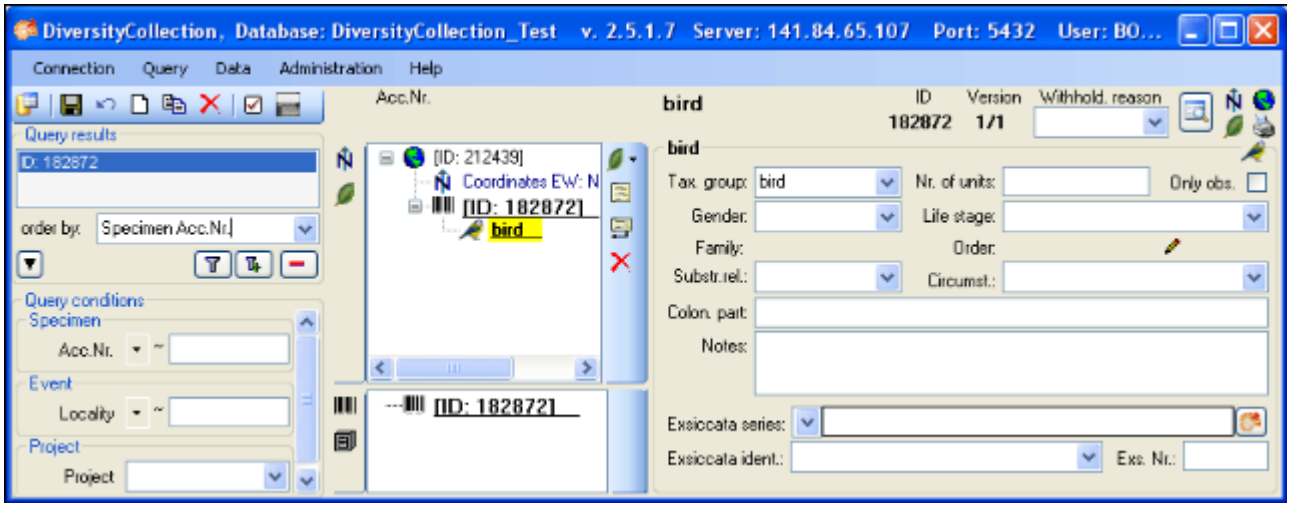
In the example below, the coordinates were selected - the fields for the data entry are located in the lower part at the right. You can add any additional details from the list above. See the geography part for further details.



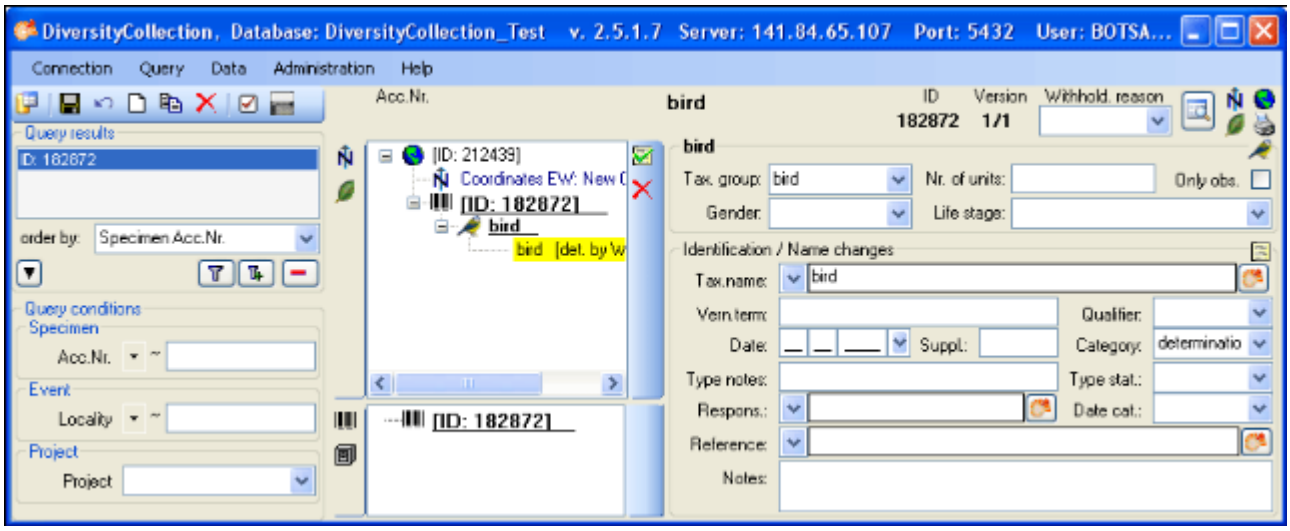
To enter the organism, choose the specimen entry in the tree. Then select the organism from the list.



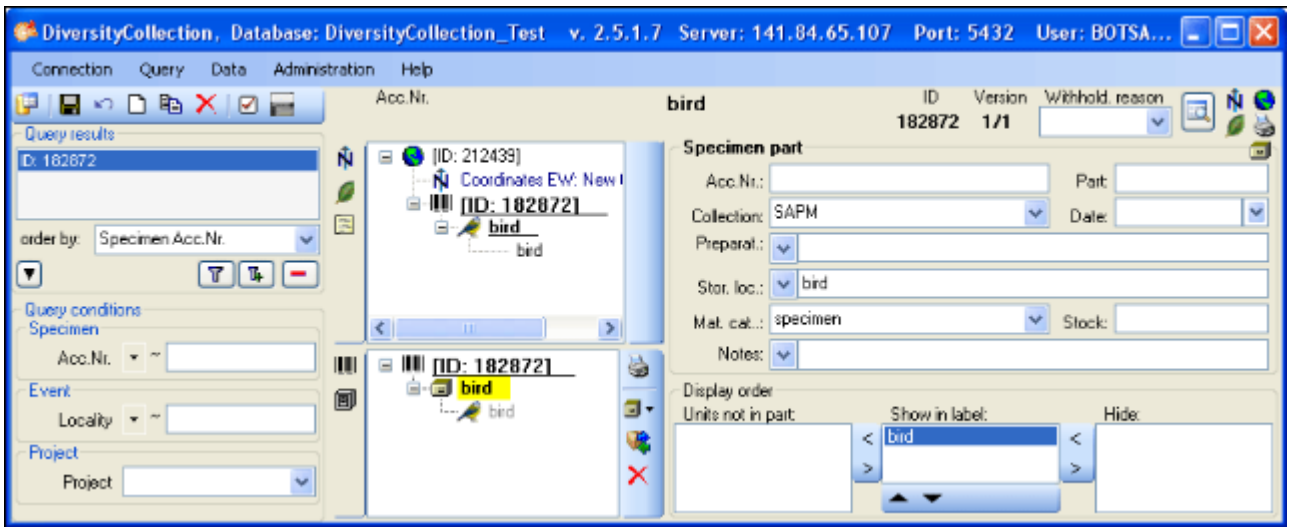
This will enter an entry for the organism unterneath the entry of the specimen as shown below. To enter details for this organism, select the entry in the tree.



To enter an identification, click on the buttom. This will enter a dataset for the idenfication of the organism underneath the organism as shown below. Select this entry to enter the details of the identification.

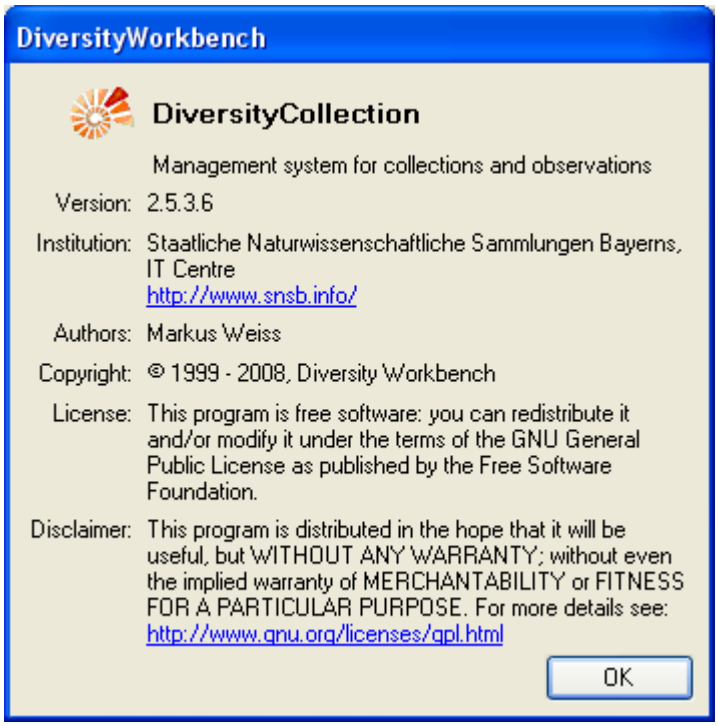


As a last step, enter the data connected with the [storage] of the specimen. In the bottom tree, select the entry for the specimen and than select a material category from the list. This will enter a dataset for a specimen part underneath the specimen as shown below. Select this entry to edit the data for the collection, the storage location etc.



Version

For information about the version of the client application choose **Help**, **Info...**



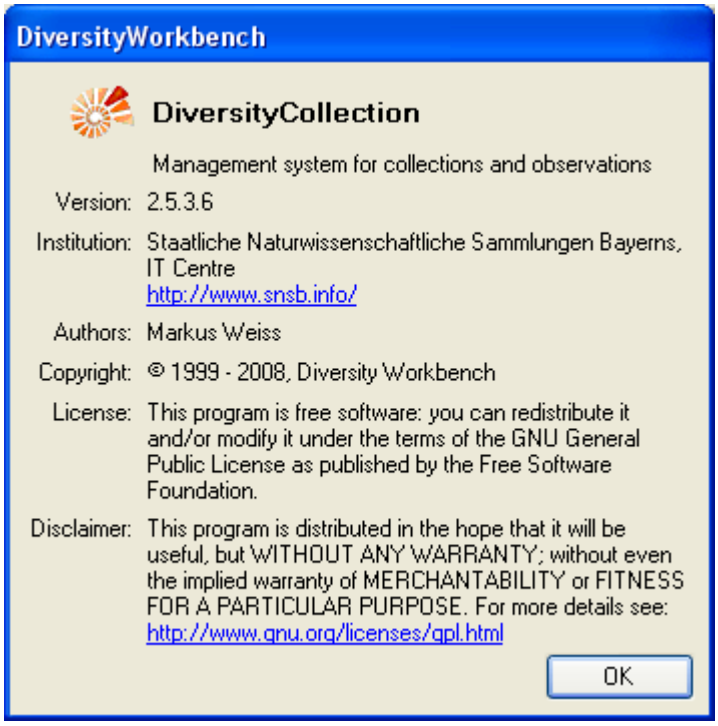
The current version in the example above is 2.5.3.6

License

This manual is copyrighted work licensed under a Creative Commons License.

All material in this manual is the property of the contributing authors and fully copyrighted. By choosing this way of publication, the contributing authors have agreed to license the work under a Creative Commons License permitting reproduction, distribution, and derivative works, requiring attribution, notice, and share-alike, and prohibiting commercial use.

For information about the license of the client software choose **Help**, **Info...**

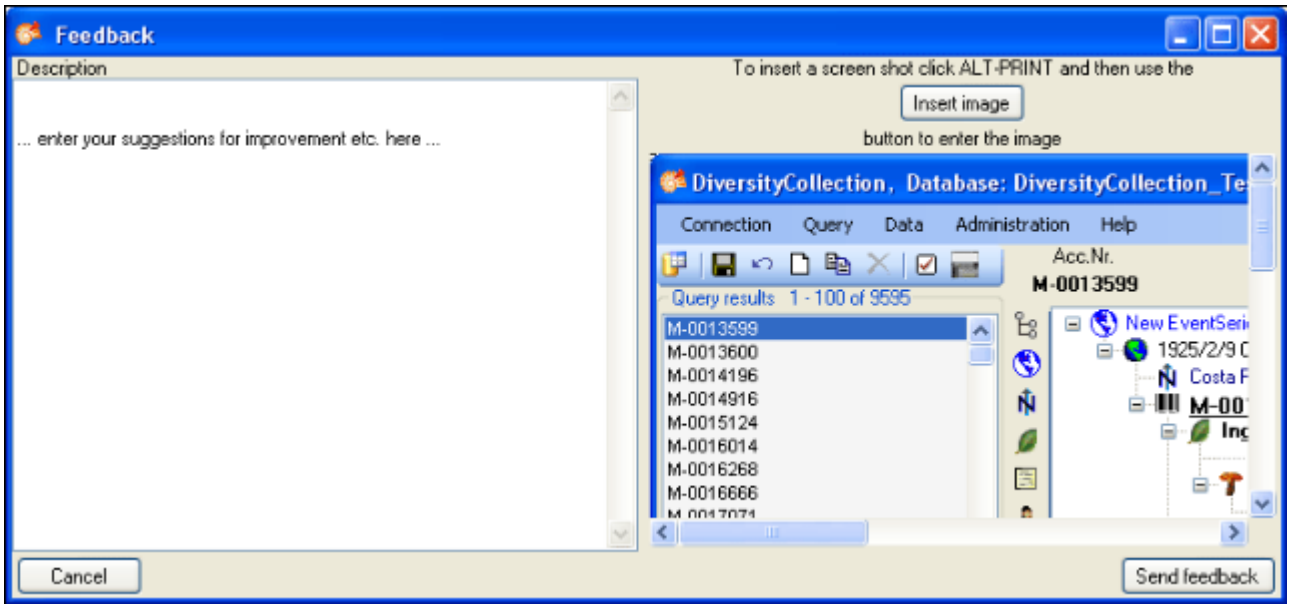


The client software is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation.

The client software is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License (GPL) for more details.

Feedback

If you have suggestions for improvement, need any changes in the programm or encouter an error you can give a feedback to the administrator. Click the [**ALT**] and [**PRINT**] buttons to get a screen shot of your current form. After creating the screen shot choose **Help - Feedback** from the menu to open the feedback form as shown below.



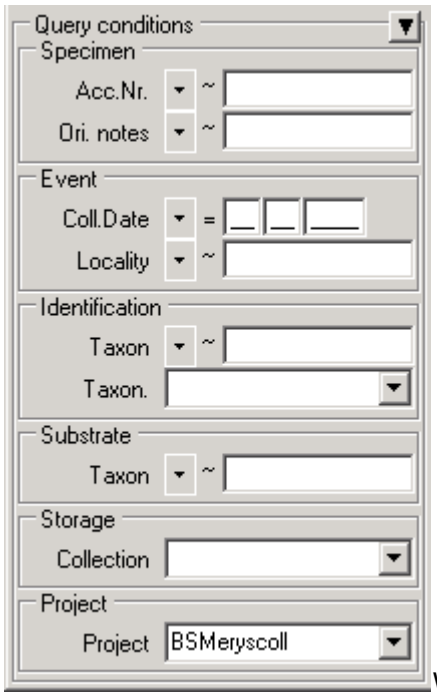
Click on the Insert image button to insert the screen shot and give a comment about your problem. Then click on the Send feedback button to send your feedback to the administrator.

Error logging

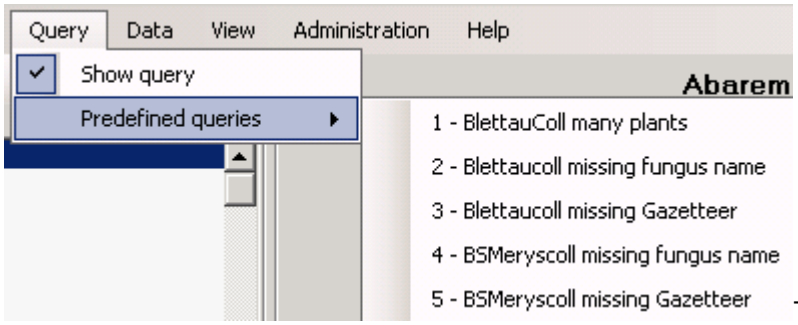
If any error messages show up through working with the application you can find further details concerning the part of the application where the error happend and the parameters involved in the file DiversityCollectionError.log located in your application directory .

Queries - Overview

To search for specimen in the database you can choose 3 options:

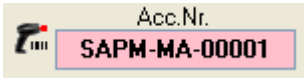


With the user defined queries, you can define any query condition.



The predefined queries are defined

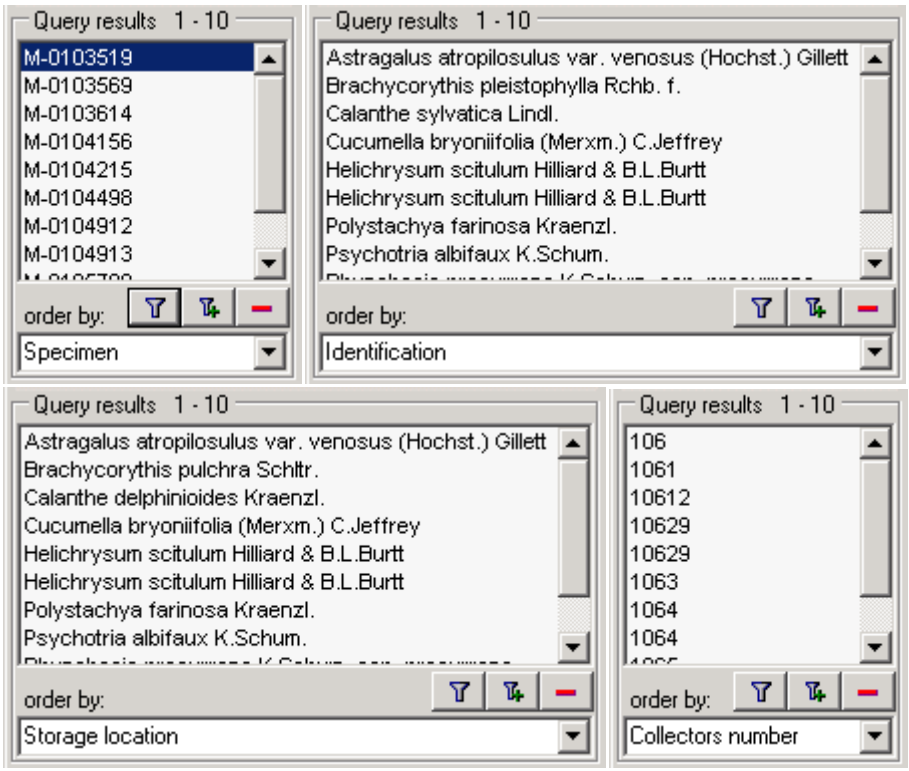
by the system administrator and are accessible via the menu.



With the scan mode you can use a barcode scanner to search for your specimen

Specimen list

The specimen list displays the collection specimen found in a query.



The specimen can be shown with their accession number, their identifications or their storage location as shown in the images above. You can restrict the maximal number of specimen together with the query options (click on the button), for example if you have a slow connection to the database. As a default the maximal number is set to 100. If the number of datasets according to your query is higher than the maximal value set in the query options this will be indicated in the header of the list.

To search for specimen, enter the restrictions in the fields for the search conditions and click on the button. The specimen found in the database will be shown in the specimen list. To add specimen with differing search conditions click on the button. If the list of items is longer than your maximal number of returned items you can browse the next items with the button. If you want to remove entries from the selected list, choose them in the list and click on the button. This will not delete the data from the database, but remove them from your query result.

AccessionNumber: One entry is shown for each collection specimen with its corresponding accession number

Identification: The last identification for every unit in a collection specimen is shown in the list. As there can be several units in one specimen several entries for one collection specimen may appear in the list.

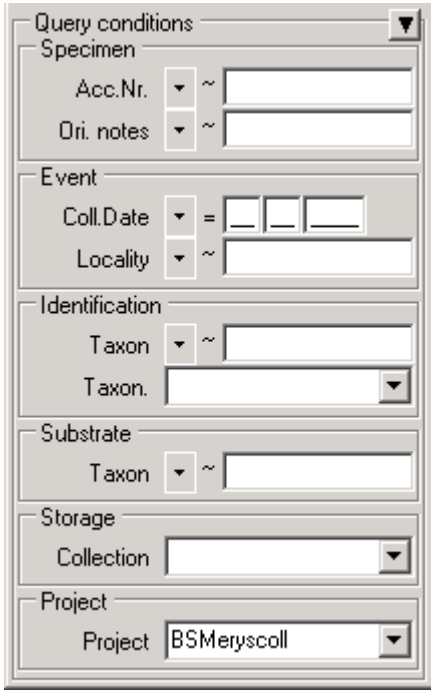
Storage location: The storage location of every sample of a specimen stored in the collections is shown in the list. As a specimen can be stored in several collections under different names several entries for one collection specimen may appear in the list.

Collectors number: The number of a collector of every sample of a specimen is shown in the list. A specimen can have several collectors each with different number. Therefore several

entries for one collection specimen may appear in the list.

Query

There are two ways to search for specimen in a collection. The options for a fast search are displayed in the main form beneath the list of the items. You can change this arrangement using the / button to place the query options on the left side of the item list.



To search for specimen, enter the restrictions in the fields for the search conditions and click on the button. The specimen found in the database will be shown in the specimen list. To add specimen with differing search conditions click on the button. To clear all entries in the query fields use the button. If the list of items is longer than your maximal number of returned items you can browse the next items with the button. To move back to the previous block of items click on the button. If you want to remove entries from the selected list, choose them and click on the button. This will not delete the data from the database, but remove them from your query result.

Within the query options you have several possibilities to specify your search restriction. Use the drop down button to change between the operator. The available operators are shown in the table below.

Operator	Meaning	Example
Text		
~	search for an entry like ...	Pinus s[iy]lvestris % (you can use wildcards)
=	search for an entry exactly equal to ...	Pinus silvestris L.
≠	search for an entry not like ...	Pinus s[iy]lvestris % (you can use wildcards)

∅	search for an entry where a value is missing ...	
-	search for an entry between ... and ...	2000 - 2005
Numeric		
=	search for an entry exactly equal to ...	2006
<	search for an entry lower than ...	2006
>	search for an entry bigger than ...	2006
-	search for an entry between ... and ...	2000 - 2005
∅	search for an entry where a value is missing ...	
Date		
=	search for an entry exactly equal to ...	20.3.2006
<	search for an entry lower than ...	20.3.2006
>	search for an entry bigger than ...	20.3.2006
Hierachy		
=	search for an entry exactly equal to ...	M-Fungi
≠	search for an entry that is not equal to ...	M-Fungi
∅	search for missing entry ...	M-Fungi
Δ	search including childs in a hierarchy ...	M-Fungi

To hide the area containing the search fields click on the button. If the search area is hidden and you want to start a new search, just click on the button.

To change the displayed fields for searching specimen click on the button to change the query options.

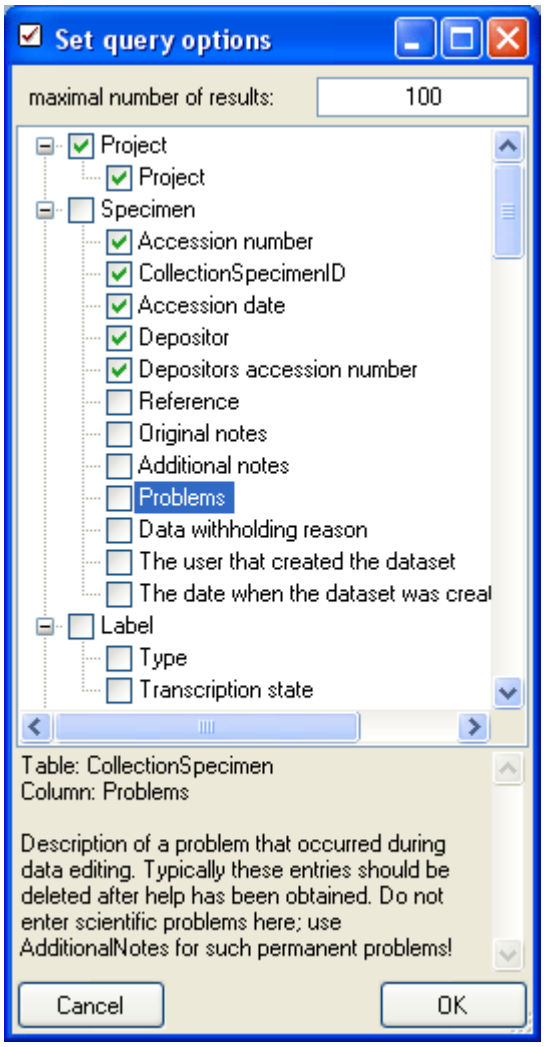
Scan mode

To search for a specimen with the help of a barcode-scanner select the **Scan mode** from the Query menu. The query part will be hidden and the field for the accession number will then be accessible for the entry with the scanner. If the field for the entry of the accession number is not activated, move the mouse to the field to activate it. Than scan the barcode and the program will search the specimen from the database

Query options

The maximal number of items shown in a query result can be set in the form for the query options. The default value is set to 100. If you have a fast connection to your database or need to see more or less results, you may change this value to any number you like.

To change the displayed fields for searching specimen click on the button. This opens a form where you can select and deselect the fields you want to be shown for searching specimen. Here you can change the maximum numbers of items that will be shown in the list. If you want certain fields to appear in the query options select or deselect them in the list.



After editing the query options click OK so the new selection will become active for the next query.

Wildcards in SQL

There are 4 different possibilities for wildcards in SQL:

% any string consisting of no, one or many characters, e.g. Pinus **%** will find anything like Pinus, Pinus sylvestris, Pinus strobus etc.

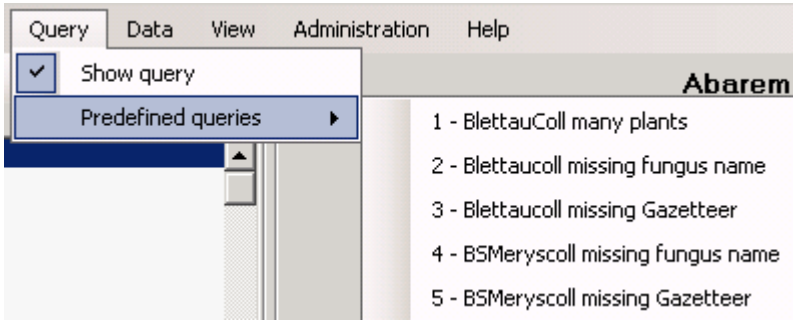
_ a single character, e.g. Pinus s**_**lvestris will find Pinus s**y**lvestris and Pinus s**i**lvestris etc.

[] any character out of a given range like **[abcde]** or **[a-e]**, e.g. Pinus s**[iy]**lvestris will find Pinus sylvestris and Pinus silvestris.

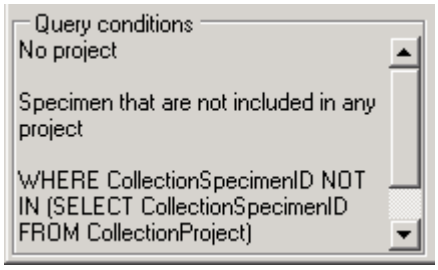
[^] any character not in a given range like **[^abcde]** or **[^a-e]**, e.g. Pinus s**[^i]**lvestris will find Pinus sylvestris but not Pinus silvestris

Predefined queries

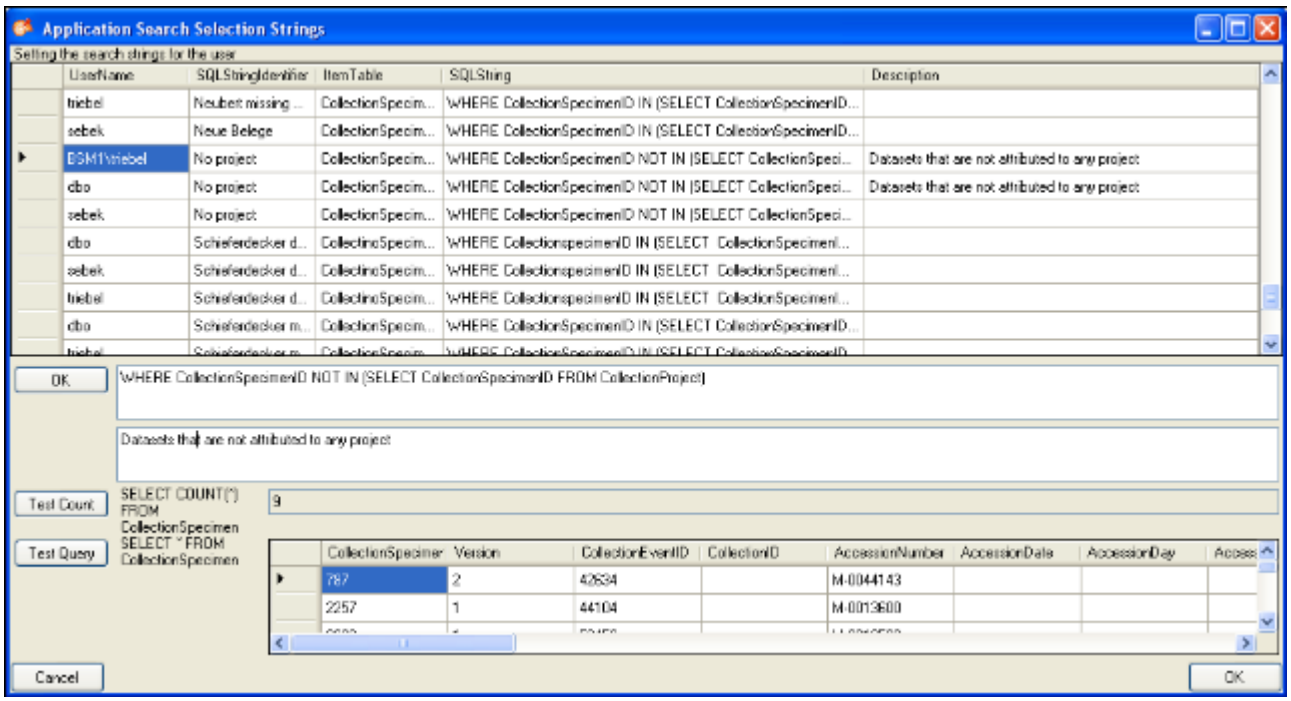
Next to the query for specimen via the query options you can define separate queries for each user. These are listed under the menu topic **Query - Predefined queries**.



If you choose one of these predefined queries, the query options will be hidden and the command of the query will be shown at the base of the specimen list. The first line shows the name of the query as choosen from the menu, the next lines contain the description followed by the part of the query command that restricts the selection of the datasets (= WHERE-clause of the SQL-statement).



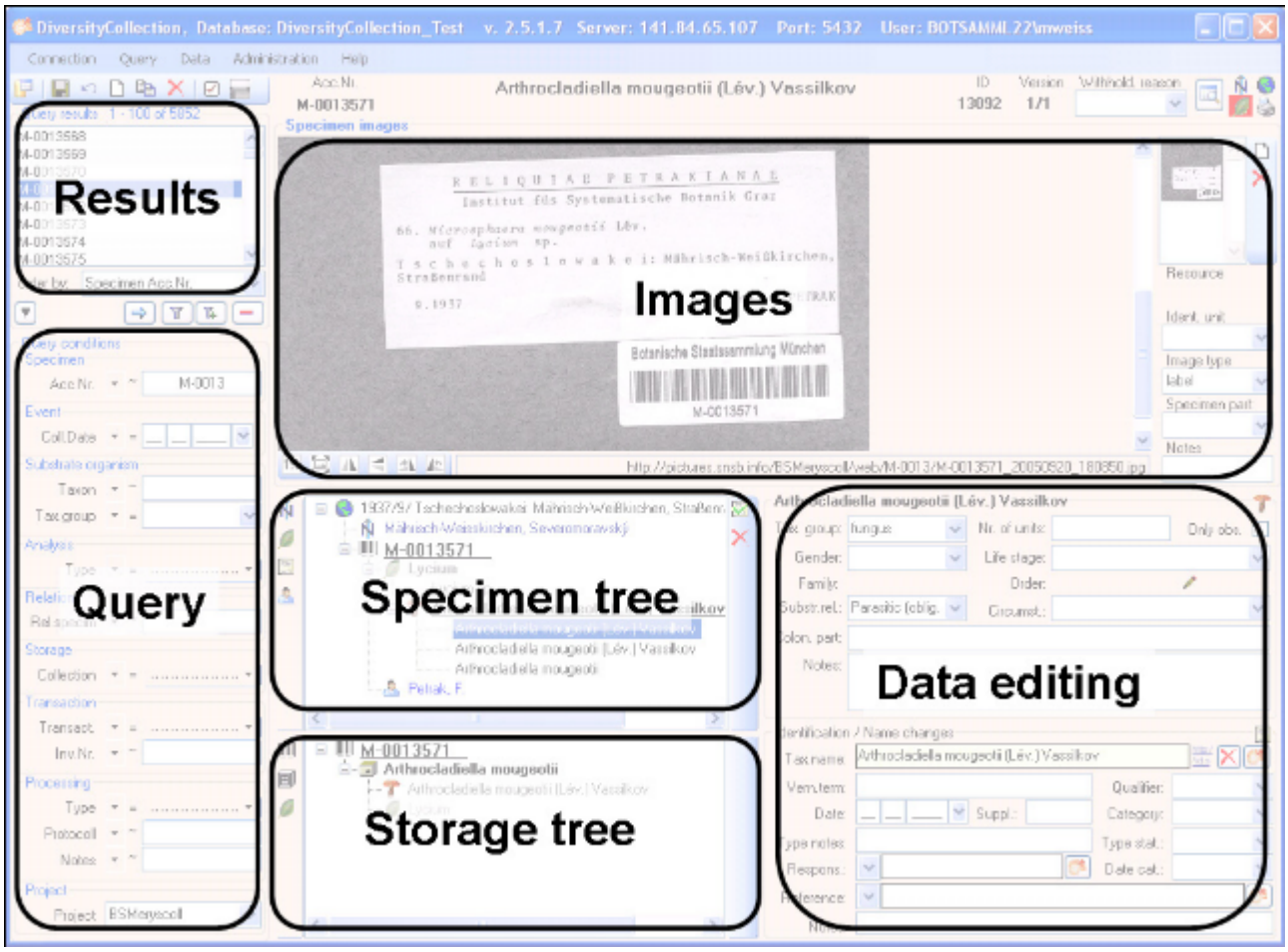
If you are an administrator you can create new queries for users. To create a predefined query choose **Administration - Queries...** from the menu. A form as shown below will open, where you can create, edit and test your queries.



In the upper field you define the WHERE-Clause of the SQL string of your query. Keep in mind that the queries can refer to different tables, depending on the order column choosen by the user. So Queries in DiversityCollection should start with the reference to the primary key of the main table (ColletionSpecimenID in table CollectionSpecimen and depending tables). In the lower field the description for the query as shown in the user interface. To test a query use the [Test count] and [Test Query] buttons.

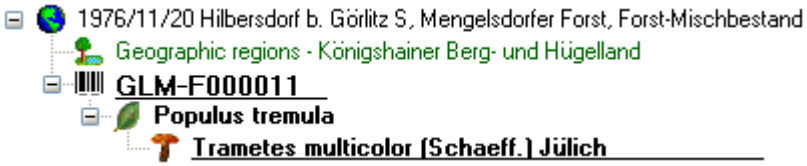
Editing the data

The main form of DiversityCollection contains two main areas. At the left you find the query and the results of this query. In the left part the data of a dataset selected in the result list is shown. The upper part of the data area shows the images, labels etc. In the lower part you find two trees that give you an overview and access to the data. The data of an entry selected in one of the trees are shown in the data editing part.

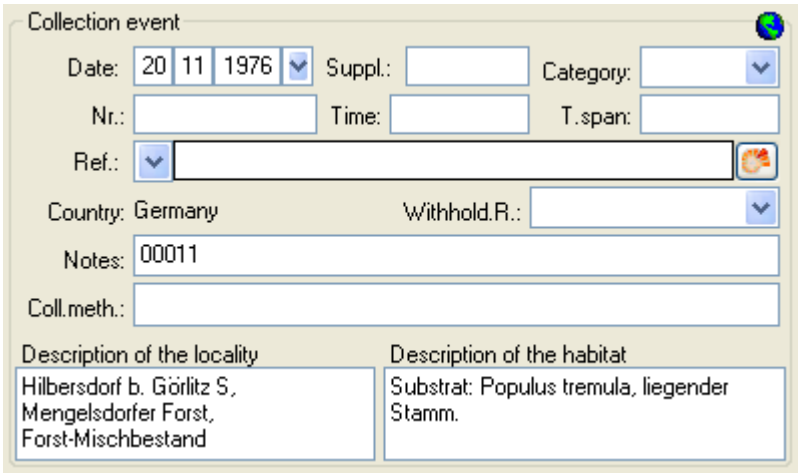


Event

Specimen stored in a collection are gathered during a collection event. This collection event keeps information about the geographic locality, the habitat, the date etc. During an collection event, several specimen may have been collected. If the specimen is not assigned to a collection event, you can assign the specimen to an existing event with a click on the button. To create a new event for the specimen click on the button. If you assign the specimen to an existing event, a form will open as shown below where you can search for the events provided in the database. Choose an event and click OK to assign the specimen to this event. In the tree, the collection event is symbolized with an icon as shown below.



To edit the data of the collection event, choose it in the tree to open the detail fields as shown below. If the collection date does not correspond to a certain day, you may use the **Suppl.** field to enter e.g. a range or a series of dates or the **T.span** field to document a certain timespan.



The text shown in the tree is composed of the date, the description of the locality. For each collection event you can enter several geographical localisations and properties. To see the locality according to the coordinates stored in the database you can check the Maps. For each collection event you can enter images related to this event.

If other specimens were collected during the same collection event, this will be visible if you show the whole hierarchy of the event series. To do this click on the button in the panel on the left of the tree. See event series for further details. Here you can move a specimen to an other event by drag and drop.

Data are stored in the table CollectionEvent.

Collection event series

If you need a hierarchical order of your collection events respectively to organise your collection events, you can do this with a collection event series. For a better differentation between events and collection event series you have a blue text in the hierarchy and the editing part and a different icon . A collection event series can contain other collection event series and collection events. Information about the geographic locality, properties of the collection site, the date etc. are stored in the collection event. To show or hide the collection event series you have two options. In the panel of the right of the tree the button will show the superior event series of the current collection event as shown below.

The button will show the whole hierarchy of the event series as in the image below.

To edit the data of an event series, select it in the tree to display the fields with the details as shown below.

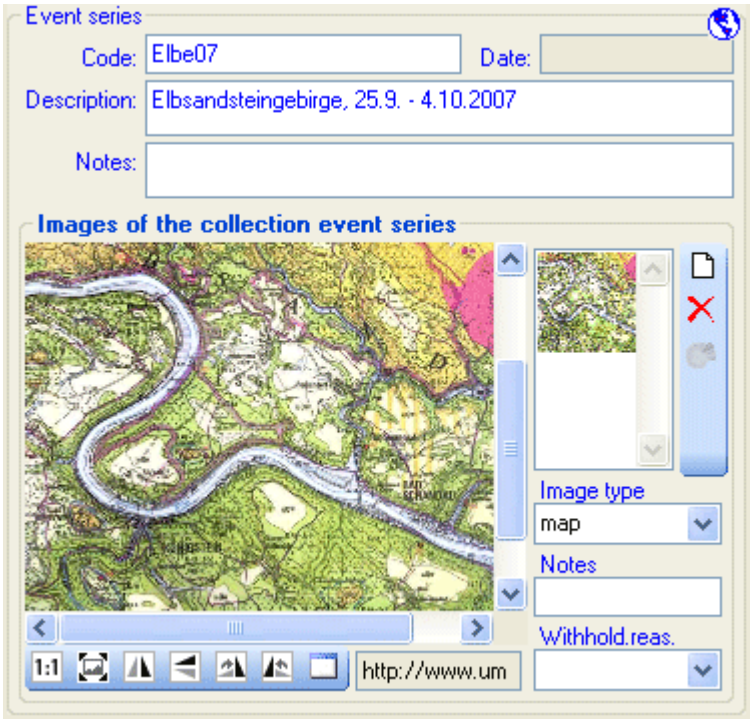
To insert a new collection event series click on the button. If there are no collection event series so far, the collection event will be placed within the new collection event series. If there are collection event series present, the new collection event series will be placed below the selected collection event series. To assign an collection event to an existing collection event series already available in the database click the icon.

To move an item within the hierarchy, just drag it with the mouse to whatever position it should be placed. Keep in mind, that specimen can only be placed in events and events only in collection event series.

If you want to delete an expedition or an event, remove all depending collection event series, events and specimen and click on the button. A specimen can not be deleted here. If you want to remove an event from an event series, click on the button to open the form for

selecting an event series. Then select nothing but simply click OK to remove the link to the event series.

The images for an event series are shown below the data of the event series as shown below.To add images to an event series, click on the button, to remove an image use the button.



To zoom a sector of the image, just drag the mouse over the image. A red square will indicate the zoomed area. To set the size of the image to the original resolution click on the button. To adapt the size of the image to the available space in the form click on the button. To change the orientation of the image use the appropriate buttons (flip horizontal, flip vertical, rotate right, rotate left). To view the image in a separate form, click on the button. If the image should not be published e.g. on the internet, enter any reason in the **Withhold. reason** - field.

If you want to change to another specimen listed in the collection event series select it in the hierarchy and click on the button.

The current specimen together with the event and all superior collection event series will be highlighted .

The data about the collection event series are stored in the table CollectionEventSeries.

Event images

Specimen stored in a collection are gathered during a collection event. To insert an image related to a collection event click on the button. If you want to delete an image, click on the button.

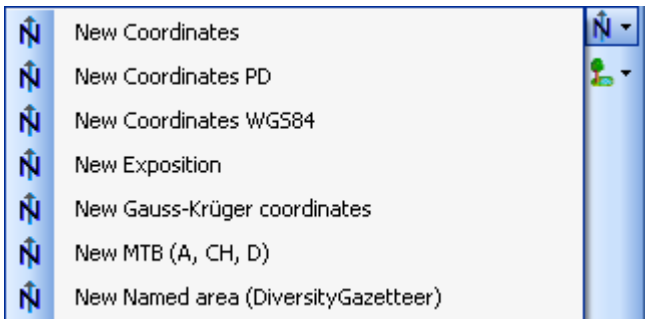
To zoom a sector of the image, just drag the mouse over the image. A red square will indicate the zoomed area. To set the size of the image to the original resolution click on the button. To adapt the size of the image to the available space in the form click on the button. To change the orientation of the image use the appropriate buttons (flip horizontal, flip vertical, rotate right, rotate left). To view the image in a separate form, click on the button. If the image should not be published e.g. on the internet, enter any reason in the **Withhold. reason** - field.

Data are stored in the table CollectionEventImage.

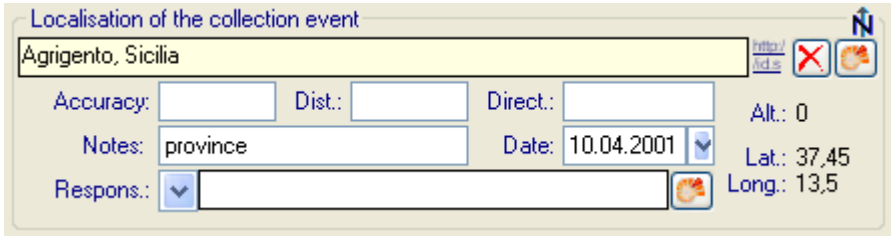
Geography

In an addition to a description of the locality you can give several localisation systems for the collection event. These entries are marked with an icon in the tree as shown below.

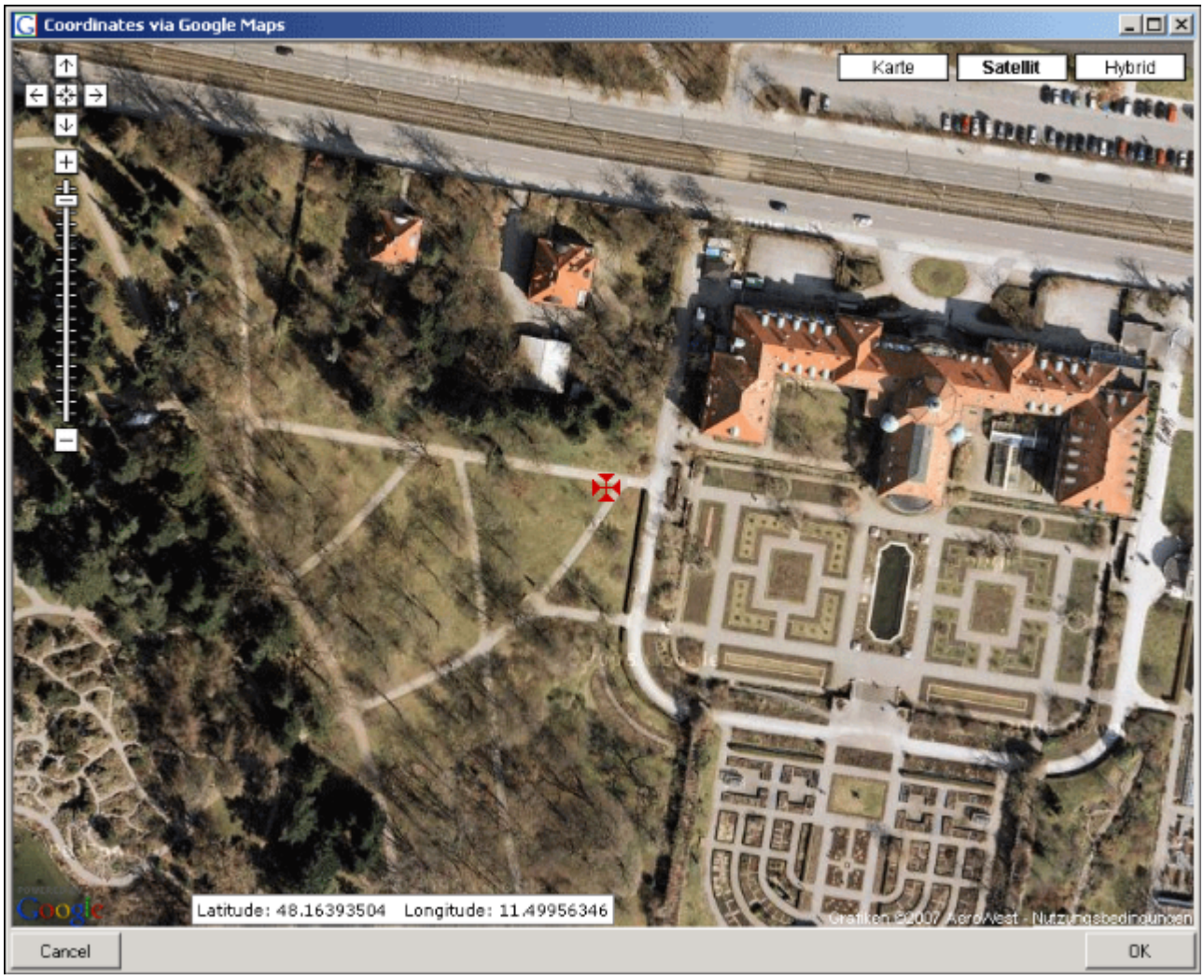
To add a new entry choose the localisation system from the drop down list as shown below. If a localisation should be removed, select it from the list and click the button.



To edit the details of an entry, select in the list to open the fields in the form on the right. The localisation system DiversityGazetteer is linked to the module DiversityGazetteer within the Diversity Workbench, providing information on geographical names as shown below.

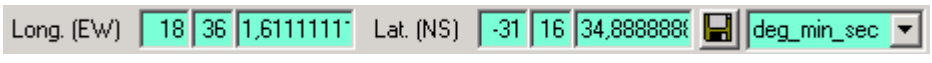


If the localisation system contains coordinates (Coordinates, Coordinates PD, Coordinates WGS84), a button will appear, that will provide you with the possibility to set the coordinates via Google Maps. Just click on the button, and a window will open where you will see the a map provided by Google Maps as shown below. If there are allready coordinates provided by a different localisation e.g. by the DiversityGazetteer, these will be taken as a starting position. So you may use an entry for a named area (DiversityGazetteer) as a approximate localisation and than additional coordinates as the more accurate localisation.



If you start with existing coordinates in your data, the system will use these as starting point. You can zoom the map, drag it to another position and change from map to satelite mode. The actual position is symbolized by the ✠ symbol in the middle of the map. The current coordinates are shown in the field at the buttom of the map. To take these coordinates in your data, just click OK.

The values for any localisation system are stored in two text fields. You can enter your vales as simple text. This may prevent any calculations with your values. So you should prefer to enter your values according to the measurement units available. Whatever antique measurement units like feet or Fahrenheit there may exist, within the Diversity Workbench measurement data are stored in units according to the The International System of Units (SI). For those who still need to use antique units DiversityCollection provides a possiblity to convert them into their modern counterparts. To enter a numeric value choose the unit you prefer from the list. The form will change as shown below.



You can than edit the values and click the button to take the changed values in your data. The system will then calculate the corresponding value for storage in the database together with the default accuracy. Correct this accuracy if you have more exact values. To indicate that the shown values are calculated from the values in the database the fields have a green background.

The altitude, the exposition or the slope may be entered as one value or as a range of two values. If you use the conversion function and have only one value, make sure, that the

second field is empty.

Available units: meter and feet.

Available units: Orientiation (N, NE, ...) and degree rel. to North.

Available units: degree and percent.

The accuracy resp. uncertainity will be calculated as an approximation in parts derived from Wieczorek, J. 2001 (MaNIS/HerpNet/ORNIS Georeferencing Guidelines. University of California, Berkeley: Museum of Vertebrate Zoology) and in parts according to Wieczorek, J., Q. Guo, and R. Hijmans. 2004 (The point-radius method for georeferencing locality descriptions and calculating associated uncertainty. International Journal of Geographical Information Science. 18: 745-767). The unit of the accuracy will allways be the unit stored in the database according to the *Système international d'unités* (SI) resp. degrees for angles.

The data for the geography are stored in the table CollectionEventLocalisation.

Event - Maps

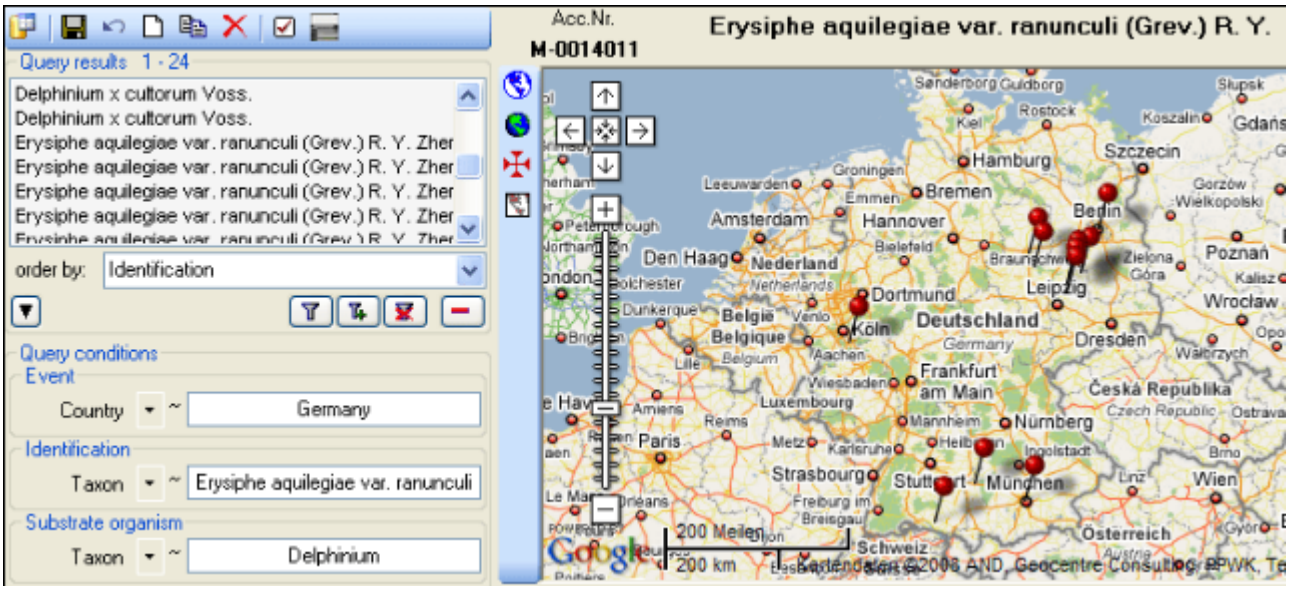
If there are coordinates available for your collection site of your locality using

To use this service, you need access to the internet. Click on the button in the control to show the maps. Click on the button to see the location of your event or if the event is part of a series the button to see all locations of the event series. The site can be displayed either as a satellite image

or a geographical map. If you click on one of the needles marking the locations of the events, a window will show you the description of the event.

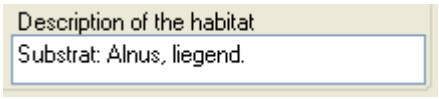
To generate a map you need coordinates stored in the table CollectionEventLocalisation.

To show a distribution map as shown below of all the specimen found in your query, use the button. This functionallity is limited in the number of specimen that can be depicted, so it may be neccessary to restrict your query.

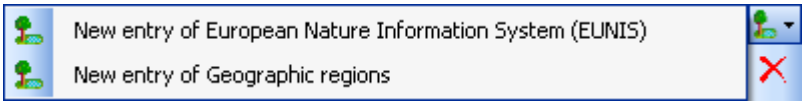

Query results 1 - 24
Delphinium x cultorum Voss.
Delphinium x cultorum Voss.
Erysiphe aquilegiae var. ranunculi (Grev.) R. Y. Zher
Erysiphe aquilegiae var. ranunculi (Grev.) R. Y. Zher
Erysiphe aquilegiae var. ranunculi (Grev.) R. Y. Zher
Erysiphe aquilegiae var. ranunculi (Grev.) R. Y. Zher
order by: Identification
Query conditions
Event
Country ~ Germany
Identification
Taxon ~ Erysiphe aquilegiae var. ranunculi
Substrate organism
Taxon ~ Delphinium
Acc.Nr.
M-0014011
Erysiphe aquilegiae var. ranunculi (Grev.) R. Y.
Hamburg
Bremen
Amsterdam
Hannover
Berlin
Den Haag
Nederland
Dortmund
Deutschland
Leipzig
Dresden
Köln
Frankfurt am Main
Nürnberg
München
Strasbourg
Paris
Wien
Česká Republika
Österreich
Schweiz
200 Meilen
200 km
Google

Habitats and properties of the collection site

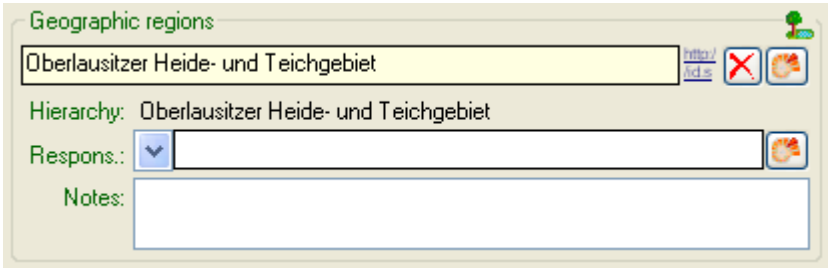
For the description of the habitat select the collection event or an existing collection site property . The description is entered in the section Description of the habitat.



To enter a new property of the collection site, select the collection event or an existing collection site property and than choose the type of the property you want to enter.



The list is dependent on the availability of terminologies for site descriptions. Details for the property can be entered if you choose the item in the hierarchy. The person responsible for the entry as well as notes can be entered in the corresponding fields as shown below.



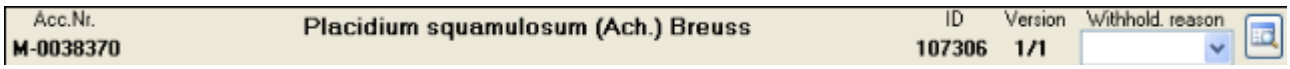
To delete entries use the button. You can enter a value by either typing the name of the collection site property or by selection from the module DiversityScientificTerms. To search for properties from this module click on the button. As responsible user the name of the current user will be inserted. You may change this by either typing or selection from the module DiversityAgents.

The data concerning the habitats are stored in the table CollectionEventProperty.

Specimen

Specimen are the entities stored in a collection. For handling the data use the appropriate buttons (new specimen, copy, delete). See the data section for further details. Directly attached to the specimen are the data about accession, label, exsiccatae, notes, reference and the availability. A specimen may be composed of several units. Any problems concerning the specimen should be entered into the problems field.

The header of the main form shows some important parameters of the selected specimen. On the left side the **accession number** is shown. In the center you find the last identification of first the identification unit within the display order. The next field shows the internal **ID** (field CollectionSpecimenID in table CollectionSpecimen) and the **Version**. For details about the version of a dataset see the version topic. The availability of a specimen can be changed by entering an appropriate reason in the field **Withholding reason**.



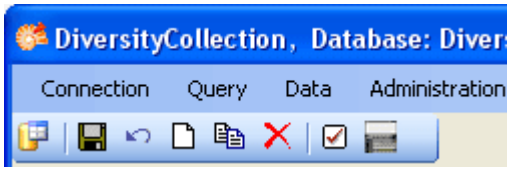
To inspect the history of a specimen click on the button. For further details see the history section.

With the buttons you can control the upper part of the form with the maps, the images for the collection event and the specimen as well as the label print. The buttons with the corresponding sector visible in the form are depicted with a red background. If there are images available and they are hidden, the backgroud will turn to yellow.

The data are stored in the table CollectionSpecimen.

Data

The controls to handle datasets in the database are located in the left upper part of the form as shown in the image below. A step-by-step introduction for the creation of new datasets is provided in the tutorial.

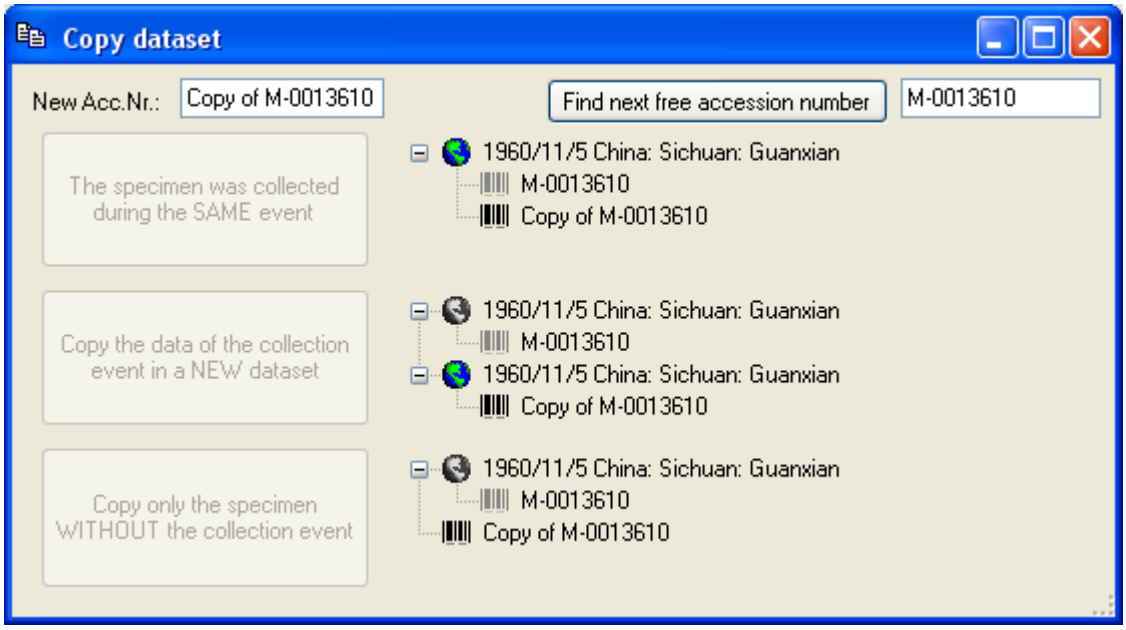


To **save** the changes in a dataset click on the button. If you change to another dataset, the current changes will be saved automatically.

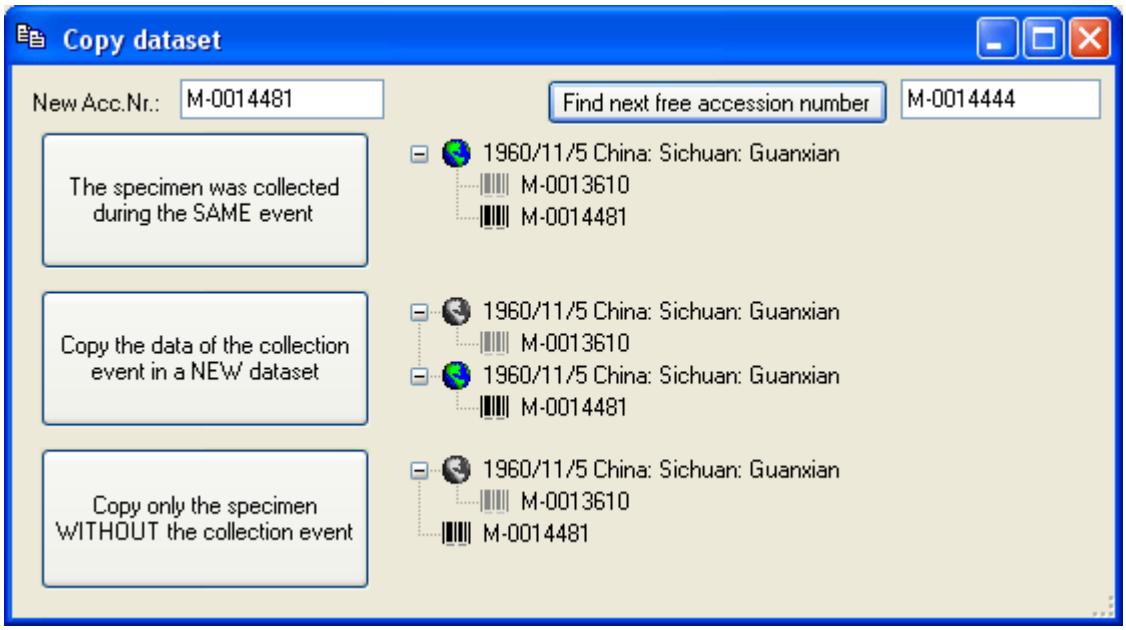
To **undo** the changes in a dataset click the button. This will recover the original data unless the changes had been saved or changes were done in certain tables or hierarchies were the data must be stored to display the hierarchy.

To create a **new** entry in the database, click on the button below the search result listbox. This will create an entry with the specimen and show it in the list.

To **copy** a specimen, choose it from the list and click on the button . If the specimen you want to copy is linked to a collection event, a form will open and provide you with several possibilities for the copy as shown below.



If the field for the new accession number contains a string like Copy of ..., use the **[Find next ...]** button to search for the next free number for the string in the second field. If the program finds a free number, it will be used as shown below.

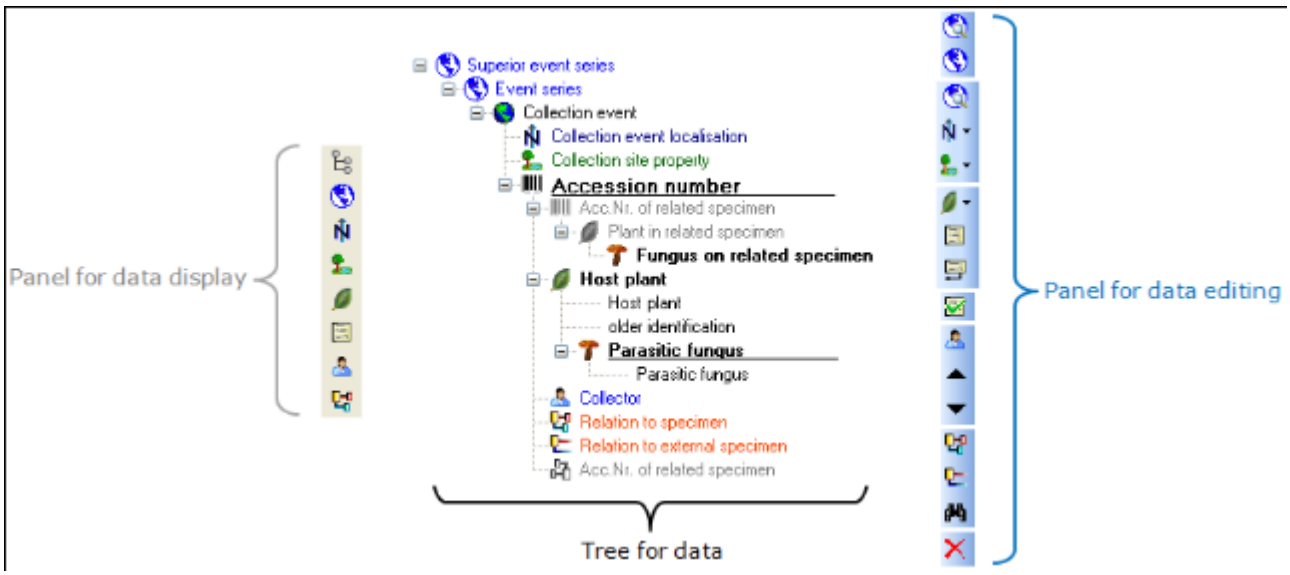


The first option will take the new dataset in the same collection event as the original data, so no new data for a collection event will be generated. The second option will create a copy of the original data for the collection event and connect the copy of the specimen with the new event. The last option will only copy the specimen data and establish no connection to an collection event.

To **delete** a dataset click on the button .

Tree for the specimen

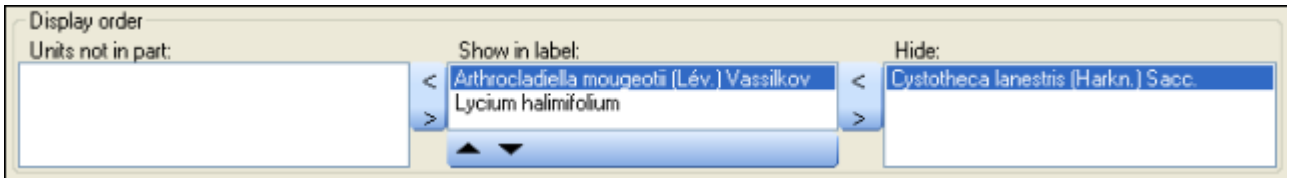
The upper tree in the form provides an overview for all the data linked with the specimen. With the buttons in the left panel you can hide or show certain nodes in the tree. So if for example you do not want to see the collectors, just click on the button and they will be hidden in the tree. The button will change to a grey version and the background will turn to yellow , to show you that there are hidden data of the collectors. If the specimen is part of a collection event and this collection event is part of a collection event series than with the first two buttons you can switch between two modes for the event series. If you click on the button only the superior event series will be shown. To shown the whole hierarchy of the event series, click on the button.



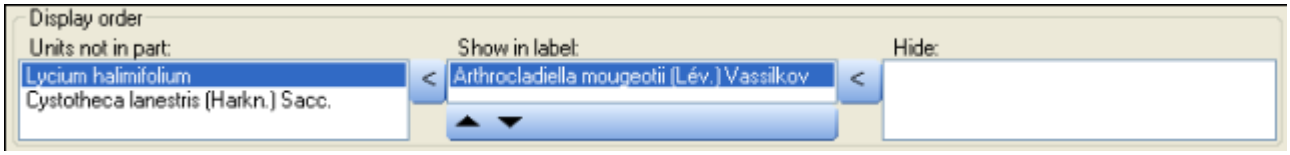
The panel on the right of the tree is for editing the data, like for example the button will insert a new collector. For further details see the special sections.

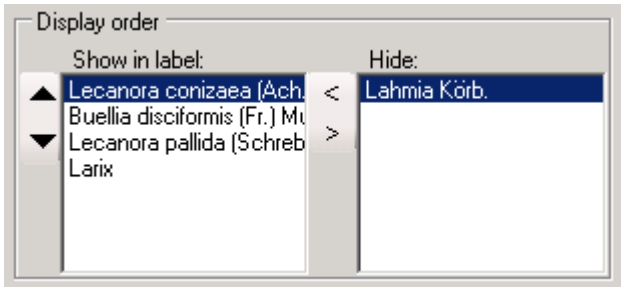
Display order

The display order defines the sequence in which the units within this specimen will appear on e.g. a label . The first unit will be printed in the header of the label, others are included in the text below. You can change the display order by using the ▲ ▼ buttons. If a unit should not appear on the label, transfer it to the hide list using the > button. The < button returns it to the list that will be shown in a label. The first unit can not be transferred to the hide list.



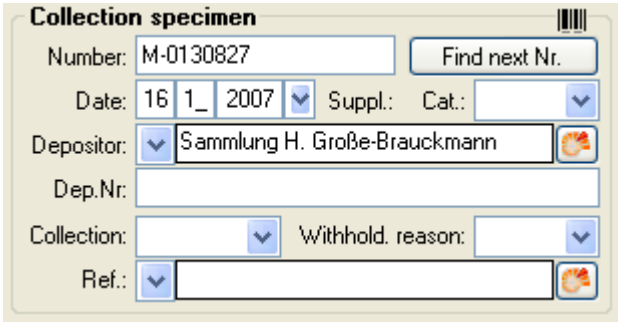
For a part of a specimen in addition to the display order you can specify if an organsim is present within the part.





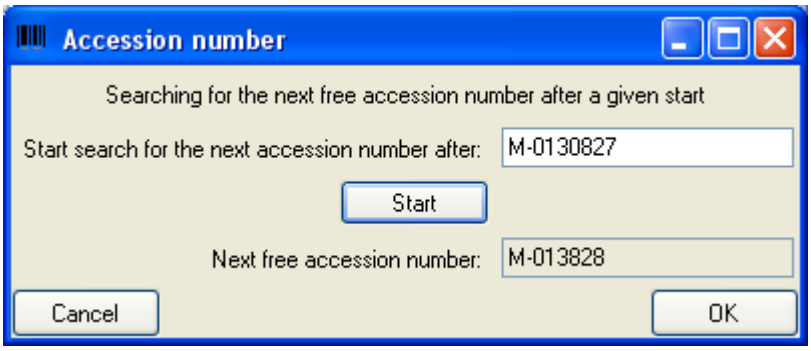
Accession

The accession of a specimen in a collection is documented with its accession number and if available the date of the accession.



If the specimen was received from an other collection or collector, this may be documented with the name of the collector (Depositor) and the number in the original collection (Dep.Nr).

If a new specimen is entered you can use the Find next button the search for the next free accession number. A form will open as shown below, where you can search for a free number. The search will start with the number you give as a starting point.



The system will try to find the next free number on the basis of the accession numbers available in the database. Click OK to use the new number for you specimen.

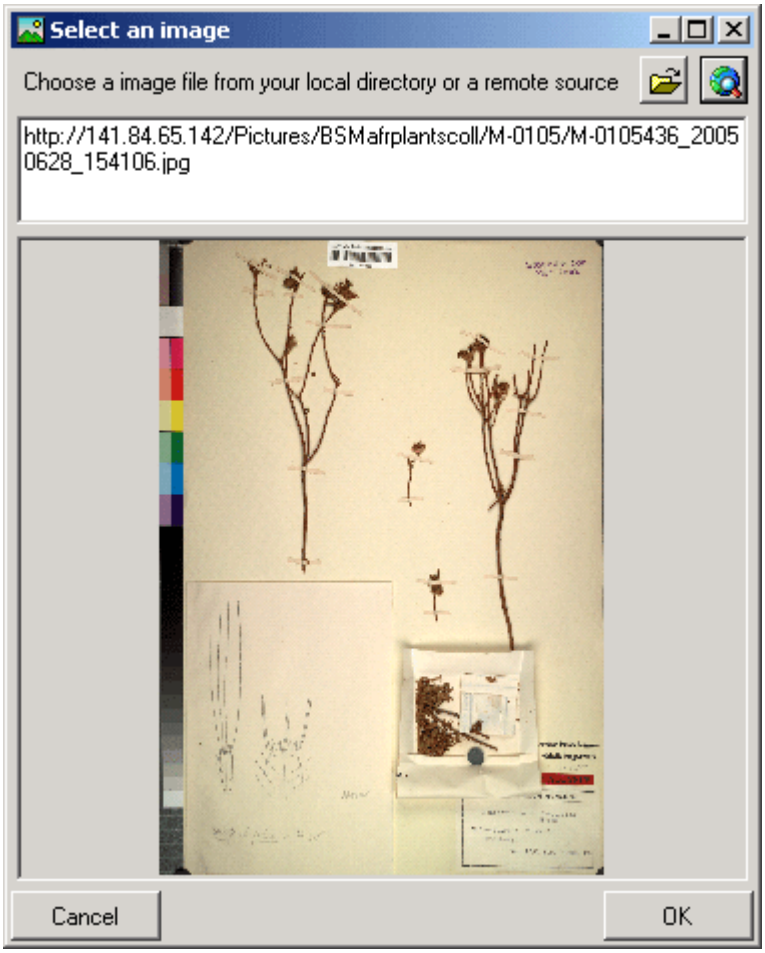
Data concerning the accession and deposition are stored in the table CollectionSpecimen.

Categories of the date:

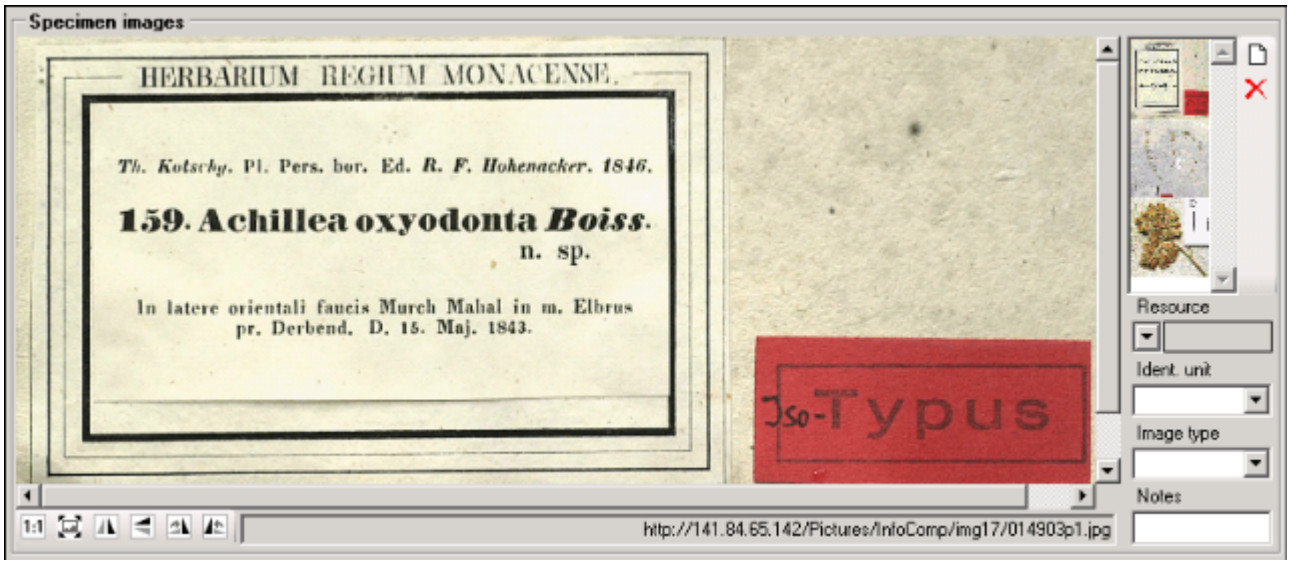
actual = The given date is the real date
collection date = The date when the specimen was collected
estimated = The given date was estimated by the user
system = The given date was generated by the system

Specimen Image

To see the specimen images activate the icon in the image selector . If you choose the option **View - Show existing images** from the menu, the images will automatically be shown. Each specimen may be documented with several images. The images may be stored local with its path or as a reference to the module DiversityResources within the Diversity Workbench. To enter a new image, click on the button. A window will open where you can enter the path and file name of the image. Click on the button to search in your local directories or on the button to search for a web address. The selected image will be shown in the preview.



To delete an image, select it from the list and click the button.

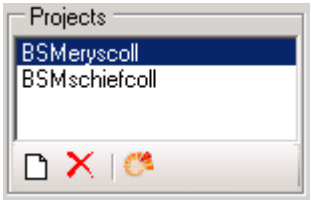


To zoom a sector of the image, just drag the mouse over the image. A red square will indicate the zoomed area. To set the size of the image to the original resolution click on the 1:1 button. To adapt the size of the image to the available space in the form click on the button. To change the orientation of the image use the appropriate buttons (flip horizontal, flip vertical, rotate right, rotate left). To view the image in a separate form, click on the button. If the image should not be published e.g. on the internet, enter any reason in the **Withhold. reason** - field.

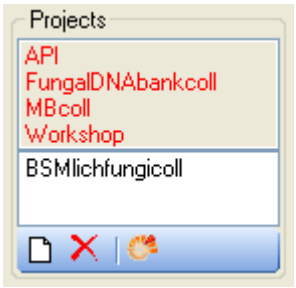
Data are stored in the table CollectionSpecimenImage.

Projects

Every collection specimen can be assigned to any number of projects. To assign a specimen to a project click on the button. To remove it from a project, select the project from the list and click the button.



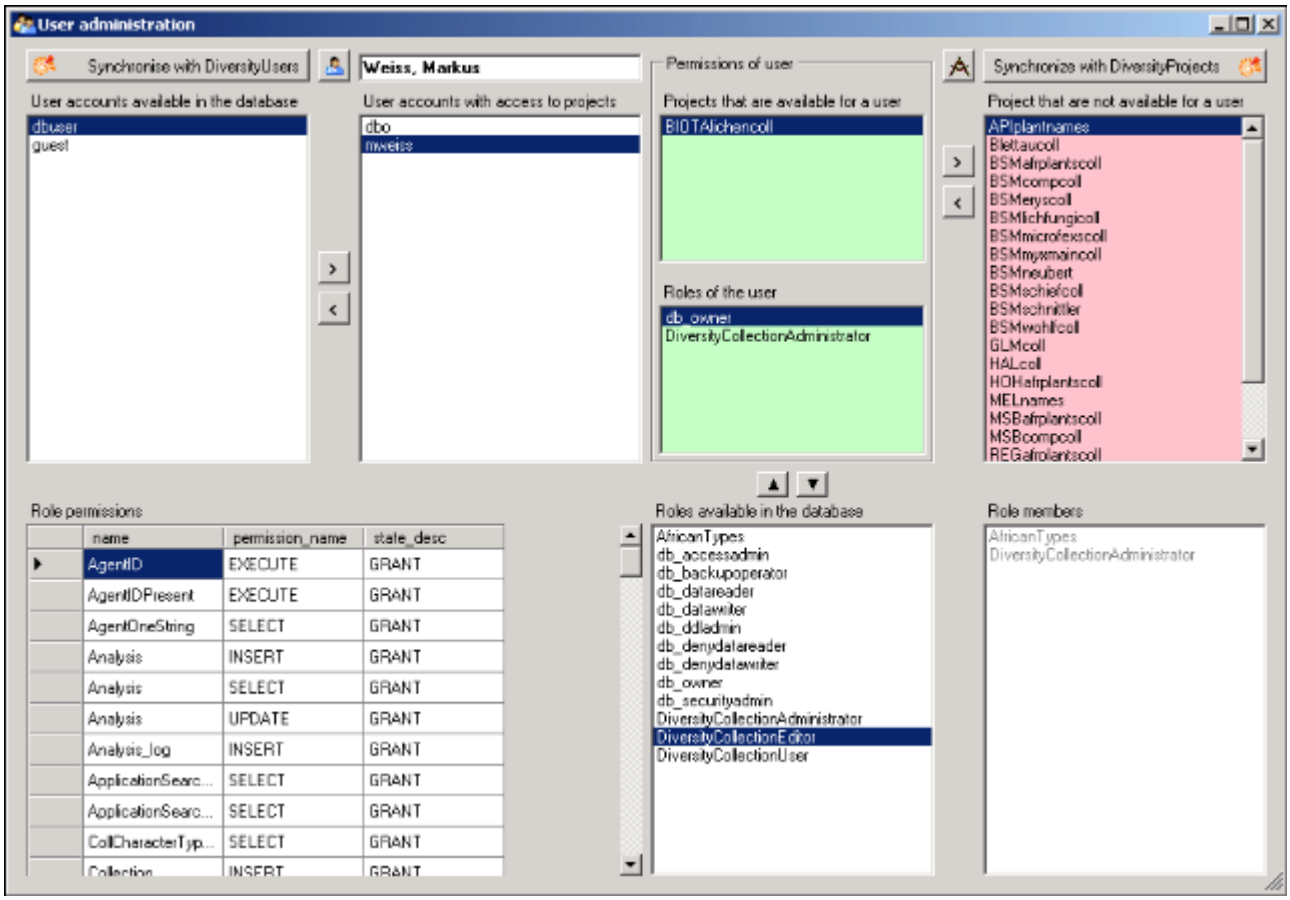
If there are projects, where you have no access to, these will be listed in a separate list at the top as shown below.



Data are stored in the table CollectionProject.

Details upon the projects within the DiversityWorkbench are stored in the database DiversityProjects. To open a project to see further information upon a project click on the button. To edit details in the projects you need the application **DiversityProjects.exe** in your application directory and access to the database DiversityProjects. To synchronize the projects listed in DiversityProject you can use the synchronize functionality in the user administration form as shown below. If DiversityProjects is not available you can create a new project with the button. If DiversityProjects is available, use the synchronize functionality

.


User administration
Synchronise with DiversityUsers
Weiss, Markus
Permissions of user
Synchronize with DiversityProjects
User accounts available in the database
dbuser
guest
User accounts with access to projects
dbo
mweiss
Projects that are available for a user
BIOTAlichencoll
Project that are not available for a user
APIplantnames
Blettaucoll
BSMafrplantscoll
BSMcompcoll
BSMeryscoll
BSMichfungicoll
BSMmicrofexscoll
BSMmyxmaincoll
BSMneubert
BSMschiefcoll
BSMschnittler
BSMwahlcoll
GLMcoll
HALcoll
HOHafrplantscoll
MELnames
MSBafrplantscoll
MSBcompcoll
REGafrplantscoll
Roles of the user
db_owner
DiversityCollectionAdministrator
Role permissions
name
permission_name
state_desc
AgentID
EXECUTE
GRANT
AgentIDPresent
EXECUTE
GRANT
AgentOneString
SELECT
GRANT
Analysis
INSERT
GRANT
Analysis
SELECT
GRANT
Analysis
UPDATE
GRANT
Analysis_log
INSERT
GRANT
ApplicationSearc...
SELECT
GRANT
ApplicationSearc...
SELECT
GRANT
CollCharacterTyp...
SELECT
GRANT
Collection
INSERT
GRANT
Roles available in the database
AfricanTypes
db_accessadmin
db_backupoperator
db_datareader
db_datawriter
db_ddladmin
db_denydatareader
db_denydatawriter
db_owner
db_securityadmin
DiversityCollectionAdministrator
DiversityCollectionEditor
DiversityCollectionUser
Role members
AfricanTypes
DiversityCollectionAdministrator

Notes and Problems

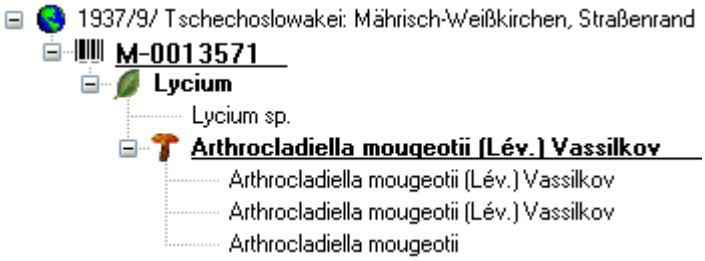
To enter notes or problems connected with the specimen select it in the hierarchy. The data form will then open the corresponding fields. The **Original** notes are the notes found on the label of the specimen, made by the original collector or from a later revision. **Additional** notes are those made by the editor of the specimen record, e. g. doubtful identification or locality.

In the **Problems** area enter the description of a problem that occurred during data editing. Typically these entries should be deleted after help has been obtained. Do not enter scientific problems here. Use Additional notes for such permanent problems!

The data are stored in the table CollectionSpecimen.

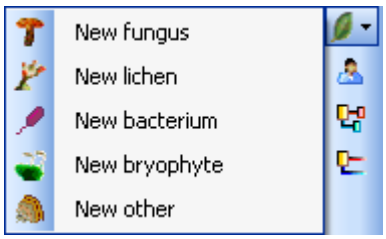
Organisms and Identifications

Each specimen can contain several organisms where each organism may have been identified several times as shown in the image below.

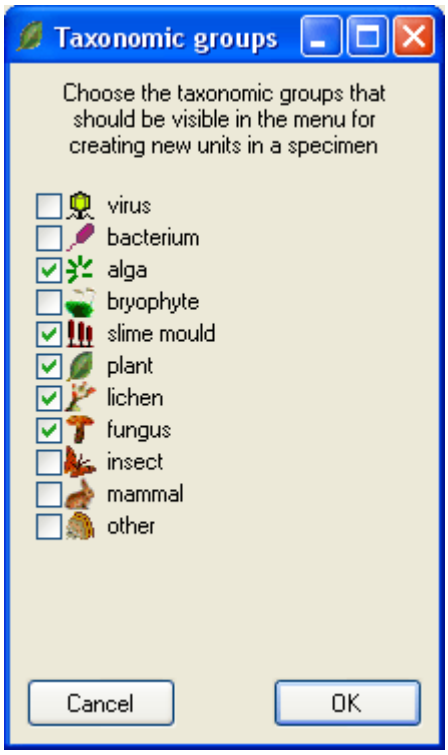


Identification unit

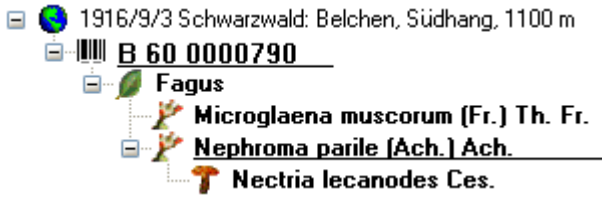
The items or organisms in one collection specimen are regarded as identification units. One specimen can contain several identification units, e.g. an insect (1) feeding on a fungus (2) growing as a parasit on a plant (3). To add a new organism choose from the taxonomic group this new organism belongs to from the drop down menu as shown below.



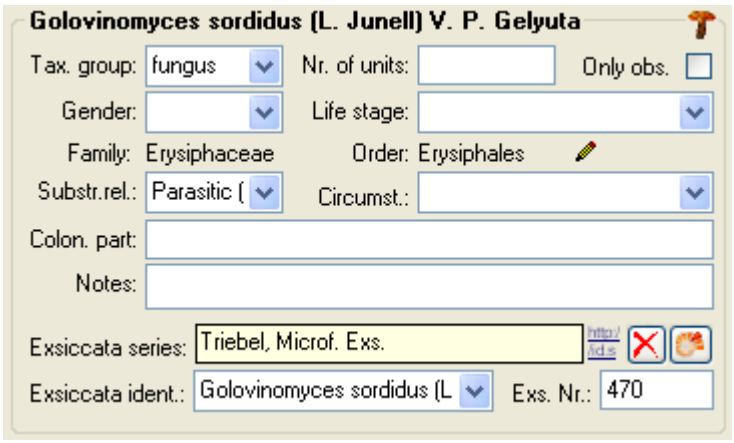
The taxonmic groups shown in the drop down menu are specified under Administration - Taxonomic groups ... . A form as shown below will open where you can specify which groups should be shown.



The organisms within the specimen are shown in the tree. To edit the relations between the organisms just do this by drag and drop. The name of the organism under which the specimen is stored in the collection is underlined. To delete an organism select it in the tree and click on the button.



To enter details about the organisms like the taxonomic group, select it in the tree. The form show at the left of the tree provides the fields related to the organism as shown below.



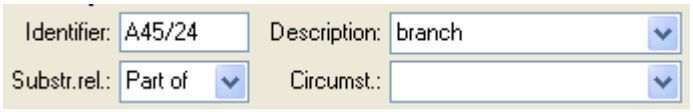
Taxonomic Hierarchy - Family and Order

In the last line you find the family and the order of the taxon. These entries are either set when linking to a taxonomic database or manually when no link to a taxonomic database exists.

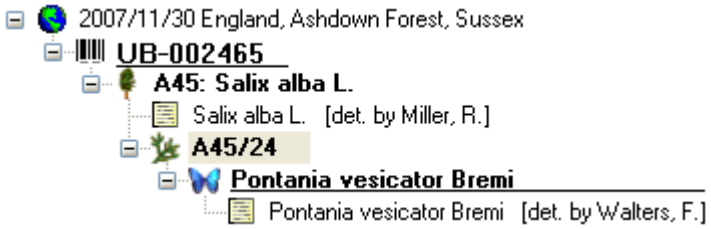
If no link to a taxonomic database exists you may enter the the family and the order after clicking on the button. To transfer these entries to other specimen with the same genus use the maintenance functions as described under Maintenance - family and order.

Parts of units

If a unit contains parts that have to be documented, you can do this by setting the relation to the substrate to "Part of".



In the tree the backgroud of a part will be grey as shown below for the unit-tree und the tree depicting the storage of a sample. The name of the part will correspond to the identifier set for this part and the icon will correspond to the description of the part provided you choose one of the preset options contained in the drop down list. You may of course enter any description for the unit. A unit that is part of another unit can not get an identification. Identifications are restricted to the main unit.



For details about exsiccatal series and analysis see the related topics. You can sort your identification units e.g. for display on a label with the display order. Each identification unit can have several identifications.

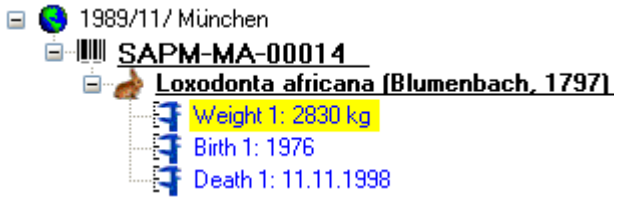
The data for the organisms are stored in the table IdentificationUnit.

Relations to the substrate organism

Association = in association with (unspecific statement if no further information about the interaction between 2 organisms is available)
Endophytic in = endophytic fungi growing in plants without causing symptoms (possibly mutualistic or opportunistic paras.)
Growing on = substrate statement: fungus growing on 2nd organism, living stages directly observed (i.e. assoc. not due to diaspores alone)
Isolated from = substrate statement: fungus isolated from 2nd organism, no living stages observed (i.e. possibly due to surface contamination)
Lichenization = mutualistic symbiosis between a fungus and algae (= 2nd organism)
Mutualism = mutualistic symbiosis, both partners profit from each other
Mycorrhiza of = mycorrhizal association of a fungus with a plant (= 2nd organism)
Parasitic (accid.) = parasitic on an 'accidential host' (in the sense of Nannfeldt): with strong infection pressure, another plant growing between infected plants may become infected.
Parasitic (facul.) = facultatively parasitic on host (= 2nd organism)
Parasitic (oblig.) = obligately parasitic on host (= 2nd organism)
Parasitic (quest.) = questionably parasitic on host (= 2nd organism), parasitic interaction is presumed, but doubtful/unverified
Parasitic on = the taxon is parasitizing its host (regardless of type, i.e. holo-/meta-/hemi-biotrophic, or pertho-/necrotrophic)
Parent/Child = parent/child association (i.e. in the literal genetical sense, not in an abstract sense)
Pollinator of = organism 1 is the pollinator of a plant (organism 2)
Predator/Food = organism 1 is the predator of organism 2 (i. e. 2 is food of 1)
Sibling = both organisms are siblings (have the same parents)

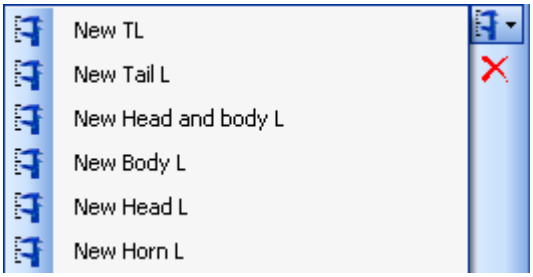
Analysis

The organisms in a specimen can be analysed by analysis types defined in the database. In the tree the analysis entries are symbolized with an icon as shown below. Only the types of analysis that were assigned to the group of the organism can be selected. An analysis is allways refered to an organism and may be referred to a part of a specimen. An analysis that refers to a part of a specimen will be shown in the tree for the parts as shown in the examples below.

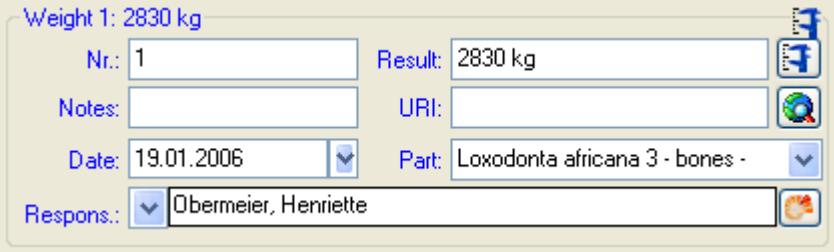


To show or hide the analysis items in the tree use the button in the left panel. This button has 3 states. In the default state the analyis entries will be shown as above. If you click a second time on the button it will change to the hierarchy mode and the entries will be shown as in the tree below. The last state will hide the entries.

To insert a new analysis select the organism which has been analysed in the upper tree. Than select the type of the analysis from the drop down list as shown below.

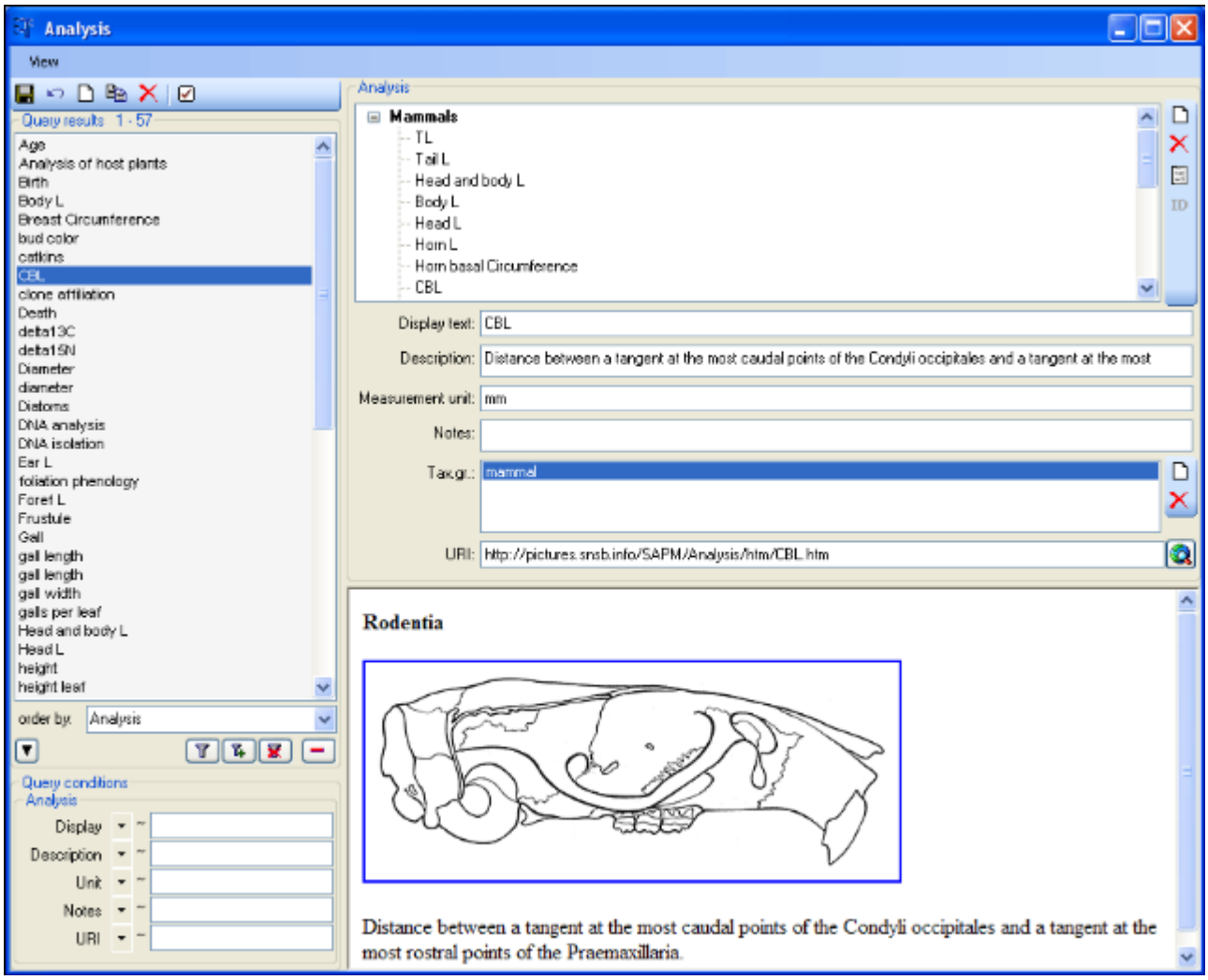


To delete an analysis, select it in the tree and use the button in the panel on the left. To edit the details of an analysis, select it in the tree to enter the fields as shown below. To enter or inspect an URI given for a single analysis, click on the button. If an analysis was performed at a part of the specimen, you can document this by choosing the respective part from the list. The analysis will then as well be shown in the tree for the parts of the specimen.

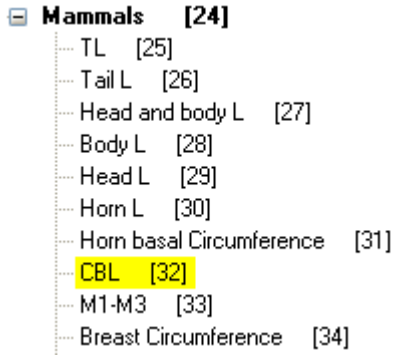


If an analysis was performed at a part of the specimen, you can document this by choosing the respective part from the list. The analysis will then as well be shown in the tree for the parts of the specimen. You can as well directly choose the organism in the part tree and create the analysis for the organism in this part in the part tree.

If you need further information about an analysis click on the button to open the form for the analysis administration. This will open a form where as an administrator you can edit the analysis types used in your collection. This form is also accessible via the menu entry **Administration - Analysis...**



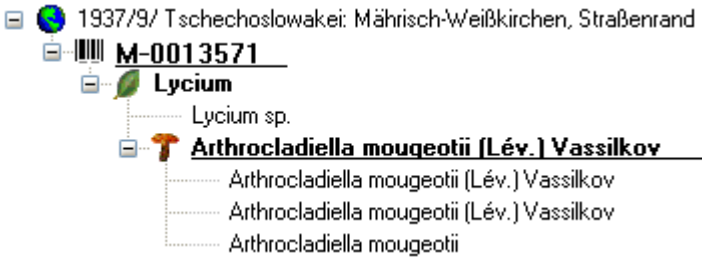
If you want to see the ID's of the datasets, click on the **ID** button. Than the ID's will be shown as in the image below.



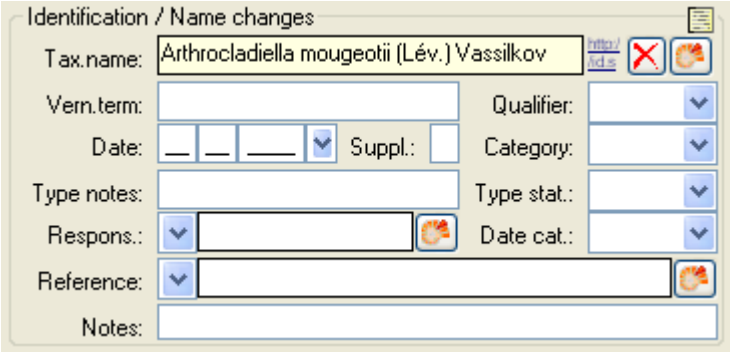
The types of an analysis are restricted by the taxonomic group of the organism or object to be analysed. Edit the list of taxonomic groups that can use a certain analysis using the and button. For details about handling the data see the data section. If you want to or inspect an URI given for an analysis type, click on the button. In the window below the URL related to the analysis can be shown.

Identification

Each organism or unit may have been identified several times. The identifications are listed in the tree under the organism as shown in the image below. To create a new identification choose the organism from the tree and click on the icon. The last identification will always be taken as the valid one and set at the top of the list. To insert an older identification at the base of the list click on the icon.



To hide / show the identifications in the tree click on the icon on the right panel next to the tree. To enter a confirmation of a identification, choose it in the tree and than click on the button. To delete an identification select it in the tree and click the button. To enter details for an identification, choose it in the tree. You than can enter the details in the form opening on the right side of the tree as shown below.



The taxonomic name of an organism can be selected from either the names already available in DiversityCollection or from the module DiversityTaxonNames. To choose from the local names type the beginning of the name and click on the drop down button to get a list of the available names. You can use wildcards according to SQL.

To select a name from an external database for taxonomic names, click on the button. A form will open where you can search for a taxonomic name.

If the identification is uncertain use one of the available qualifiers:

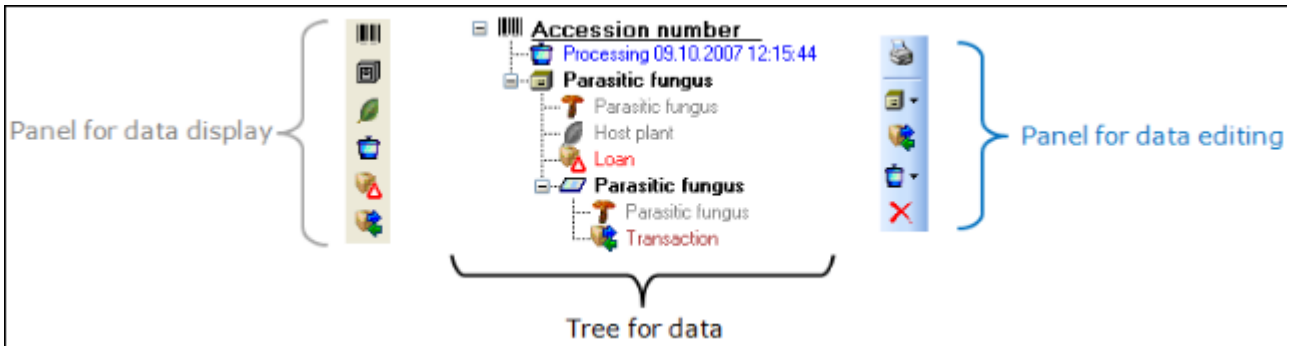
?	Unspecified doubtful identification (aff. or cf., or applicable taxonomic rank not known)
aff. forma	'affinis' = similar to, but probably not the named forma
aff. gen.	'affinis' = similar to, but probably not the named genus
aff. sp.	'affinis' = similar to, but probably not the named species
aff. ssp.	'affinis' = similar to, but probably not the named subspecies
aff. var.	'affinis' = similar to, but probably not the named variety
agg.	aggregate
cf. forma	Doubtful identification of forma or forma specialis. Example: 'Ech. ferruginea cf. f. rubra'
cf. gen.	Doubtful genus identification. Examples: 'cf. Abies alba', 'cf. Abies', '?Abies', 'Abies?'
cf. sp.	Doubtful species identification (genus identification certain). Example: 'Abies cf. alba'
cf. ssp.	Doubtful identification of subspecies. Example: 'Abies alba cf. ssp. alba'

cf. var.	Doubtful identification of variety. Example: 'Echinonema ferruginea var. campestris'
s. l.	sensu latu
s. str.	sensu strictu
sp.	Only the genus is given, the information relates to a single, unspecified species
sp. nov.	Only the genus is given, it is assumed that the unidentified species is new
spp.	Only the genus is given, the information covers several unspecified species

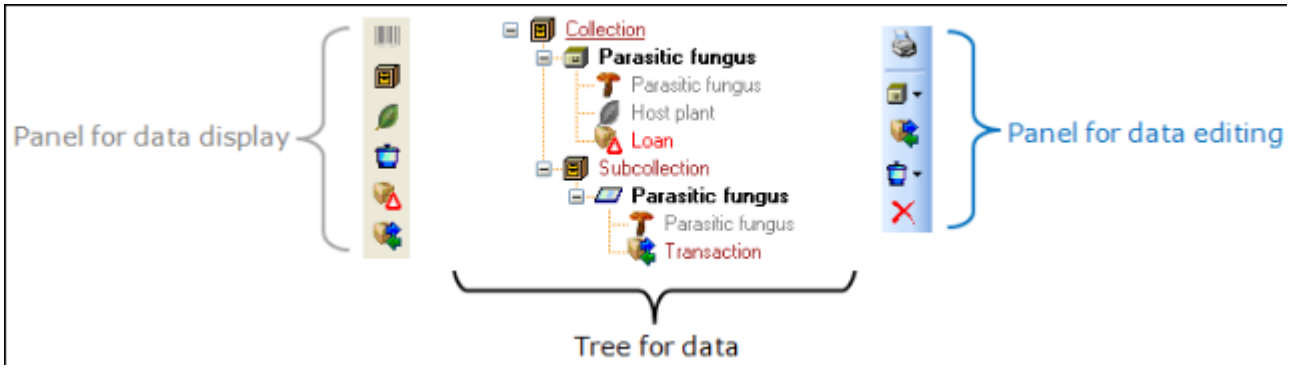
In the database the data for the idenfication are stored in table Identification.

Storage and trees for the parts of the specimen

In the tree at the base of the form the parts of the specimen with their related data are shown. This tree depicts the parts either in accordance to their relation to each other in the specimen tree (click the icon in the panel on the left) ...



... or depending on the collections where the parts are stored (click the icon in the panel on the left).

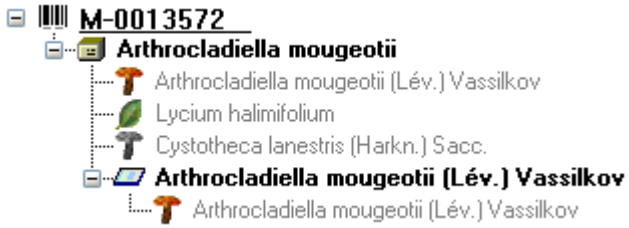


With the buttons in the left panel you can hide or show certain nodes in the tree. So if for example you do not want to see the processings, just click on the button and they will be hidden in the tree. The button will change to a grey version and the background will turn to yellow , to show you that there are hidden data of the processings.

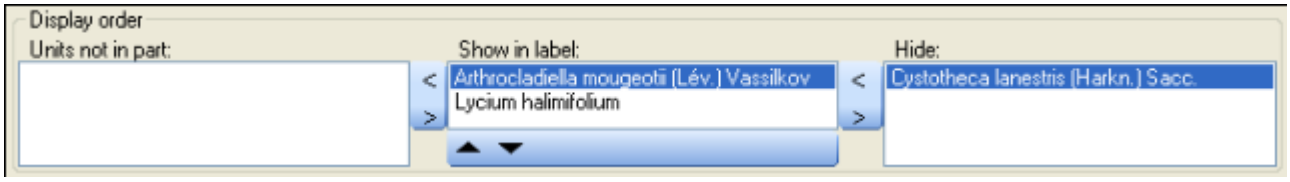
The panel on the right of the tree is for editing the data, like for example the button will insert a new transaction. For further details see the special sections.

Parts of the specimen

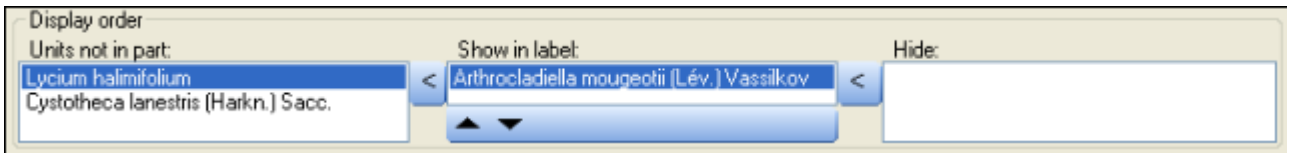
A specimen can contain several parts that may be stored in several collections. These parts may have diverse material categories, e.g. herbarium sheets, microscopic slides etc. The parts are shown in the part tree under the main tree. This tree depicts the parts either in accordance to their relation to each other in the specimen tree (click the icon in the panel on the left) or depending on the collections where the parts are stored (click the icon in the panel on the left) - see images below. To add a new part, use the drop down field in the panel on the right of the tree and choose the material category of the new part. You can set the default collection for new parts. If the default collection is not set, you will be asked to name the collection of the new part. If the part is derived from another part, select this in the tree before creating the new part. The new part will then be a child of the old part. You can later edit these relations by drag & drop. If you want to create a copy of a part, use the button in the panel right of the tree.



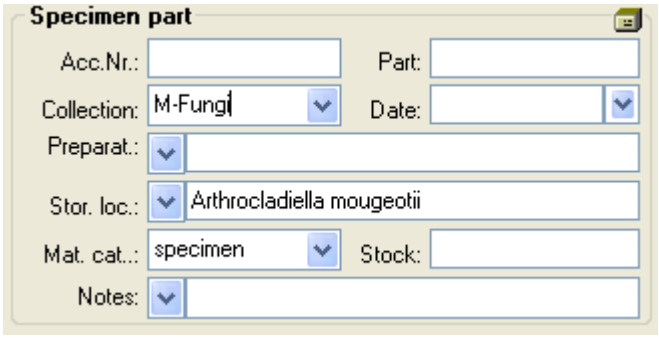
If the organims found in a sample are present, they are shown in the tree. To edit the presence, select the specimen part in the tree and edit the **Display order** as shown below.



In the expample above, all organisms are present in the specimen, but the fungus *Cystotheca lanestris* was choosen to not appear on the label and transferred to the **Hide** list. This is symbolized by a grey icon in the tree. The slide, symbolized by the icon, was created from the sample, so in the specimen tree it is shown as a child of the specimen symbolized by the icon. Concerning the organisms, only the fungus Arthrocladiella is found on the slide. So in the tree it is the only organsism shown as child of the slide. In the display order the two other organisms appear in the **Units not in part** list (see below).



To move the organisms between the lists use the **>** and **<** buttons. In the list **Show in label** you can change the sequence of the organisms with the ▲and ▼buttons in the panel at the base of the list. The organism at the top will be the one that will be taken as the main organism, e.g. for printing a label. To edit the details of a part, select it in the tree to open the fields as shown below.



The identifier, e.g. the taxonomic name under which the specimen can be found in the collection is entered in the field **Storage location**. You can search for this entry in the query.

To enter the **storage location** you can use the drop down button to select from the identifications within the specimen. The drop down button for the Notes will provide you with a list of all entries in this field, filtered with your entry in the text field. You may use wildcards like "%" or "_" (see wildcards).

In the collection tree, the specimen are to shown with their dependence upon each other, but where they are located with the collections. To get informations about a collection just select it in the tree. If you have the necessary rights, you can use the button to open for the administration of the collections. This form is as well accessible under the menu entry **Administration - Collections...**

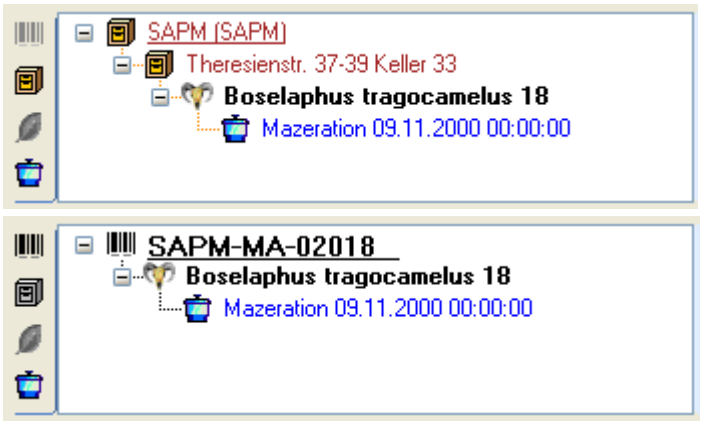
The material category of a specimen part is indicated by the icon in the hierarchy:
bones: bones or skeleton from vertebrates
cultures: living cultures of organisms
drawing: original line or color drawing
herbarium sheets: capsules or sheets as stored in a botanical collection
icones: icones, images etc. stored in a botanical collection
micr. slide: glass plate with sections of specimen for microscopic studies
specimen: specimen stored in a collection

Specimens can be included in a transaction and every stored part of a specimen can be treated with several processings.

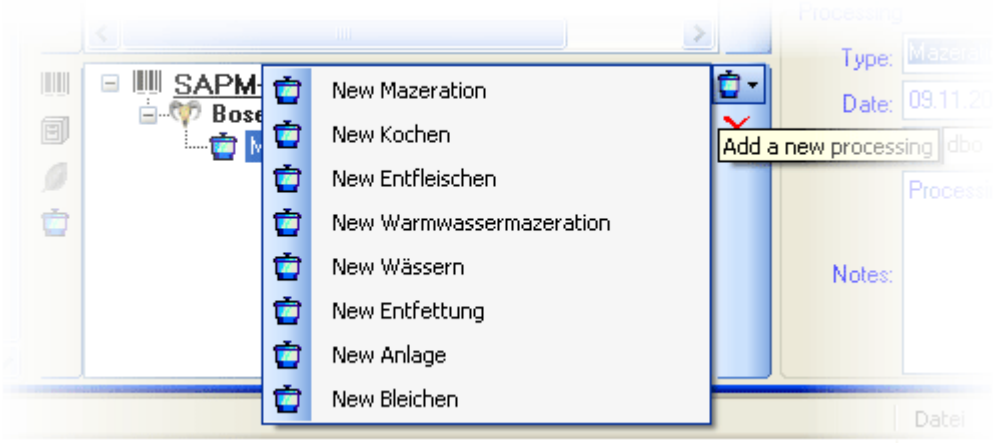
Data are stored in the table CollectionSpecimenPart.

Processing

Every specimen can be processed, e.g. for preparation or preservation. Any number of processings can be stored for a specimen or a part of the specimen. In the hierarchy for the parts the processings are indicated by the icon and a blue text.If there are any processings this will be indicated by the icon in the tool bar. You can hide or show the processings in the hierarchy with a click on the icon. The images below show a processing in either the collection or the specimen tree for the parts.

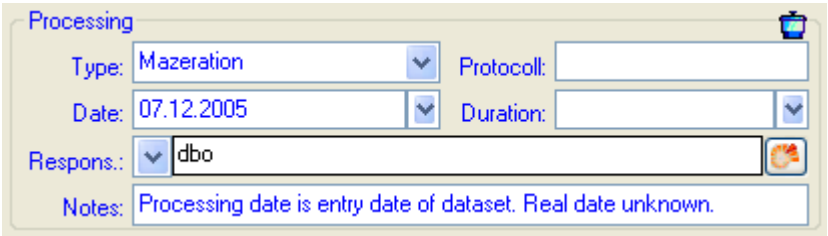


You can enter a new processing for either the whole specimen or a part of the specimen. To do this select the specimen or a part in the hierarchy and then choose a processing from the list as shown below.

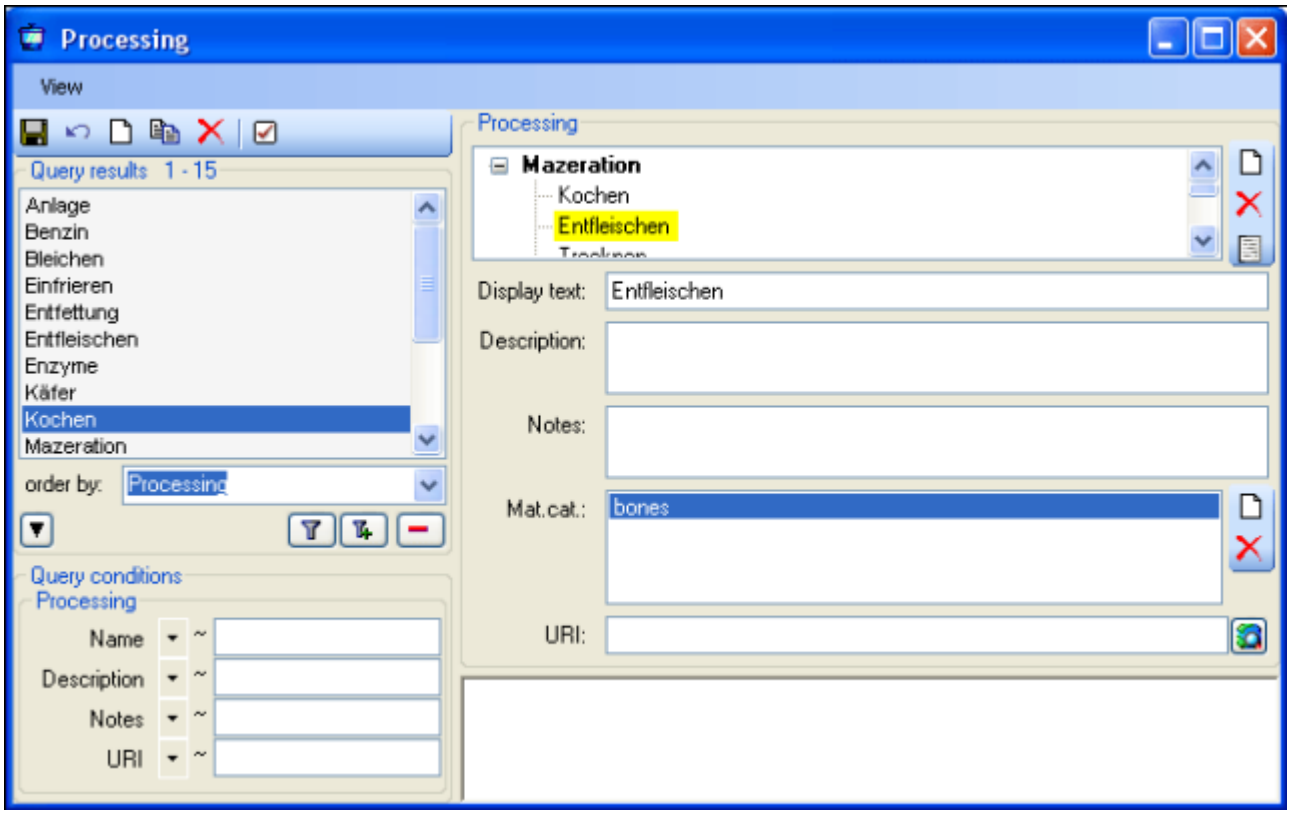


To delete a processing select it and click on the button. To edit the data for a processing select it in the hierarchy. The form will show you the fields related to the processing as shown below.

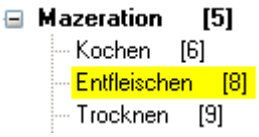
In the form select the type of your processing from the list.



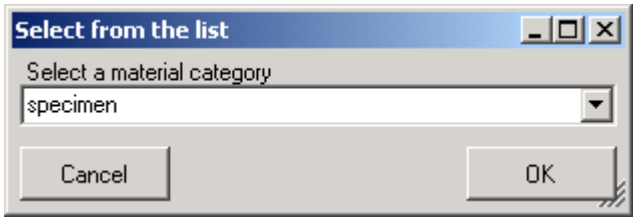
As an administrator you can edit the processing types under the menu entry **Administration - Processing...**



If you want to see the ID's of the datasets, click on the **ID** button. Than the ID's will be shown as in the image below.



n this form you can define the processings used in your collection. To add a new processing click the button, to delete a processing select it and click the button. The processings are related to material categories within the database. To add a new material category that should be connected to the selected processing, click on the button in the panel right of the material categories. A form will open, where have to select the material category that should be entered for the processing.



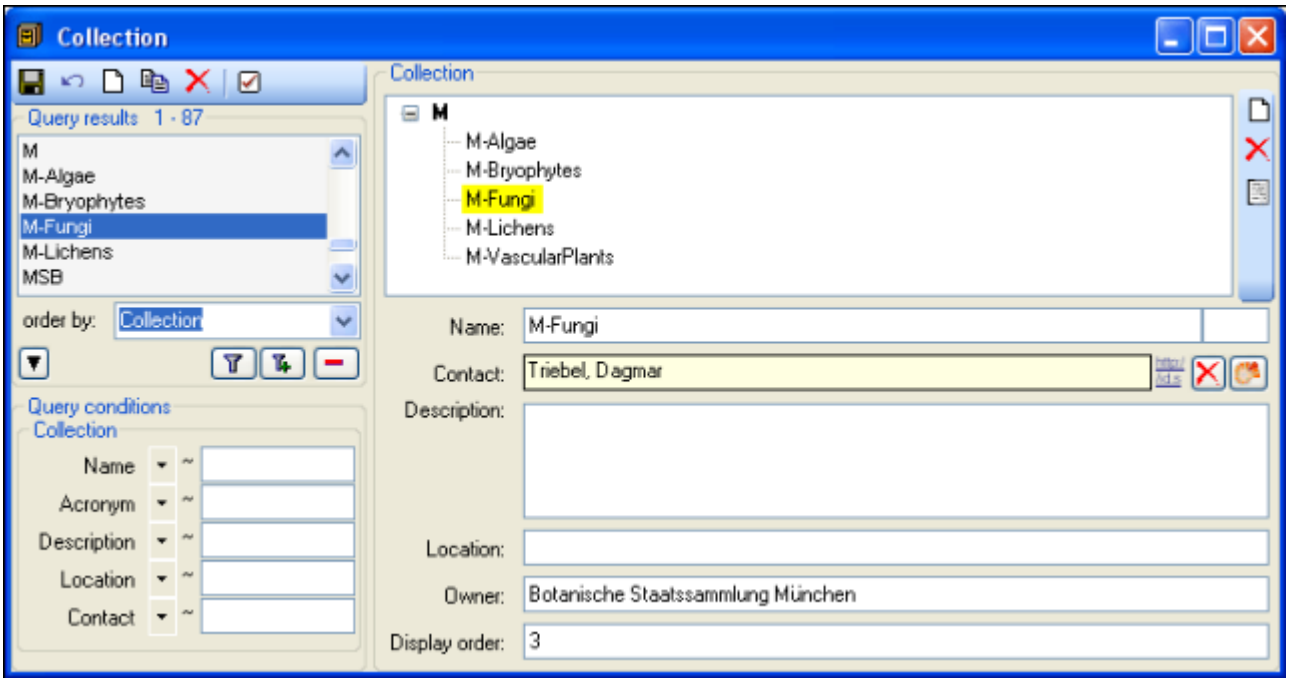
The data of the processing are stored in the table CollectionSpecimenProcessing .

Collection

The organisms in a specimen can be stored in several collections. To see the place where a specimen is stored use the lower tree and click on the icon to select the display according to the collections as shown below.

To edit the details of a collection, select it in the tree to display the datafields. At the base of the fields you find the default collection, used for the creation of new specimen parts. Use the button to edit this entry.

If you need further information about a collection click on the icons. To open the form for the collection administration choose **Administration - Collections...** from the menu (this menu entry is only available for administrators). This will open a form as shown below where as an administrator you can edit the collections used in your database.

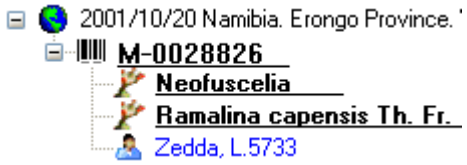


The specimens stored in a selected collection are listed in the specimen list. You can hide / show this list using the / button. To change to a specimen from this list click on the button.

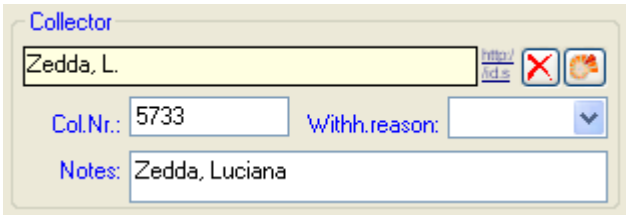
For details about handling the data see the data section. Data of the collections are stored in the table Collection.

Collector

The people or groups responsible for the collection of the specimen are stored in the table CollectionAgent. There may be several collectors for one specimen. In the tree the collectors are symbolized by the icon as shown below.



To hide or show the collectors in the tree use the icon in the left panel. To insert a collector, choose either the specimen or an existing collector in the tree and then click on the icon in the right panel. To enter detail for a collector, select it in the tree. The form on the right of the tree will then display the fields for the details as shown below.



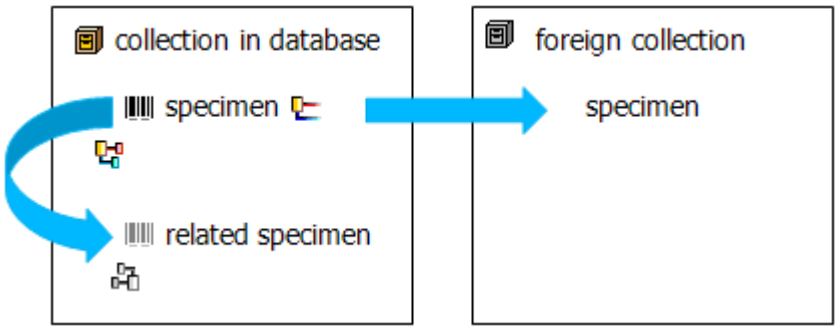
If you want to refer to the module DiversityAgents for the collectors click the button. The sequence of the collectors (e.g. for print on a label) is stored with the CollectorsSequence which you can change by clicking on the and buttons in the panel on the right of the tree.

To delete a collector, select him from the list and click the button.

You can search for the collectors number in the query.

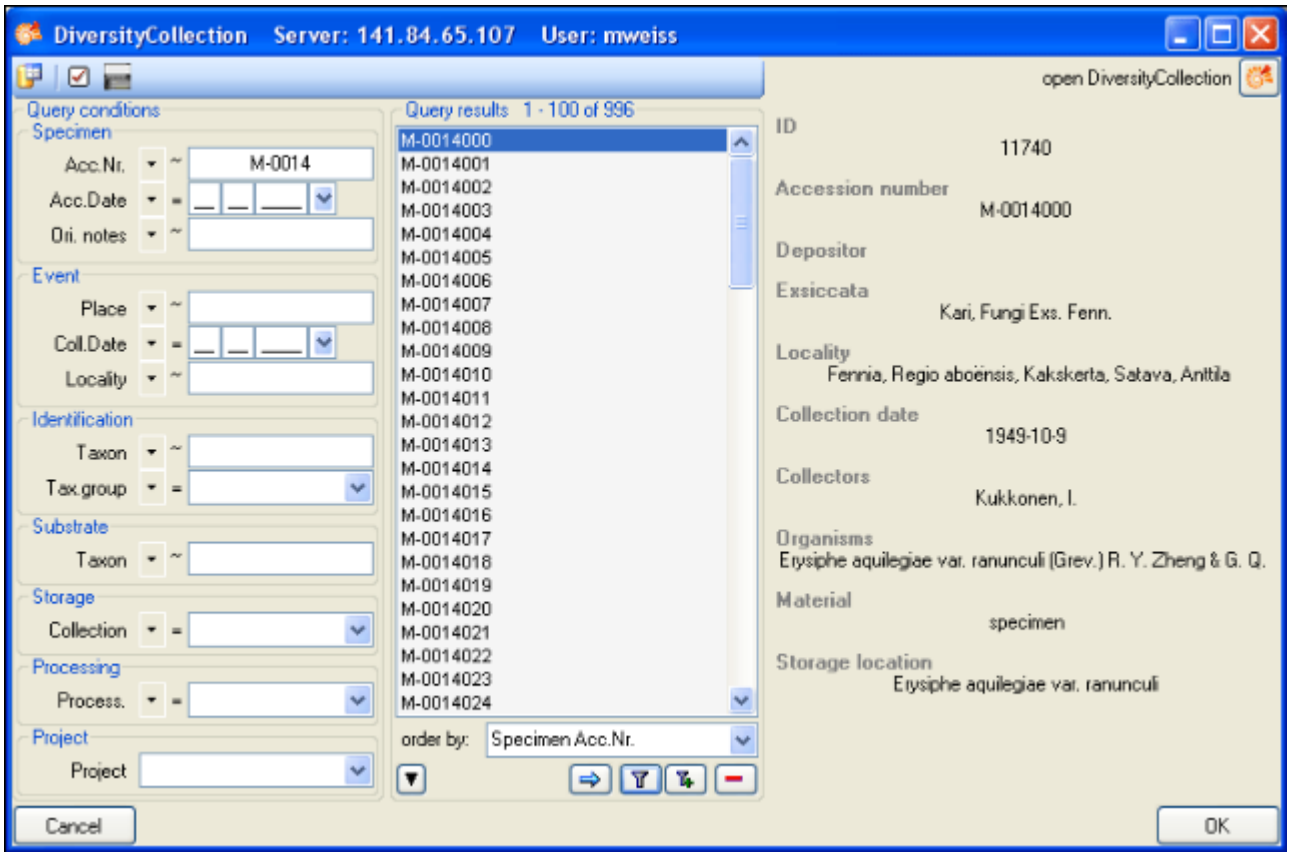
Relation

Each specimen may have relations to other specimens. This can be documented by adding relations to these specimens. To add a relation select the specimen in the tree (symbolized by the barcode). Then you can either add a relation to a local specimen using the button or a relation to a remote specimen using the button. To show or hide the relations in the tree use the and buttons.

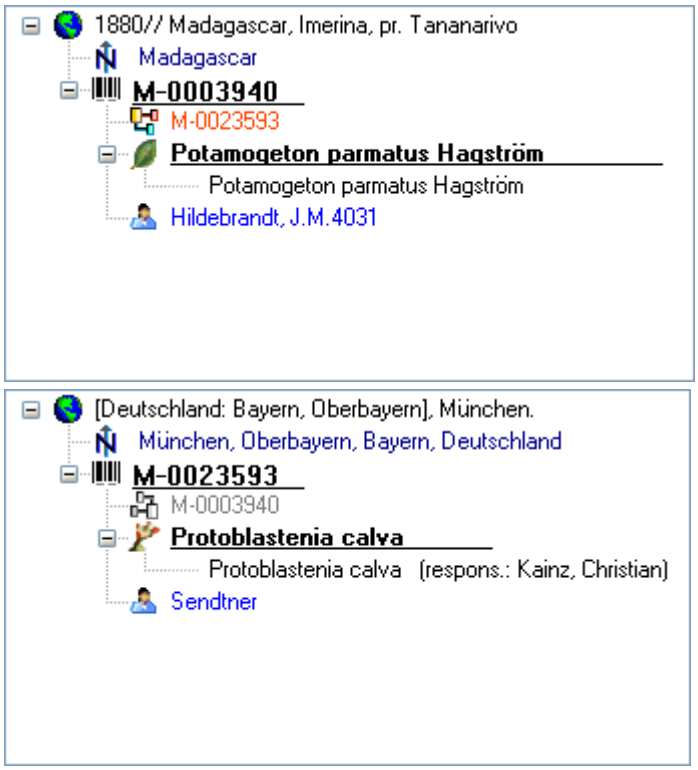


Interal relation

To add an internal relation to a local specimen use the button. A window will open where you can search for the related specimen.



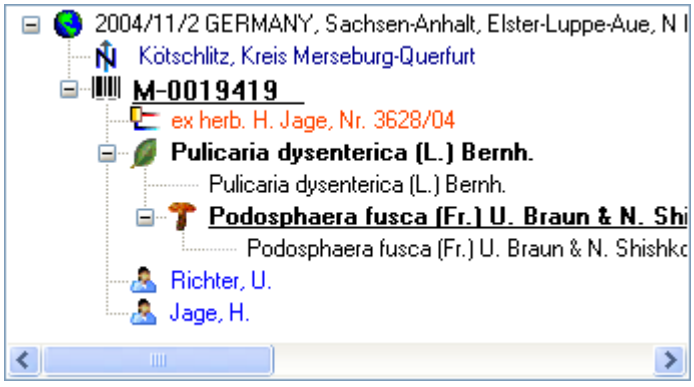
If you found the specimen click OK to close the form and establish the relation. The left image below you can see an internal relation, while the image on the right shows an invers relation from another specimen to the current specimen. In the tree an invers relation will then be displayed with an icon as shown below.



With internal relations you can navigate to the related specimens using the button.

External relation

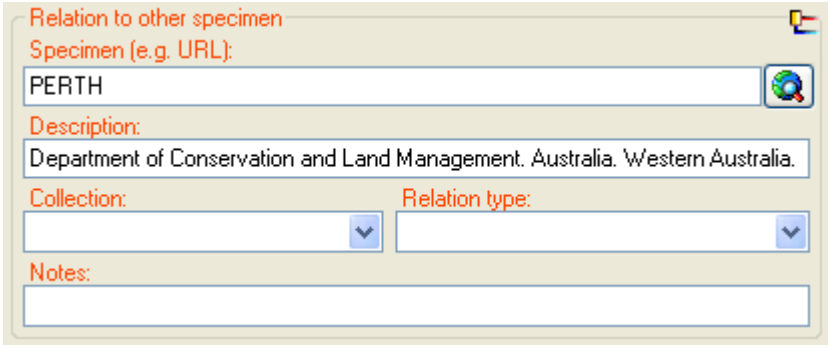
To add an external relation to a specimen that is not administrated in the database DiversityCollection use the button. The image below you can see an external relation in the tree.



To delete a relation select it from the list and click the button.

Editing the data

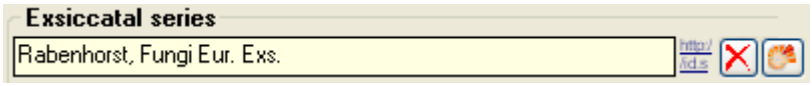
To edit the data for the relation, click on the entry in the tree to open the corresponding form as shown below for a relation to an external specimen.



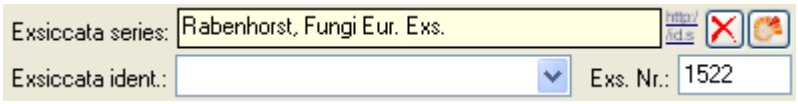
The data are stored in the table CollectionSpecimenRelation.

Exsiccatae

A specimen may be part of an exsiccatal series. These series are administrated in the Diversity Workbench module DiversityExsiccatae. To enter the exsiccatal series select the specimen or the organism resp. identification unit in the tree. With the specimen the exsiccatal series is shown at the base of the data form as shown below.



As there may be several organisms with separate exsiccatal numbers in one specimen the numbers are handled together with the identification units. Select the concerned organism to enter data in the form as shown below.



You can either type the name of the exsiccatal series or choose one from the module DiversityExsiccatae. If you click on the Link a form will open showing the link to the entry as derived from the module.

The data about the exsiccatal series are stored in the table CollectionSpecimen, the data about the number and the taxon are stored in the table IdentificationUnit.

Access to the data

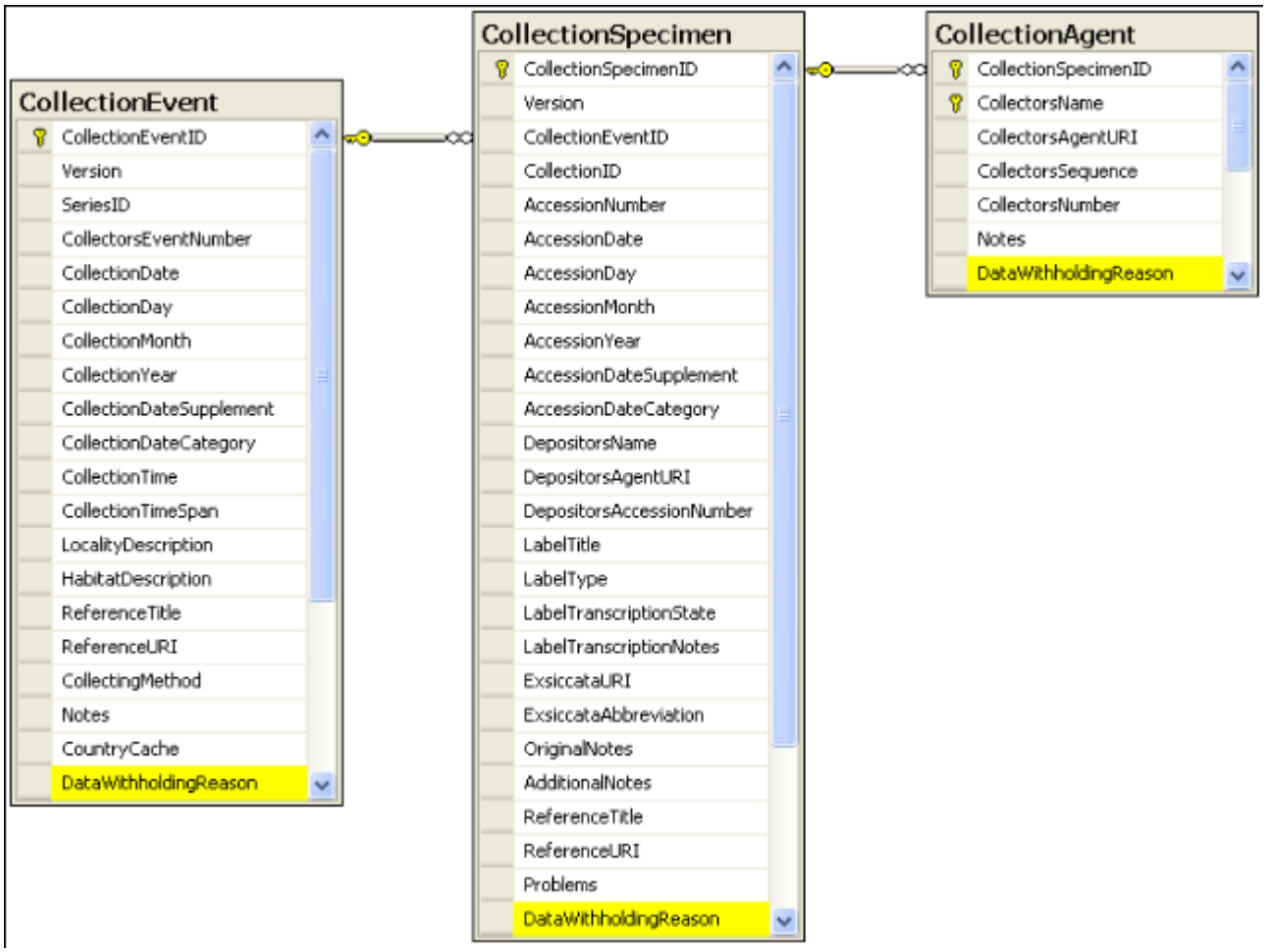
To get access to the data, you have to take several hurdles. In DiversityCollection, you must be a member of one of the user groups. You have only access to those data, that are listed in the projects you have access to. For external users, data may be blocked by entries in the data witholding reasons.

Availabiltiy of datasets

The data of certain parts of the database can be blocked for remote access. If you want to prevent access for the whole dataset, enter your reason in the field **Withholding reason** in the header (see below).

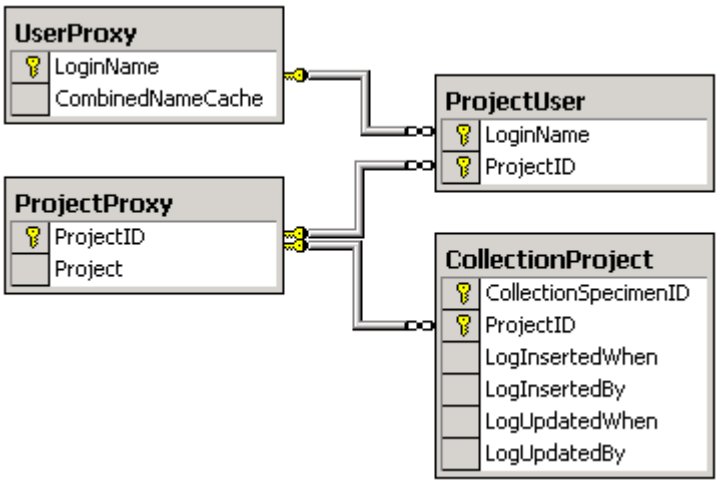
If you want to block only the access to the data of the collection event or the collector choose them in the tree and enter your reason for withholding the data in the corresponding fields. If these fields are empty, the data will be published.

Within the database a user has only access to the data of those projects for which he has the permission. If a dataset belongs to a certain project, this is stored in the table CollectionProject. To prevent a publication of the data these can be blocked for the whole dataset (table CollectionSpecimen), for the locality and collection site informations (table CollectionEvent) and for the collectors (table CollectionAgent) - see overview below.



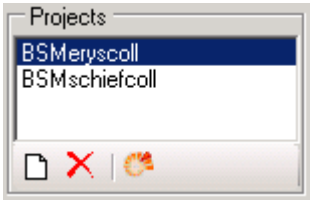
Permissions for projects

The access for the user to the data within the database are stored in the tables shown below. The tables UserProxy and ProjectProxy are related to the Diversity Workbench modules DiversityUsers and DiversityProjects respectively.

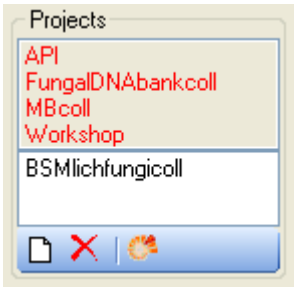

UserProxy
LoginName
CombinedNameCache
ProjectProxy
ProjectID
Project
ProjectUser
LoginName
ProjectID
CollectionProject
CollectionSpecimenID
ProjectID
LogInsertedWhen
LogInsertedBy
LogUpdatedWhen
LogUpdatedBy

Projects

Every collection specimen can be assigned to any number of projects. To assign a specimen to a project click on the button. To remove it from a project, select the project from the list and click the button.



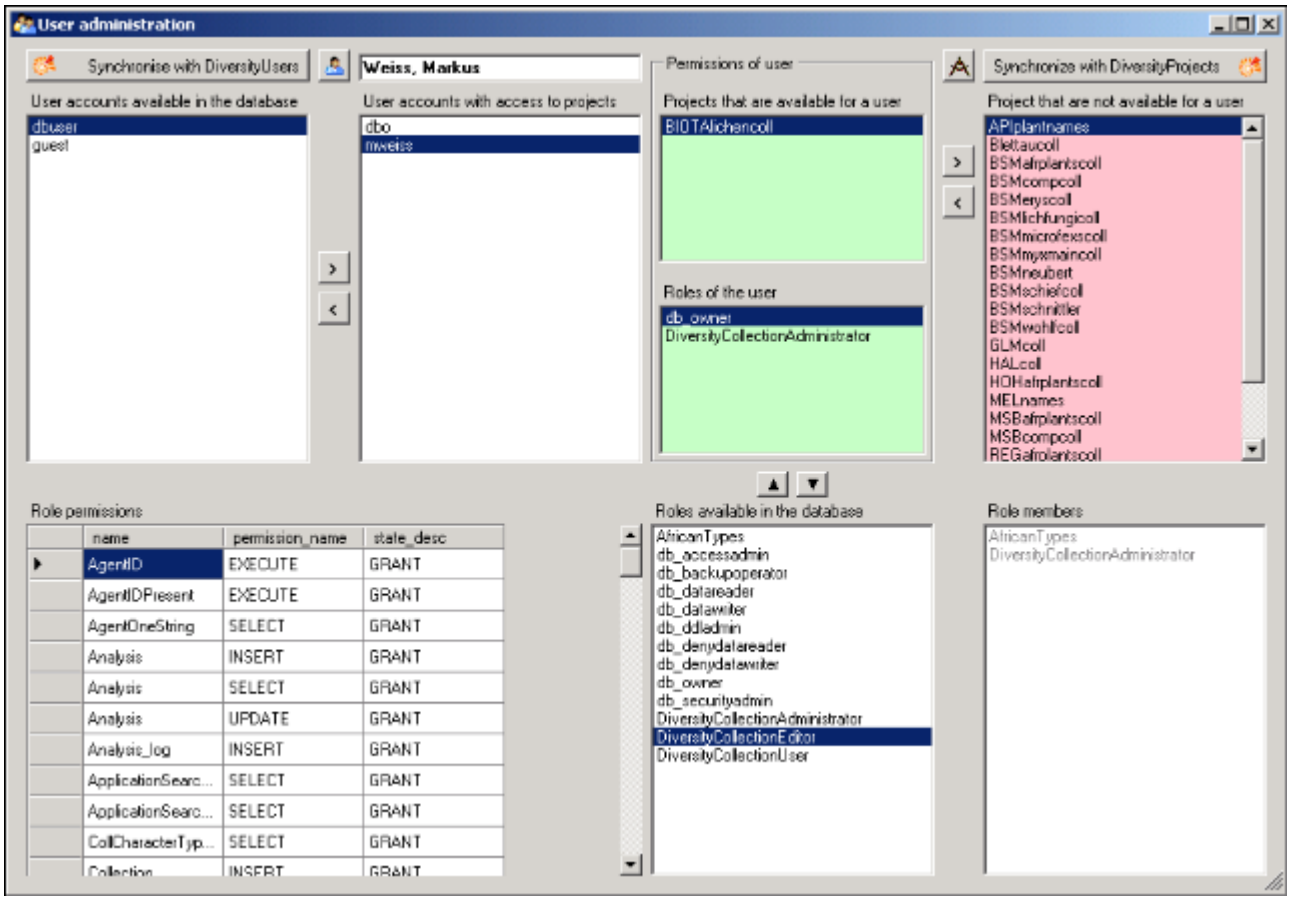
If there are projects, where you have no access to, these will be listed in a separate list at the top as shown below.



Data are stored in the table CollectionProject.

Details upon the projects within the DiversityWorkbench are stored in the database DiversityProjects. To open a project to see further information upon a project click on the button. To edit details in the projects you need the application **DiversityProjects.exe** in your application directory and access to the database DiversityProjects. To synchronize the projects listed in DiversityProject you can use the synchronize functionality in the user administration form as shown below. If DiversityProjects is not available you can create a new project with the button. If DiversityProjects is available, use the synchronize functionality

.


User administration
Synchronise with DiversityUsers
Weiss, Markus
Permissions of user
Synchronize with DiversityProjects
User accounts available in the database
dbuser
guest
User accounts with access to projects
dbo
mweiss
Projects that are available for a user
BIOTAlichencoll
Project that are not available for a user
APIplantnames
Blettaucoll
BSMafrplantscoll
BSMcompcoll
BSMeryscoll
BSMichfungicoll
BSMmicrofexscoll
BSMmyxmaincoll
BSMneubert
BSMschiefcoll
BSMschnittler
BSMwahlcoll
GLMcoll
HALcoll
HOHafrplantscoll
MELnames
MSBafrplantscoll
MSBcompcoll
REGafrplantscoll
Roles of the user
db_owner
DiversityCollectionAdministrator
Role permissions
name
permission_name
state_desc
AgentID
EXECUTE
GRANT
AgentIDPresent
EXECUTE
GRANT
AgentOneString
SELECT
GRANT
Analysis
INSERT
GRANT
Analysis
SELECT
GRANT
Analysis
UPDATE
GRANT
Analysis_log
INSERT
GRANT
ApplicationSearc...
SELECT
GRANT
ApplicationSearc...
SELECT
GRANT
CollCharacterTyp...
SELECT
GRANT
Collection
INSERT
GRANT
Roles available in the database
AfricanTypes
db_accessadmin
db_backupoperator
db_datareader
db_datawriter
db_ddladmin
db_denydatareader
db_denydatawriter
db_owner
db_securityadmin
DiversityCollectionAdministrator
DiversityCollectionEditor
DiversityCollectionUser
Role members
AfricanTypes
DiversityCollectionAdministrator

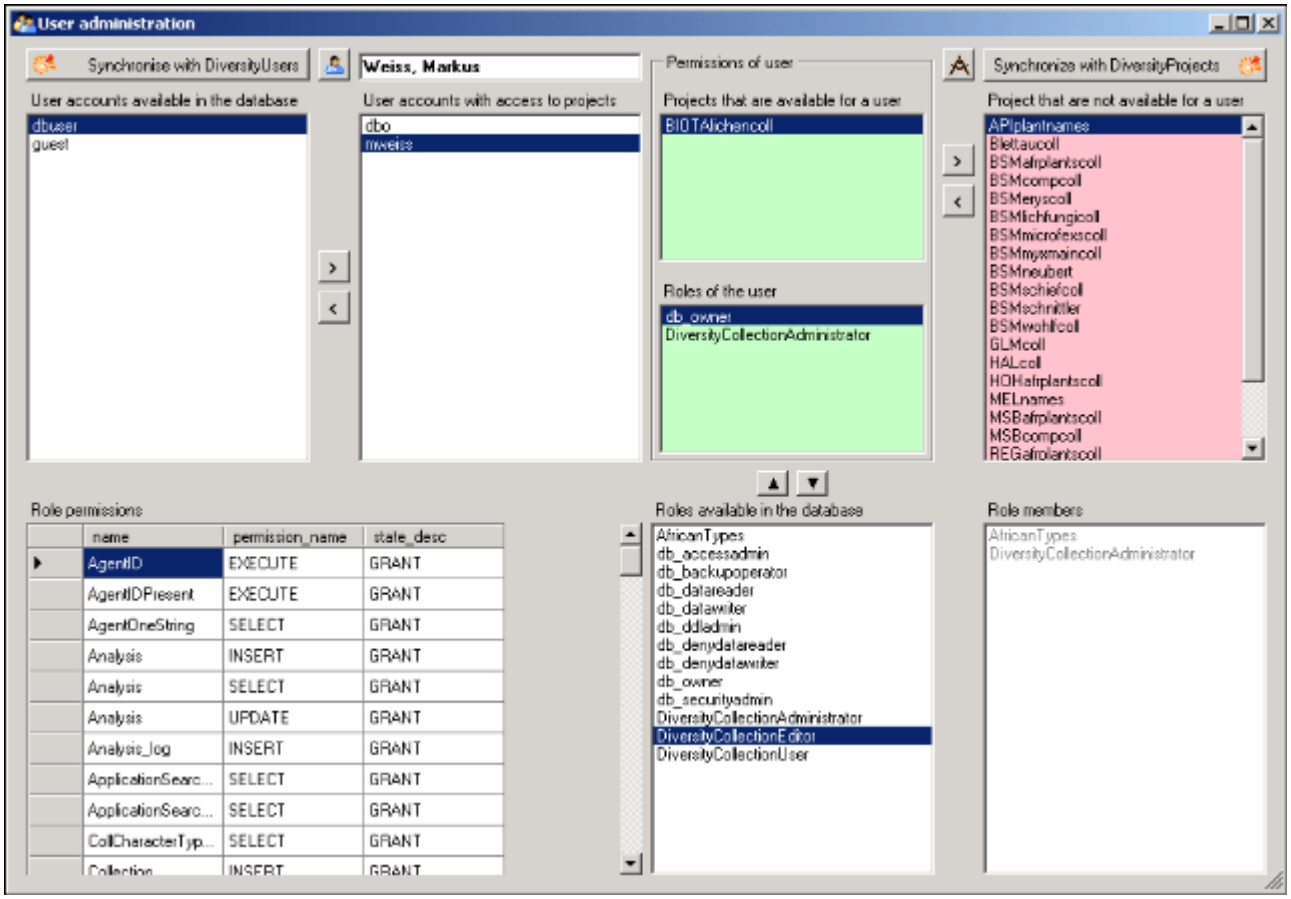
User administration

The permissions of users in the database are set via roles and the access to the projects. To set the permissions choose **Administration** - **User** ... from the menu. A form will open where you can change the permissions of the user. The first list shows the user accounts that are available in the database but have no access to any of the projects. To synchronize this list with the Diversity Workbench module DiversityUsers click on the

button. To create a new SQL-Server User, click on the button. To create a new Windows user with access to the database resp. allow an existing user the access to the database use the Microsoft SQL Server Management Studio (see the installation section for further details). To permit access to a project click on the button. To remove a user from this list use the button.

The area **Permission of user** shows the projects a user has access to and the roles of the user within the database. To move users between the lists Projects that are available for a user and Projects that are not avialable for a user use the and buttons. To create a new project click on the button. If you use the Diversity Workbench module DiversityProjects, you can create a new project there and user the synchronize button

. To change the roles of a user use the and buttons. Underneath the project list for a user you find the list of the Roles of the user. This list can be changed by using the and buttons. Underneath the roles list you find the list with the roles available in the database. On the left side of this list the permissions of the selected role are listed and on the right side the members of this role.



The permissions of the roles are shown in the lower left part of the form.

The role **DiversityCollectionUser** can see the data within the permitted projects, but can not change anything.

The role **DiversityCollectionTypist** can edit a part of the user defined data.

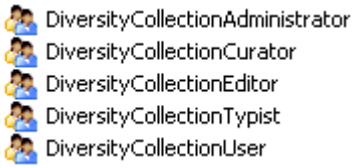
The role **DiversityCollectionEditor** can change the user defined parts of the data.

The role **DiversityCollectionCurator** has the same rights as the group DiversityCollectionEditor and in addition can handle transactions, i.e. shipments of specimen between collections.

The role **DiversityCollectionAdministrator** can delete data, edit the contents of internal tables, change user permissions etc.

Security

A user can be in 5 groups with diverse rights in the database where the higher groups have all rights of lower groups in addition to special rights for this group, e.g. DiversityCollectionUser can only read the data of certain tables while DiversityCollectionTypist has the rights of DiversityCollectionUser and additionally can edit the data in certain tables - see overview below.



Summarzied overview of the permissions of the groups

Group	**Permissions in addition to lower group**
Administrator	Delete data, edit user permissions
Curator	Create and edit transactions (e.g. loans)
Editor	Create new entries and delete details (not whole datasets)
Typist	Edit data
User	See the data of the data tables

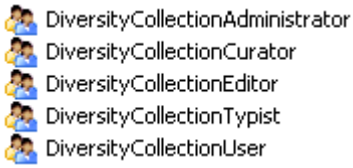
If you are an Administrator you can add a user to one of these groups

Any user may have access to several projects.

Transactions

Prerequisites

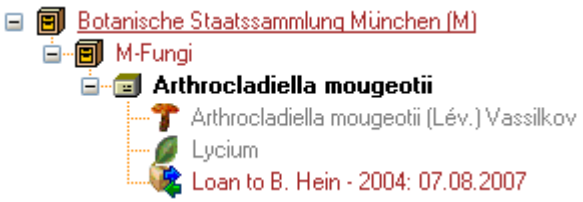
The transfer of specimen between collections is handled with transactions. There are two prerequisites to use transactions in DiversityCollection. You must be in the group DiversityCollectionCurator (see user administration).

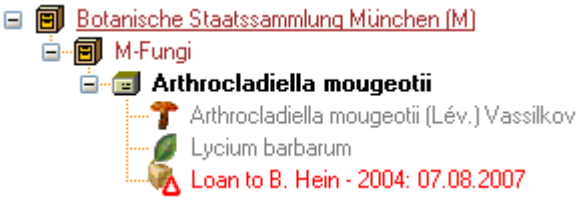


and the collections you have the right to handle transactions for must have been assigned to you as shown in the curators section.

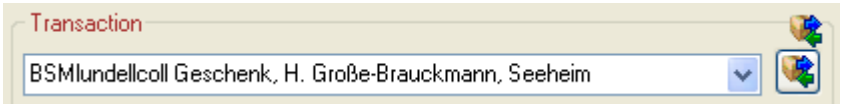
Visibility of transactions

A specimen may be involved in diverse transactions between collections like loan, purchase, gift or exchange. The transactions are shown in the tree for the parts of a specimen and symbolized with an icon or for specimens that are still on loan an icon as shown below.





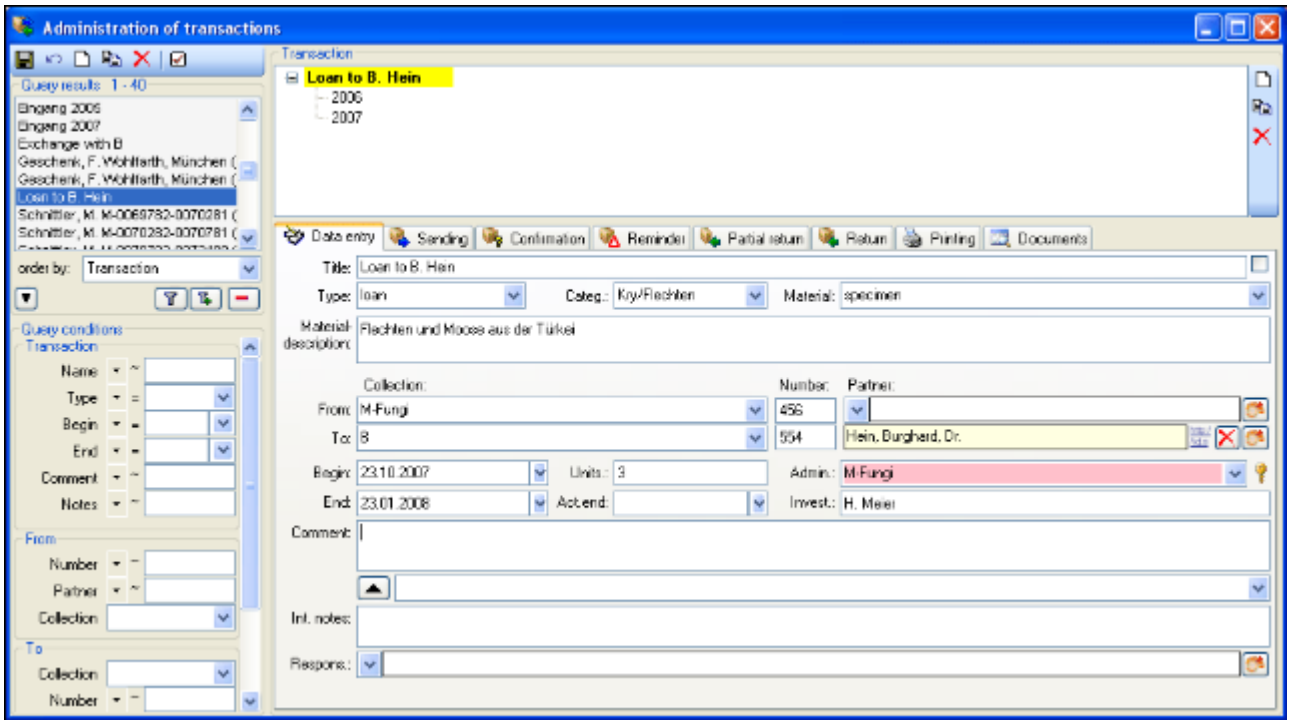
If you are a curator you can take a look at the details of a transaction. Select the specimen part in the tree and then click on the button that will be shown in the details next to the name of the transaction. This will open the form for the transactions as described below.



To edit the transactions choose **Administration - Transactions** from the menu. A form will open as shown below. In the tree a the top the relations between the transactions are shown. To create a transaction, dependent on another transaction, choose the superior transaction in the tree and click on the button in the panel right from the tree. With the button you can create a copy of a transaction. To delete a specimen from the list click on the button. The data for the transactions are stored in the tables Transaction and CollectionSpecimenTransaction .

Data entry

On this page you can edit the data of a transaction. If a transaction is linked to an administrating collection (symbolized by the key), it can be edited exclusively by curators of this collection. To appoint currators choose **Administration - Curators** from the menu.

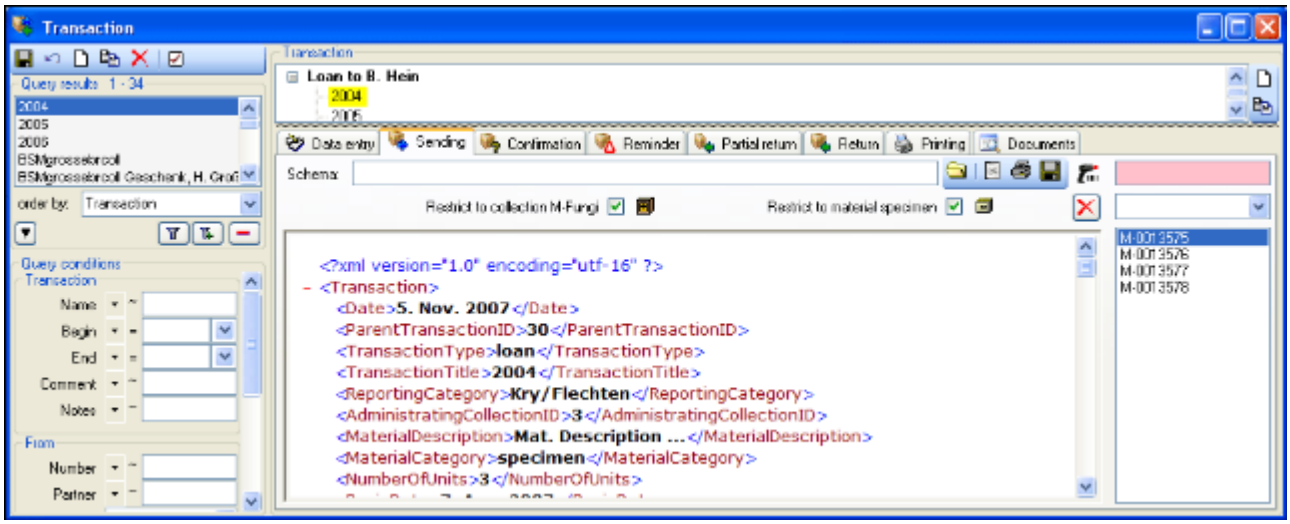


Depending on the type of the transaction additional tab pages will present for creating documents.

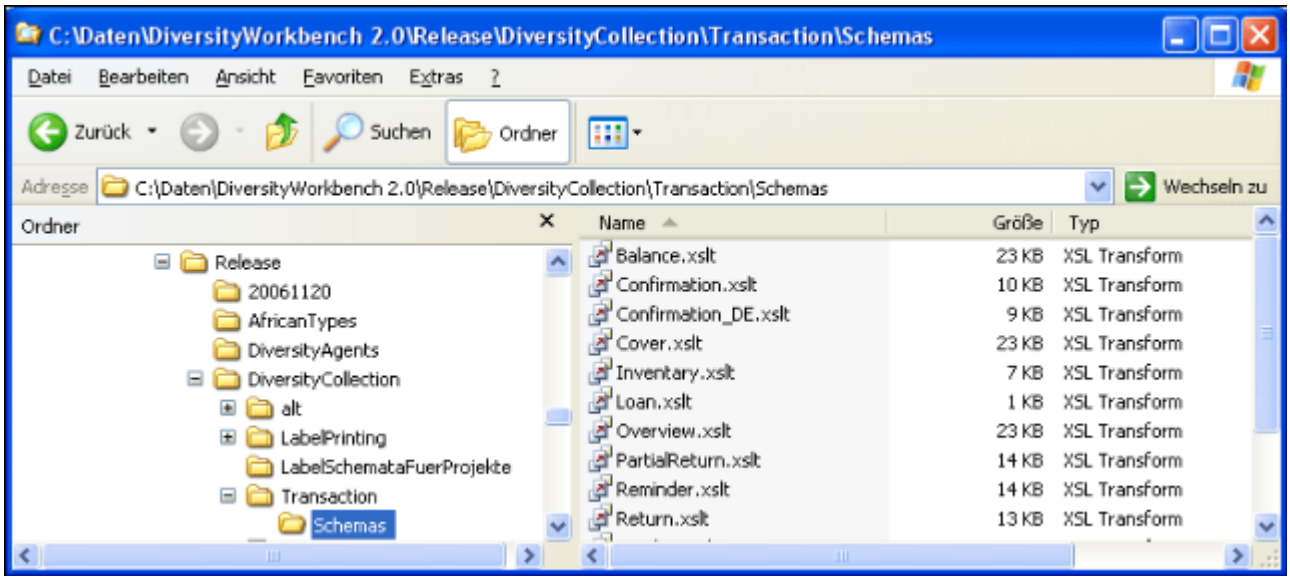
Transaction type	***Tab pages***					
Sending	Loan	Borrow	Exchange	Gift	Purchase	
Confirmation	Loan	Borrow	Exchange	Gift	Purchase	Request
Reminder	Loan	Borrow	Exchange	Gift	Purchase	
Partial return	Loan	Borrow				
Return	Loan	Borrow				
Printing	Loan	Borrow	Exchange	Gift	Purchase	Request
Documents	Loan	Borrow	Exchange	Gift	Purchase	Request
Balance			Exchange			

Common notes

On the tab pages for generating documents you have to specify a xml-schema for your document - click on the button to choose from the predifined schemas or create your own schema. If you omit to specify a schema for your form you will see the XML output created by the database as shown below.



To choose a schema click on the button and select a schema from the list. DiversityCollection contains several ready to use schemas in the folder **Transaction - Schemas**.

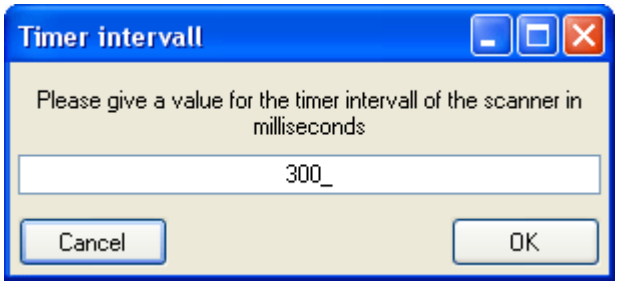


Feel free to change these schemas to your own needs. Choose the schema you need and then click on the button to create a document. To print the document, use the button. To store the current document in the documents for later reference click on the button.

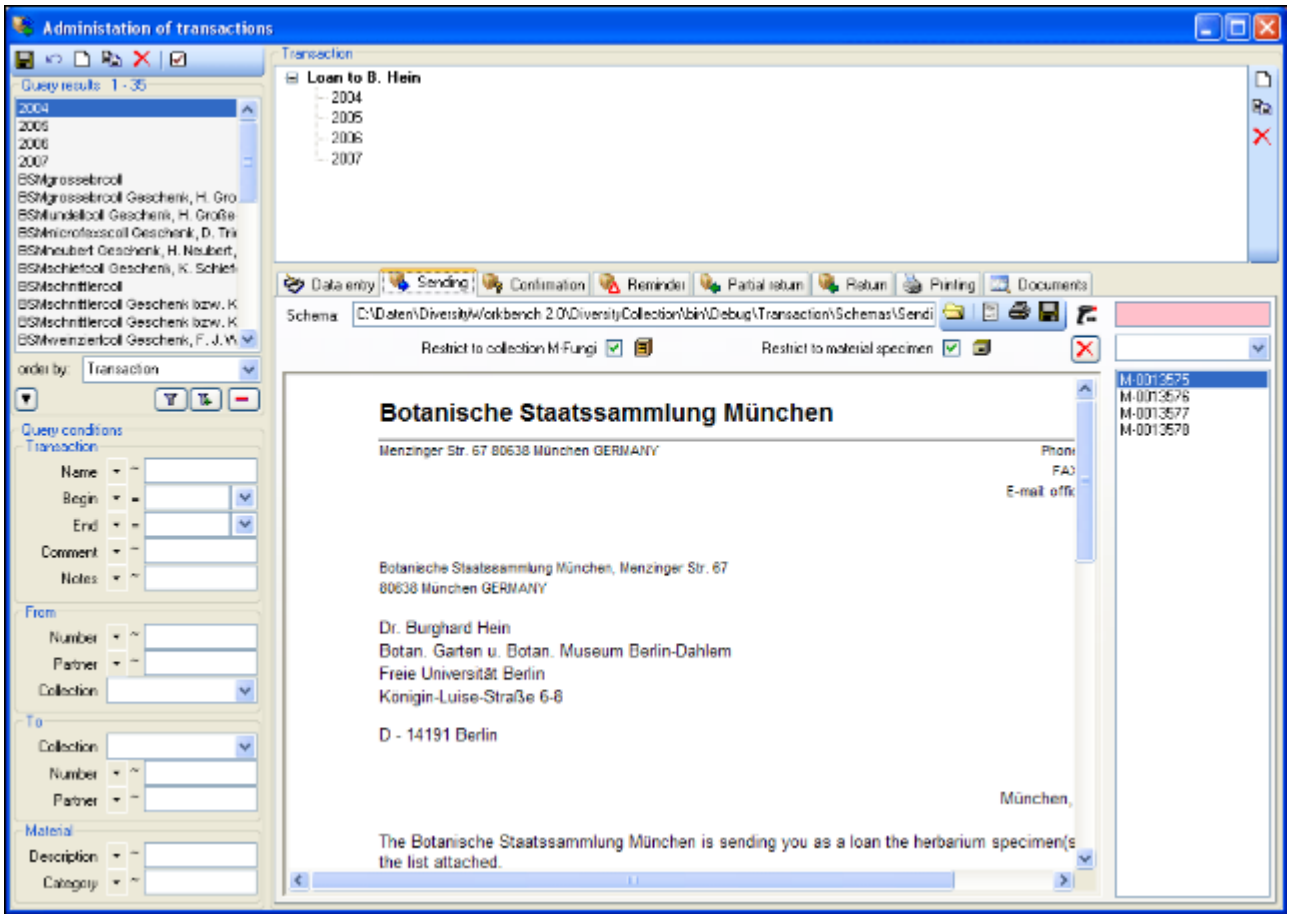
Transactions - Sending

In the sending tab page of the transaction form you can create the covering note for a sent sample. To choose a schema click on the button and select a schema from the list. DiversityCollection contains several ready to use schemas in the folder **Transaction - Schemas**.

To add a specimen to the list, you can either use the combobox or a barcode scanner. In case you scanner is reading only parts of the barcode try to adjust the timer intervall - click on the button to open a form as shown below.



To restrict the selection of the specimens and parts of the specimens, you can check the **Restrict to collection** ... and **Restrict to material** ... options. To remove a specimen from the list, use the button. Click on the button to create the document as shown below.



Transactions - Confirmation

On this page you can create an inquiry letter for a package sent, e.g. if a parcel was sent to a loan taker and no confirmation that the parcel reached its destination was returned so far.

With the button, choose the schema you need. Click on the button to create a document. To print the document, use the button. To store the current document in the documents for later reference click on the button.

Transactions - Reminder

If a loan is due to return resp. a loan taker did not meet the deadline, you can create a prompt note here.

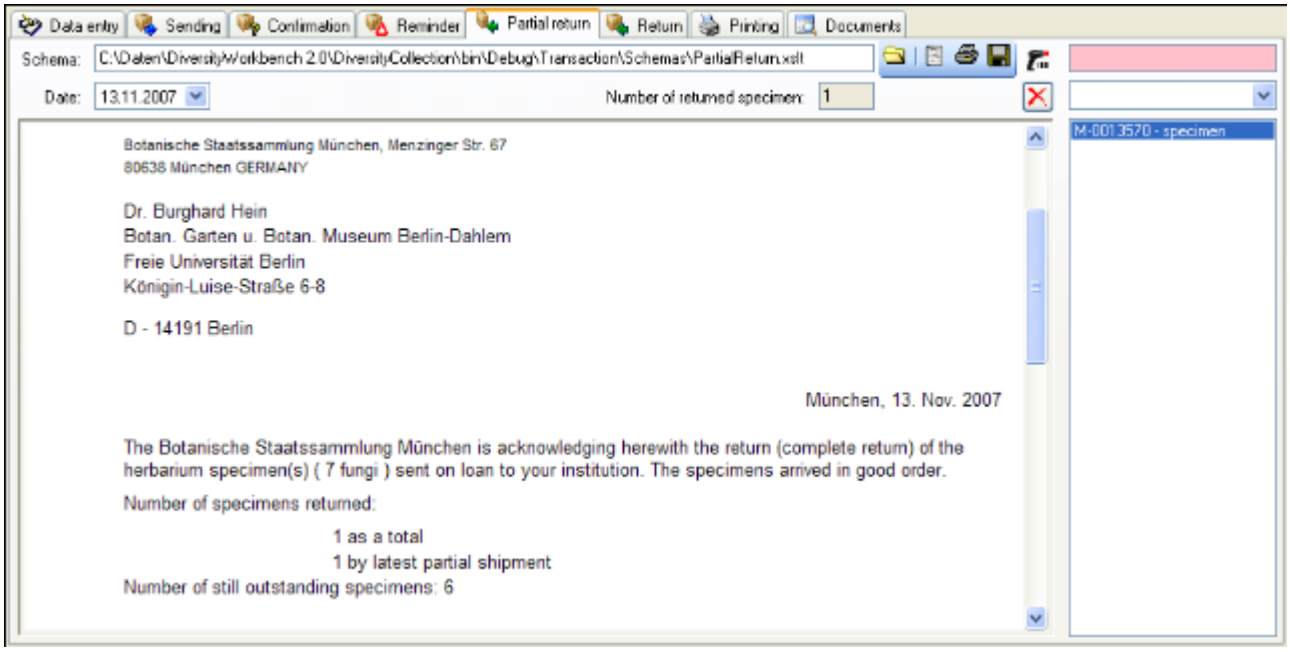
With the button, choose the schema you need. Click on the button to create a document. To print the document, use the button. To store the current document in the documents for later reference click on the button.

The at the upper right the missing specimen and at the lower right the returned specimen are listed.

Transactions - Partial Return

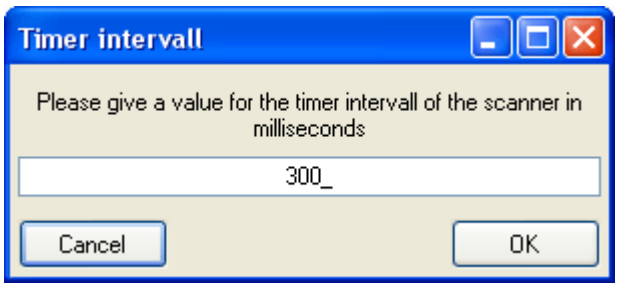
If parts of a loan were returned, create a letter of acknowledgment here.

With the button, choose the schema you need. Click on the button to create a document. To print the document, use the button. To store the current document in the documents for later reference click on the button.



You have two option to enter returned specimen.

You may either use a barcode scanner - move the mouse cursor into the field for the barcode detection, then scan the barcode. The specimen will be automatically inserted into the list. In case your scanner is reading only parts of the barcode try to ajust the timer intervall - click on the button to open a form as shown below. Here you can set the interval to a value that is compatible to your scanner.

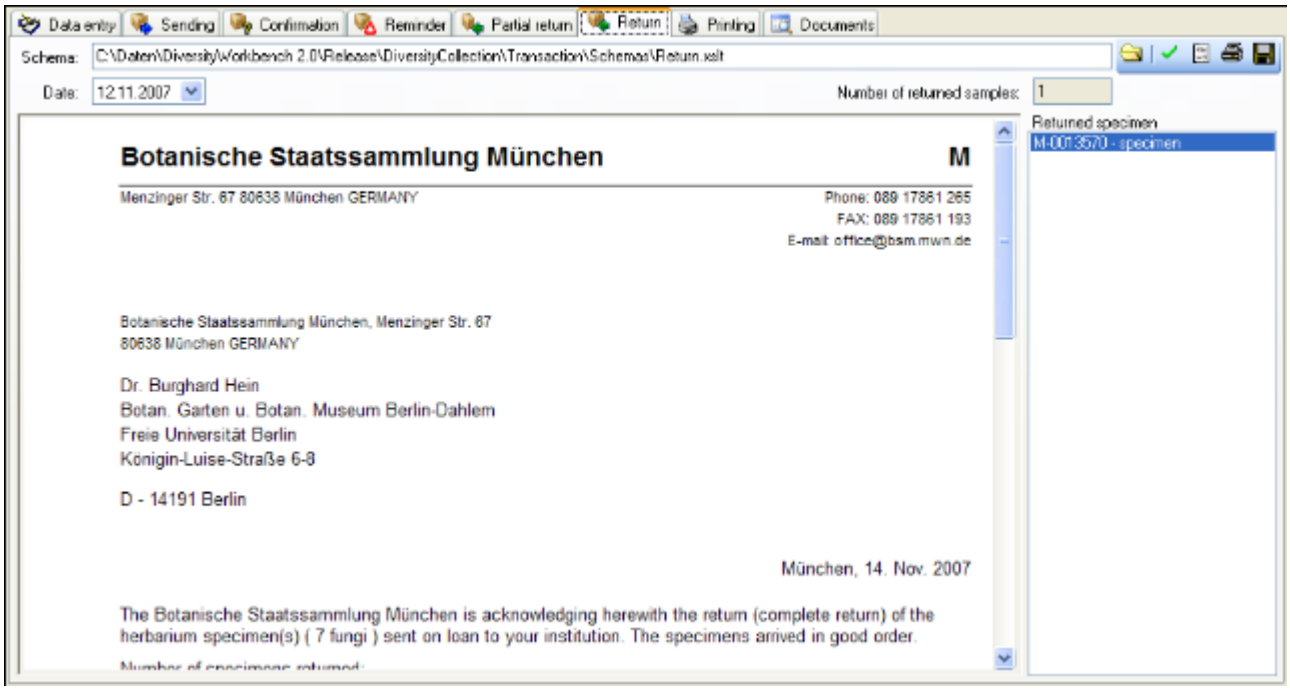


If you lack a scanner or you do not use barcodes, use the drop down box to select the specimen from the list.

To remove a specimen from the list use the button.

Transactions - Return

If a loan is returned, create a letter of acknowledgment here. With the button, choose the schema you need. If all specimen were returned, click the button. This take all remaining specimen into the list of the returned specimens. With the date field you can define a different date as the start if e.g. you starting to register the returned specimen some days before. Click on the button to create a document. To print the document, use the button. To store the current document in the documents for later reference click on the button.

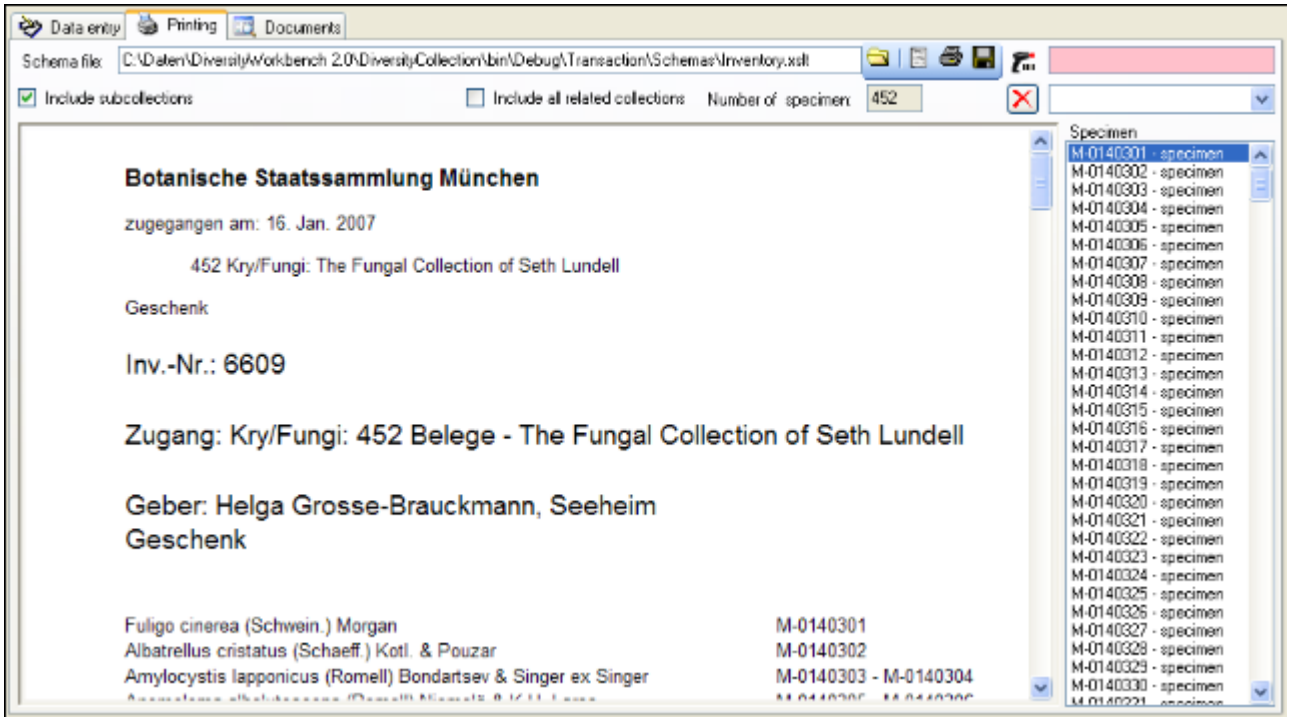


Transactions - Printing / Inventory

Besides the special pages for creating letters along with a loan this page provides the possibility to print cover letters, inventories and the like. You can restrict the selected specimen to the current collection, include the subcollections or all related collections - just activate the corresponding checkboxes.

To add a specimen to the list, you can either use the combobox or a barcode scanner. To use the barcode scanner, move your mouse cursor into the pink field right from the scanner. In case your scanner is reading only parts of the barcode try to adjust the timer intervall - click on the button to open a form where you can set the timer intervall to a different value. To remove specimen form the list, select it and then click on the button.

With the button, choose the schema you need. Click on the button to create a preview of the document. To print the document, use the button. To store the current document in the documents for later reference click on the button.



Transactions - Documents

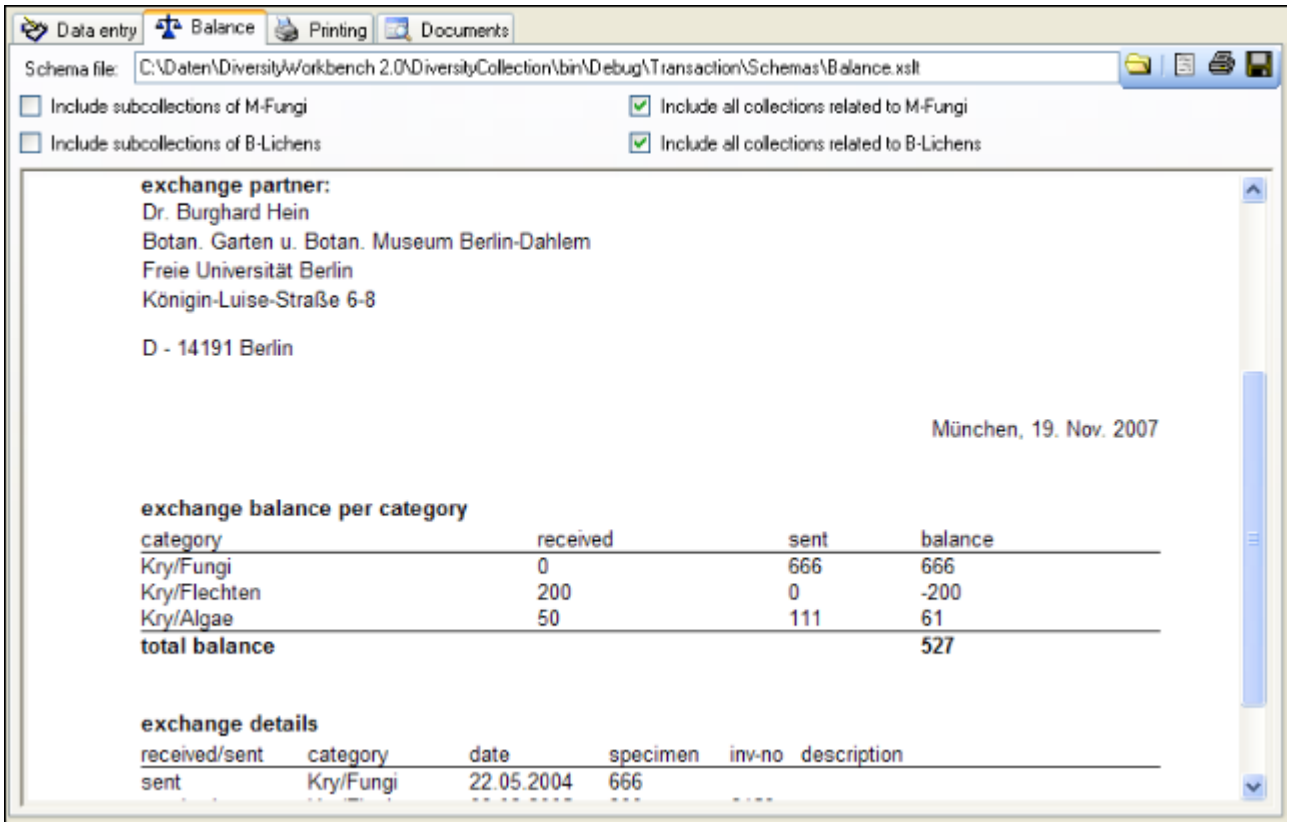
This page stores all the documents created or received along with a transaction.

To add a document, create a screeshot of this document, then create a new entry (click on the button) and insert the screenshot with the Add image from document button. In the lower part you can enter any text related to the document.

Transactions - Balance

The balance for the exchange between two collections.

With the button, choose the schema you need. Click on the button to create a document. To print the document, use the button. To store the current document in the documents for later reference click on the button. To include either the subcollections of any related collections to the collections of the current transaction select the corresponding checkboxes.

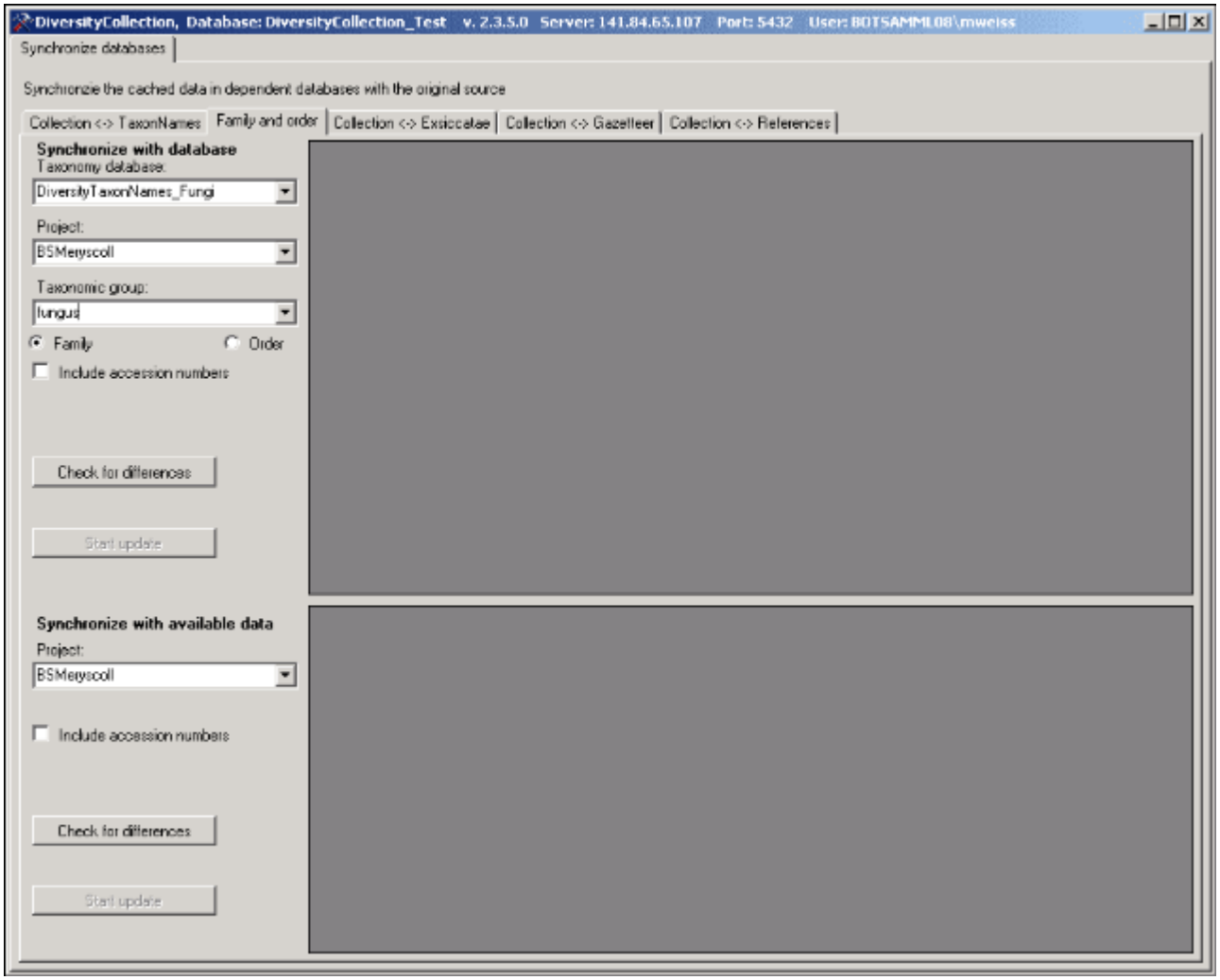


Maintenance

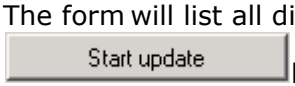
DiversityCollection can be linked several other modules within the Diversity Workbench. If you link a datasource to another module, the URI of the dataset in the other module together with several cached values will be stored in DiversityCollection. Along with changes in the other modules these cached values may differ from the original values. To get the actual values you can set these with the maintenance function within DiversityCollection.

toChoose Administration -> from the menu. A form as shown below will open. To transfer the higher taxonomic entries choose the tab Family and order. The upper part will synchronize the entries with taxonomic databases like DiversityTaxonNames_Fungi. Select a taxonomic database, a project and a taxonomic group, then choose if you want to synchronize the Family or th order. If you want to take a look at the single datasets you have to check the "Include accession number" checkbox. Than start the query with a click on the

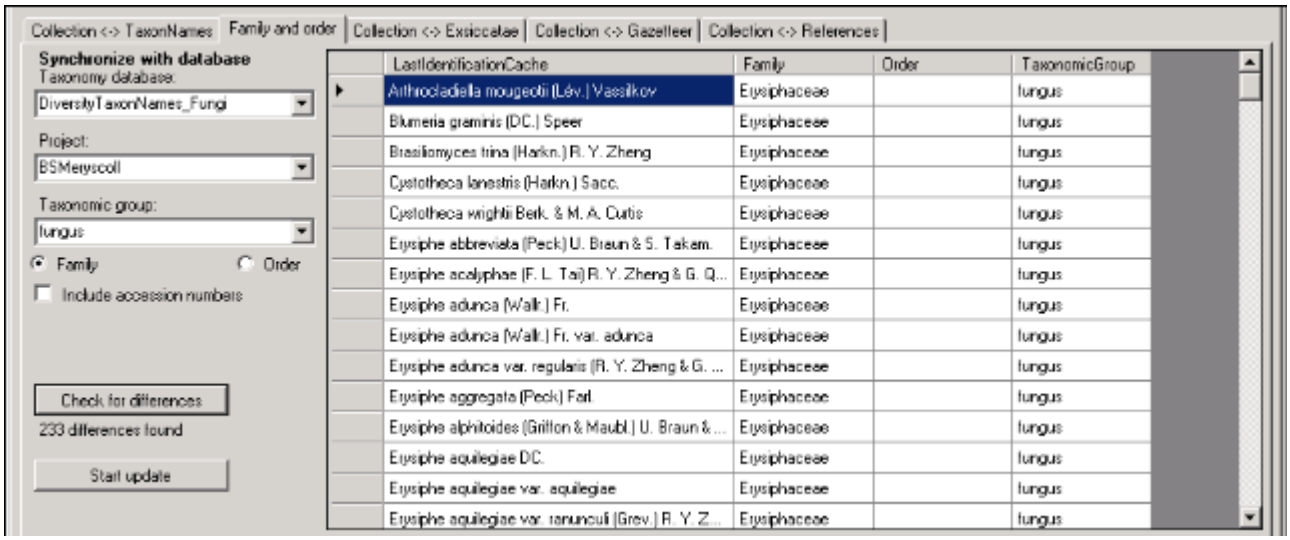
button.



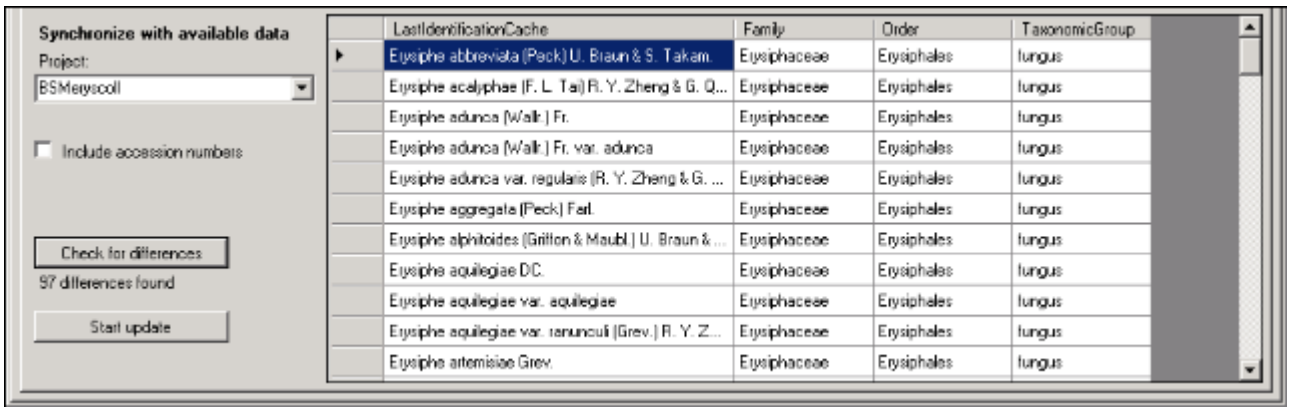
The form will list all differences found as shown below. To update the database click on the



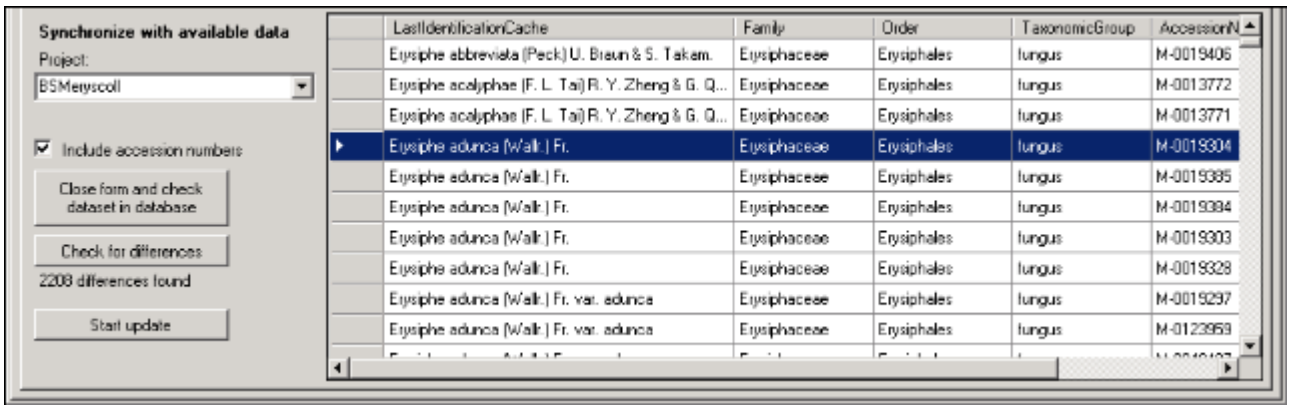
button.



In the lower part you can synchronize your entries within the database. Choose a project and click on the Check for differences button to start the query. To transfer the higher taxa to the found dataset click on the Start update button.



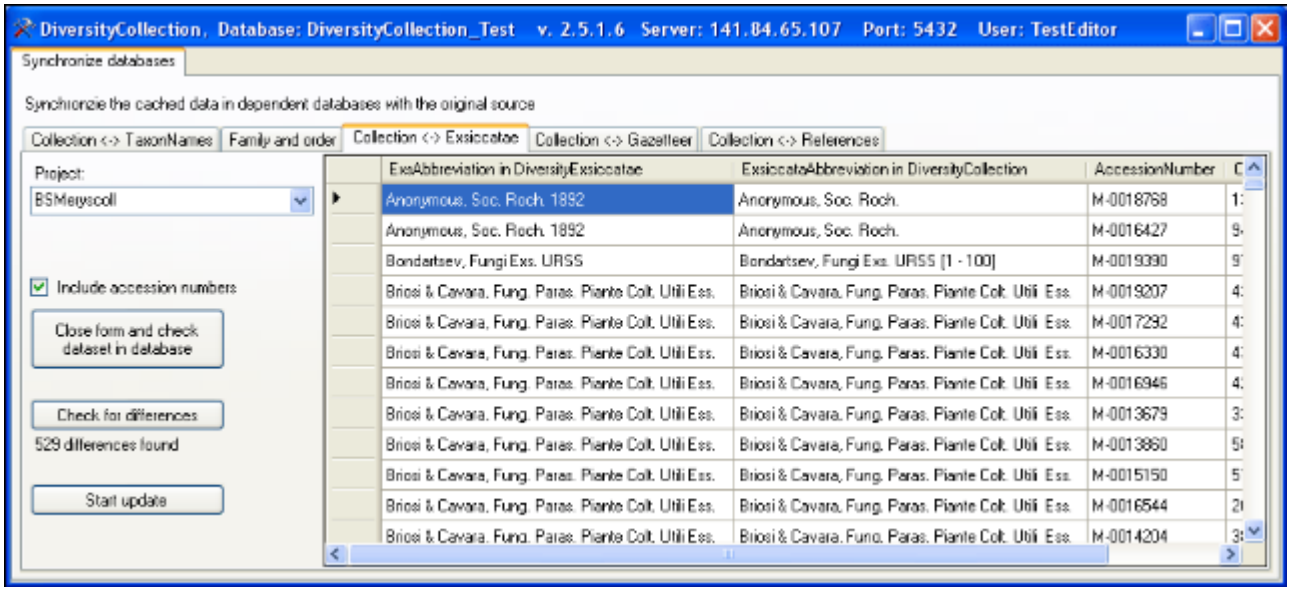
If you want to have a more detailed look on you data you have to check the "Include accession number" checkbox before starting the Query. A button Close form and check dataset in database will appear that will take you back to a single dataset in the database.



Maintenance - Exsiccatae

To synchronize the abbreviations titles of the titles of the exsiccatal series that are linked to the module DiversityExsiccatae choose **Administration -> Maintenance** from the menu. A form as shown below will open. On the tab page **Collection <-> Exsiccatae** select the project for which the titles should be synchronized. The title is stored in the field ExsiccataAbbreviation in the table CollectionSpecimen. Select the project that should be synchronized. Than start the query with a click on the

button.



The form will list all differences found. To update the database click on the

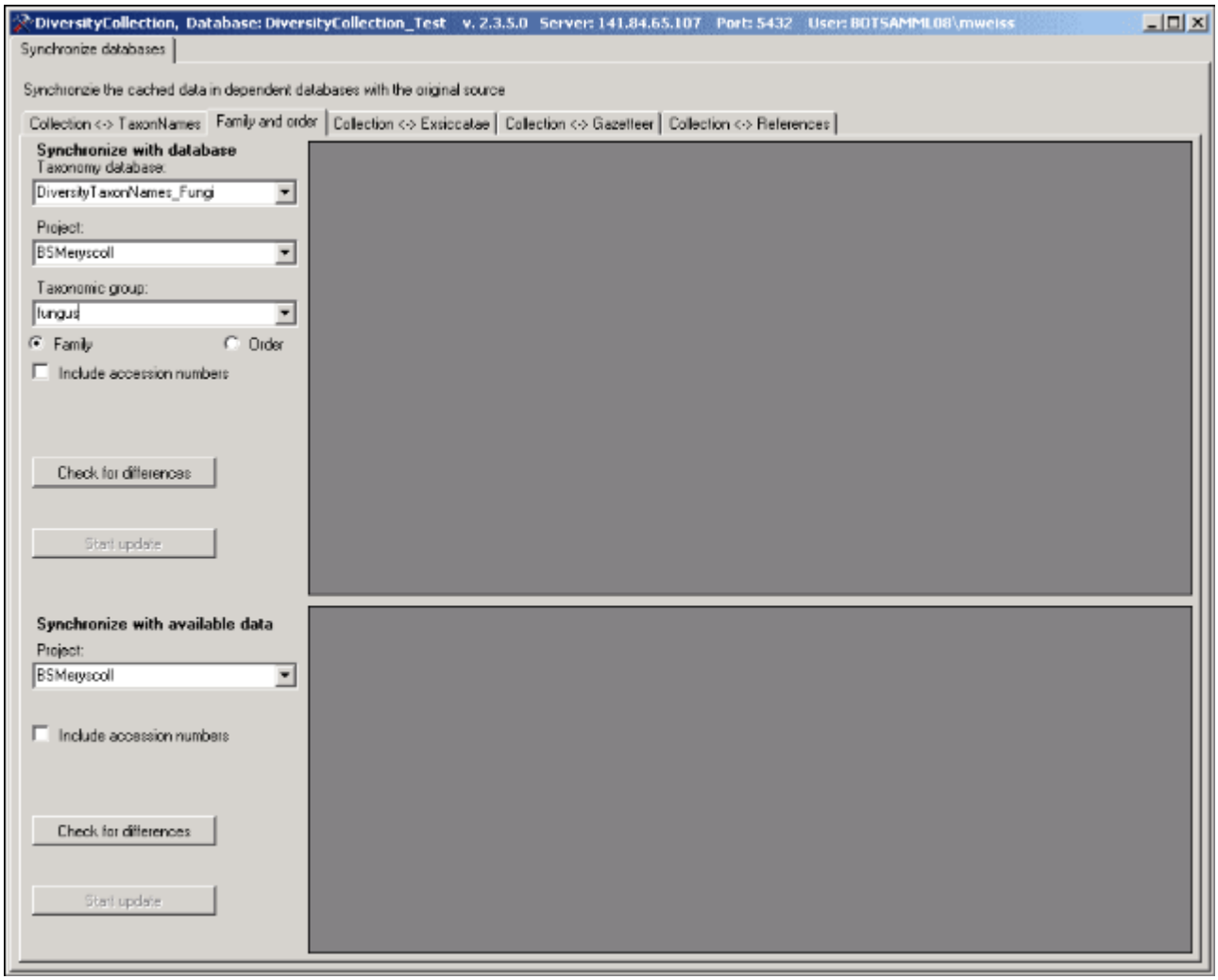
button. If you want to have a more detailed look on you data you have to check the "Include accession number" checkbox before starting the Query. A button

will appear that will take you back to a single dataset in the database.

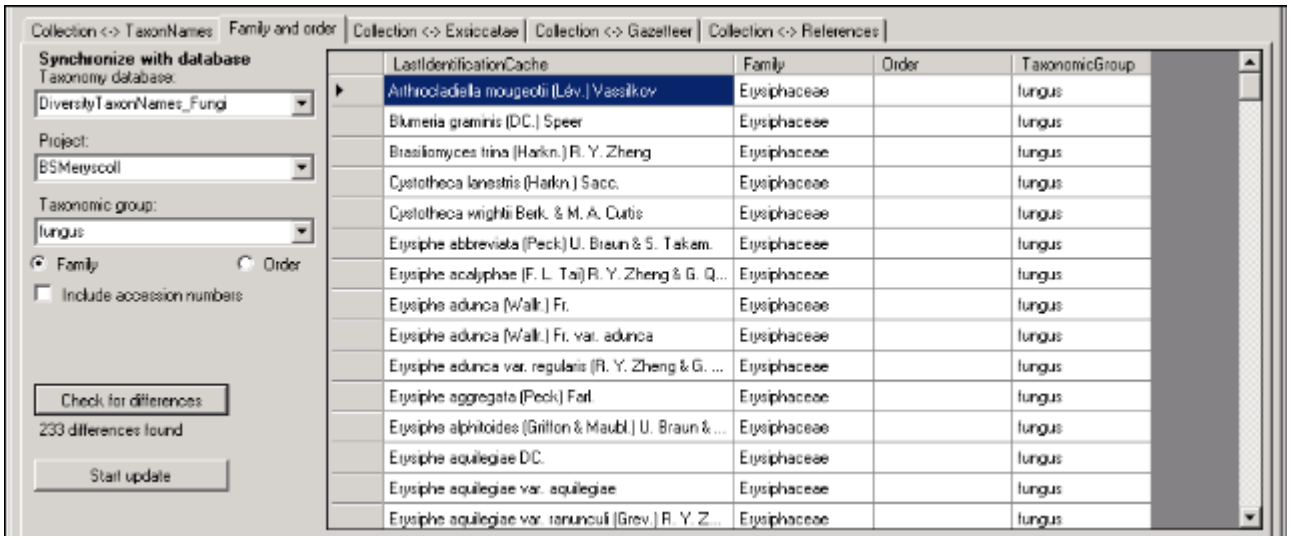
Maintenance - Family and Order

To synchronize the entries for the family and order of taxa derived from the module DiversityTaxonNames choose **Administration -> Maintenance** from the menu. A form as shown below will open. To synchronize the higher taxonomic entries for entries linked to the module DiversityTaxonNames choose the tab **Family and order**. These are stored in the fields FamilyCache and OrderCache in the table IdentificationUnit .

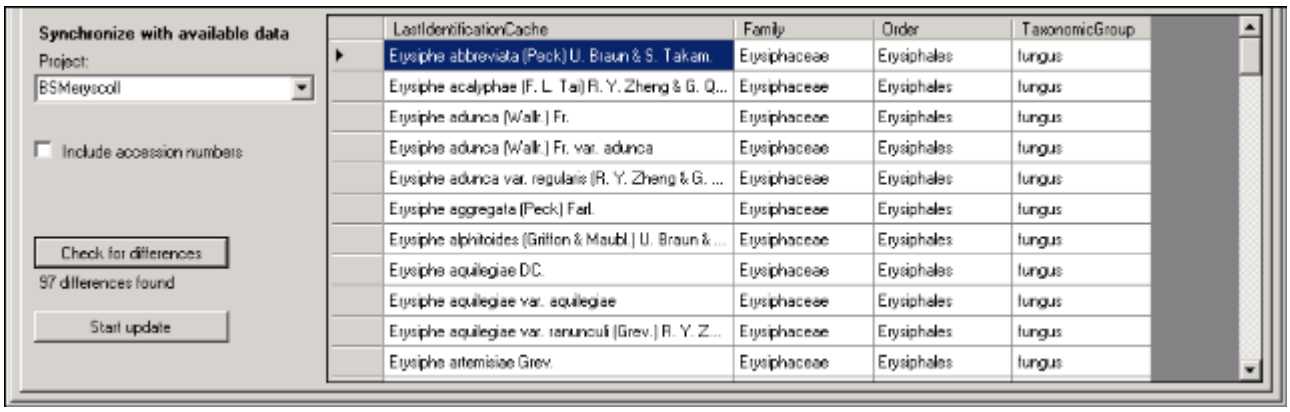
In the upper part will synchronize the entries with taxonomic databases like DiversityTaxonNames_Fungi. Select a taxonomic database, a project and a taxonomic group, then choose if you want to synchronize the Family or the order. If you want to take a look at the single datasets you have to check the "Include accession number" checkbox. Than start the query with a click on the Check for differences button.



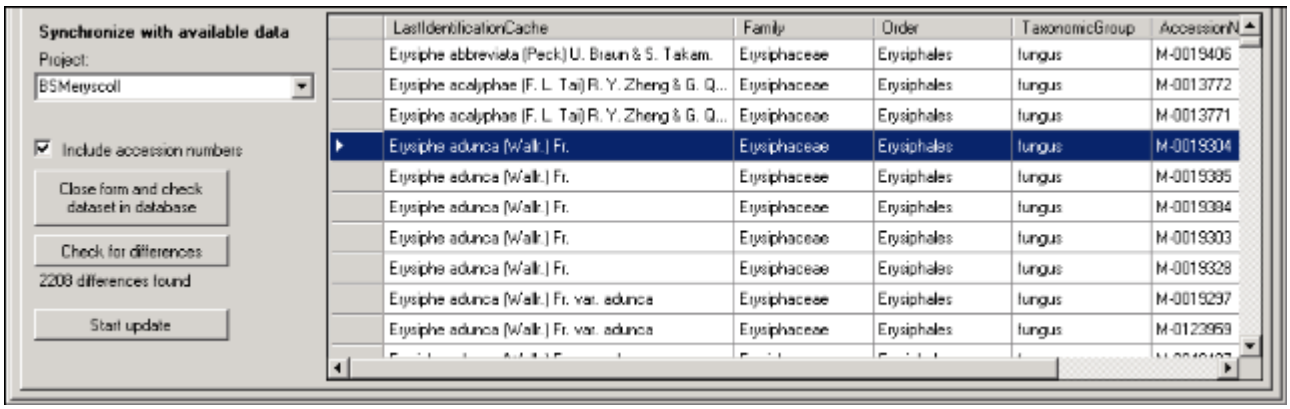
The form will list all differences found as shown below. To update the database click on the Start update button.



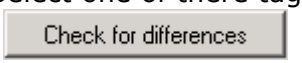
In the lower part you can synchronize your entries within the database. Choose a project and click on the Check for differences button to start the query. To transfer the higher taxa to the found dataset click on the Start update button.



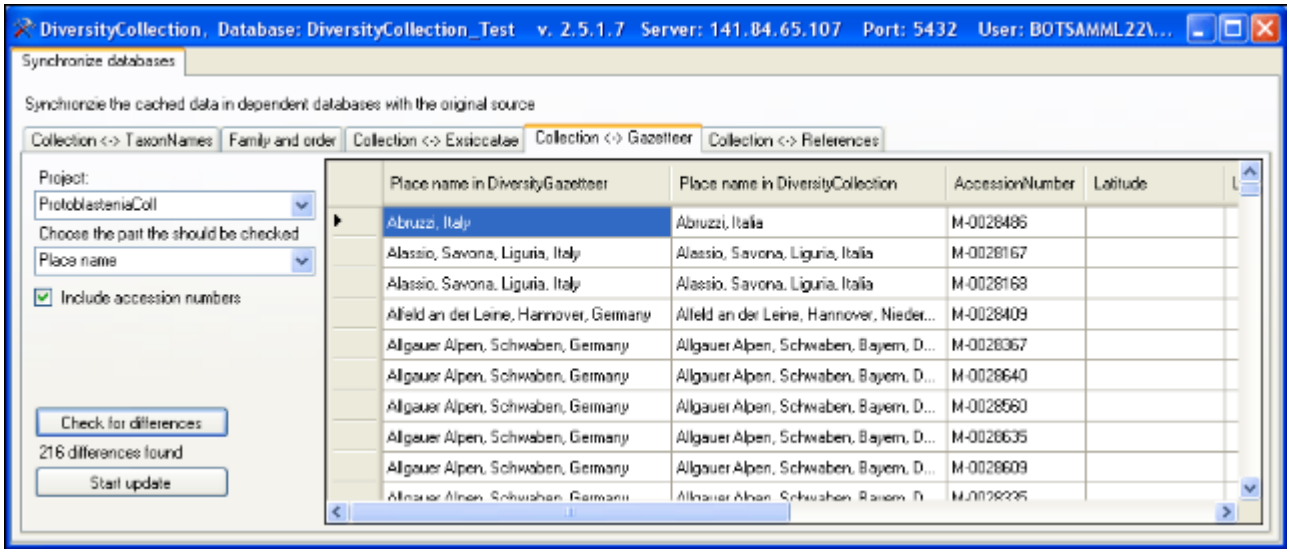
If you want to have a more detailed look on you data you have to check the "Include accession number" checkbox before starting the Query. A button Close form and check dataset in database will appear that will take you back to a single dataset in the database.



Maintenance - Gazetteer

To synchronize the entries derived from the module DiversityGazetteer choose **Administration -> Maintenance** from the menu. A form as shown below will open. On the tab page **Collection <-> Gazetteer** select the project for which the entries should be synchronized. There are 3 targets for the synchonization: Place name in the field Location1 in table CollectionEventLocalisation, Country stored in the field CountryCache in the table CollectionEvent and the Coordinates stored in the fields AverageLatitudeCache and AverageLongitudeCache in the table CollectionEventLocalisation. Select one of there tagets for the synchronization. Than start the query with a click on the 

button.



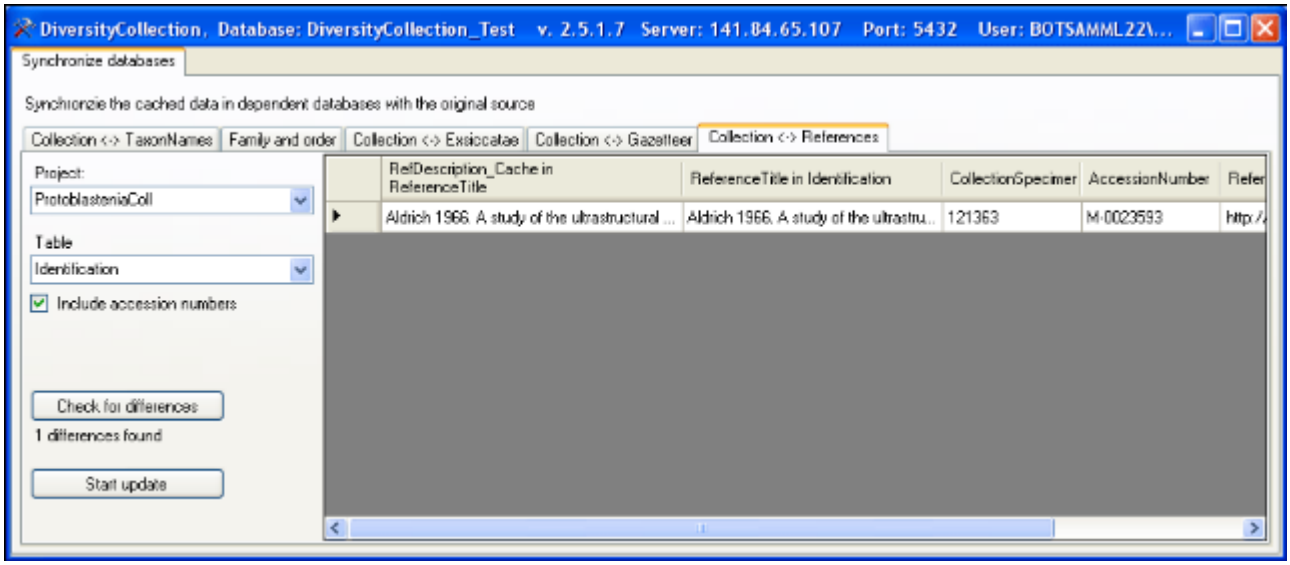
The form will list all differences found. To update the database click on the

button. If you want to have a more detailed look on you data you have to check the "Include accession number" checkbox before starting the Query. A button

will appear that will take you back to a single dataset in the database.

Maintenance - References

To synchronize the titles of the references that are linked to the module DiversityReferences choose **Administration -> Maintenance** from the menu. A form as shown below will open. On the tab page **Collection <-> References** select the project for which the reference titles should be synchronized. There are 3 tables which may contain links to DiversityReferences: CollectionEvent, CollectionSpecimen and Identification. Choose one of these tables for the synchronization.Then start the query with a click on the Check for differences button.



The form will list all differences found. To update the database click on the Start update button. If you want to have a more detailed look on one of the datasets in the list data you have to check the "Include accession number" checkbox before starting the Query. A button Close form and check dataset in database will appear that will take you back to a single dataset in the database.

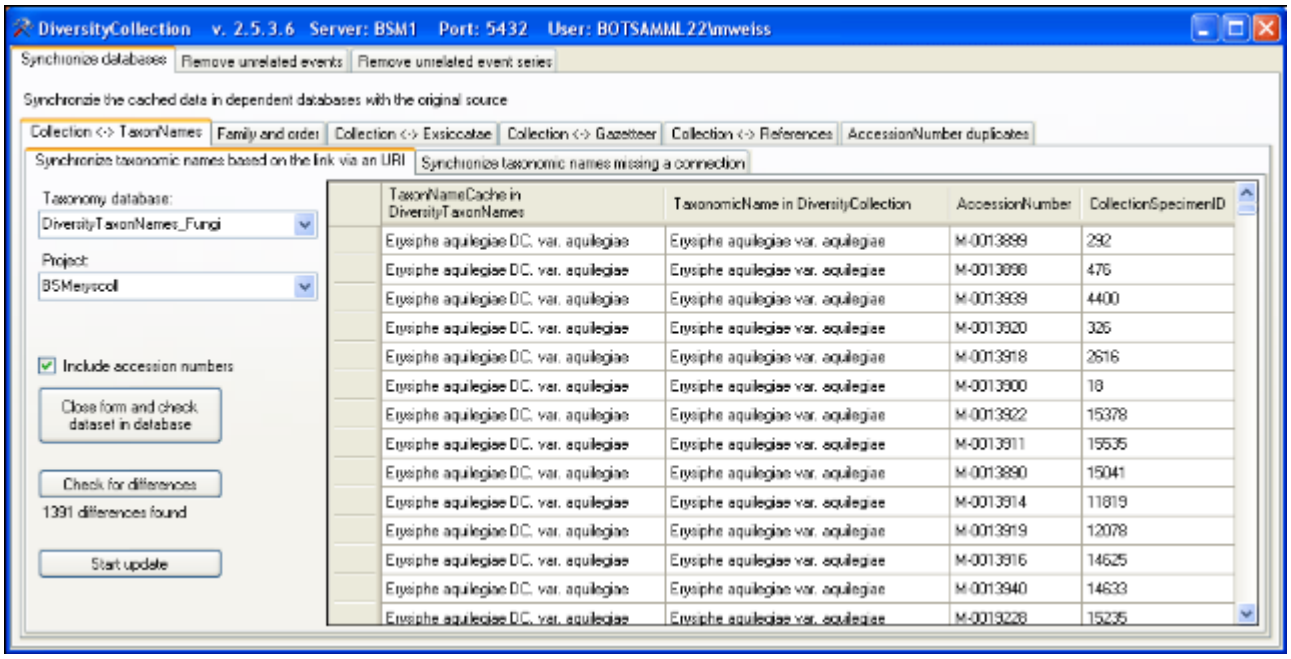
Maintenance - Taxonomic names

To synchronize the entries for the taxonomic names derived from the module DiversityTaxonNames choose **Administration -> Maintenance** from the menu. A form as shown below will open. To synchronize the taxonomic names for entries linked to the module DiversityTaxonNames choose the tab **Collection <-> TaxonNames**. There are two ways to synchronize taxonomic names. You can either **Synchronize taxonomic names based on the link via an URI** for entries where the link to a taxonomic database is allready established or you can **Synchronize taxonomic names missing a connection**, where no link is established and you can query for identical names in one of the databases.

Synchronize taxonomic names based on the link via an URI

The taxonomic names are stored in the field TaxonomicName in the table Identification. Select a taxonomic database and a project that should be synchronized. Than start the query with a click on the

button.



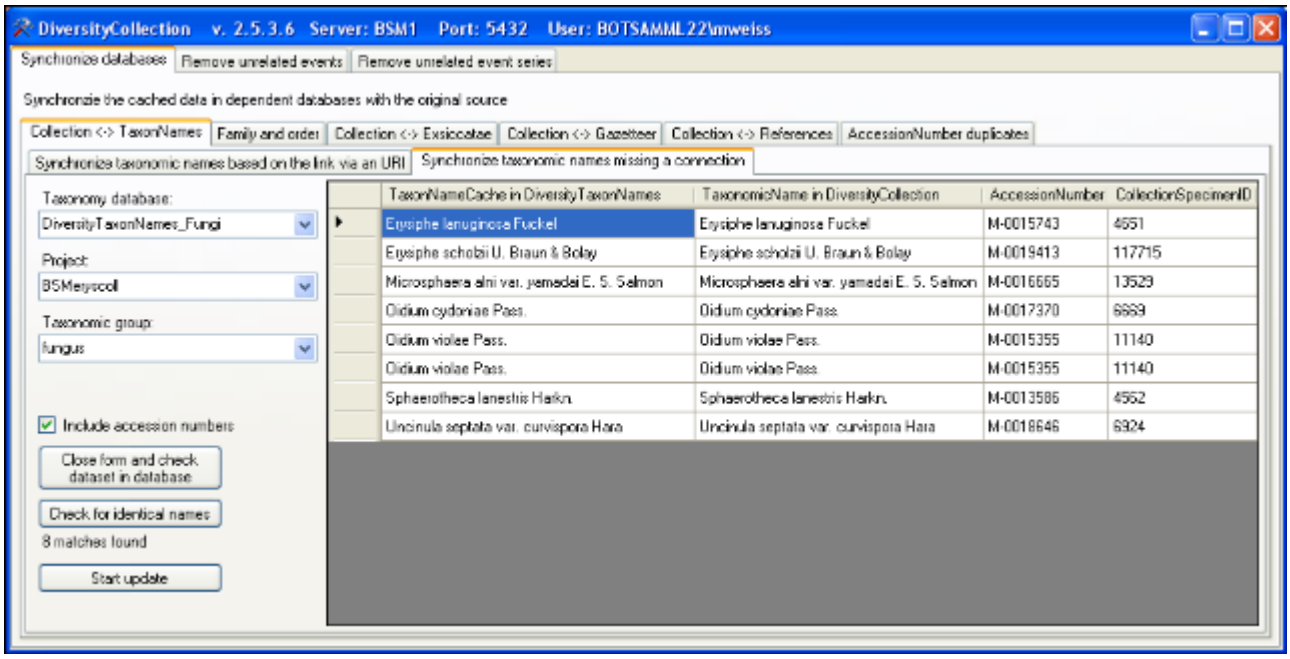
The form will list all differences found. To update the database click on the

button. If you want to have a more detailed look on your data you have to check the "Include accession number" checkbox before starting the Query. A button

will appear that will take you back to a single dataset in the database.

Synchronize taxonomic names missing a connection

Select a taxonomic database, a project and a taxonomic group to search for identical names. Than start the query with a click on the **Check for identical names** button.



The form will list all matches found. To insert the links to the database click on the

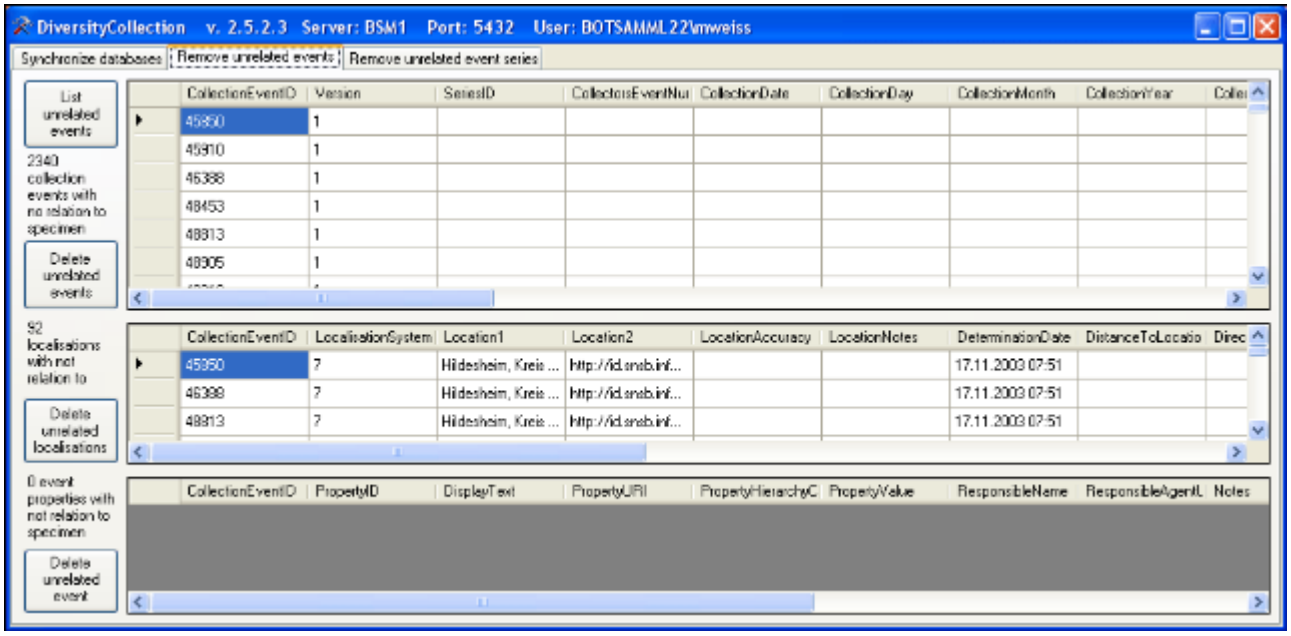
button. If you want to have a more detailed look on your data you have to check the "Include accession number" checkbox before starting the Query. A button

will appear that will take you back to a single dataset in the database.

Maintenance - Unrelated data

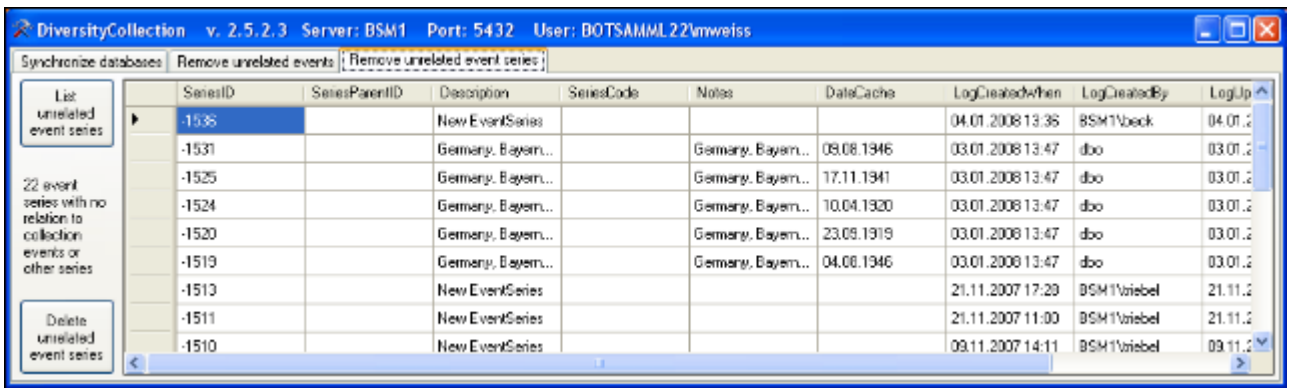
Collection events

Provided you have the proper rights, you can remove unlinked datasets in the tables CollectionEvent and CollectionEventSeries. To delete events that are not linked to any data in the database, use the tab page **[Remove unrelated events]**. These unrelated dataset may e.g. stem from specimen that were transferred to another collection event. Click the **[List unrelated events]** button to list all events that are not related to a specimen. The found collection events will be listed in the upper part of the form. The lower parts show the localisations and the event properties related to these collection events. These must be deleted to delete the related events. Otherwise only the events with no relations to localisations or properties will be deleted.



Collection event series

To delete collection event series with no relation to collection events or other event series, choose the tab page **[Remove unrelated event series]**. Click the **[List unrelated event series]** to list these series and the [Delete unrelated event series] button to delete them. .



Import and export

For the import and export you have several options:

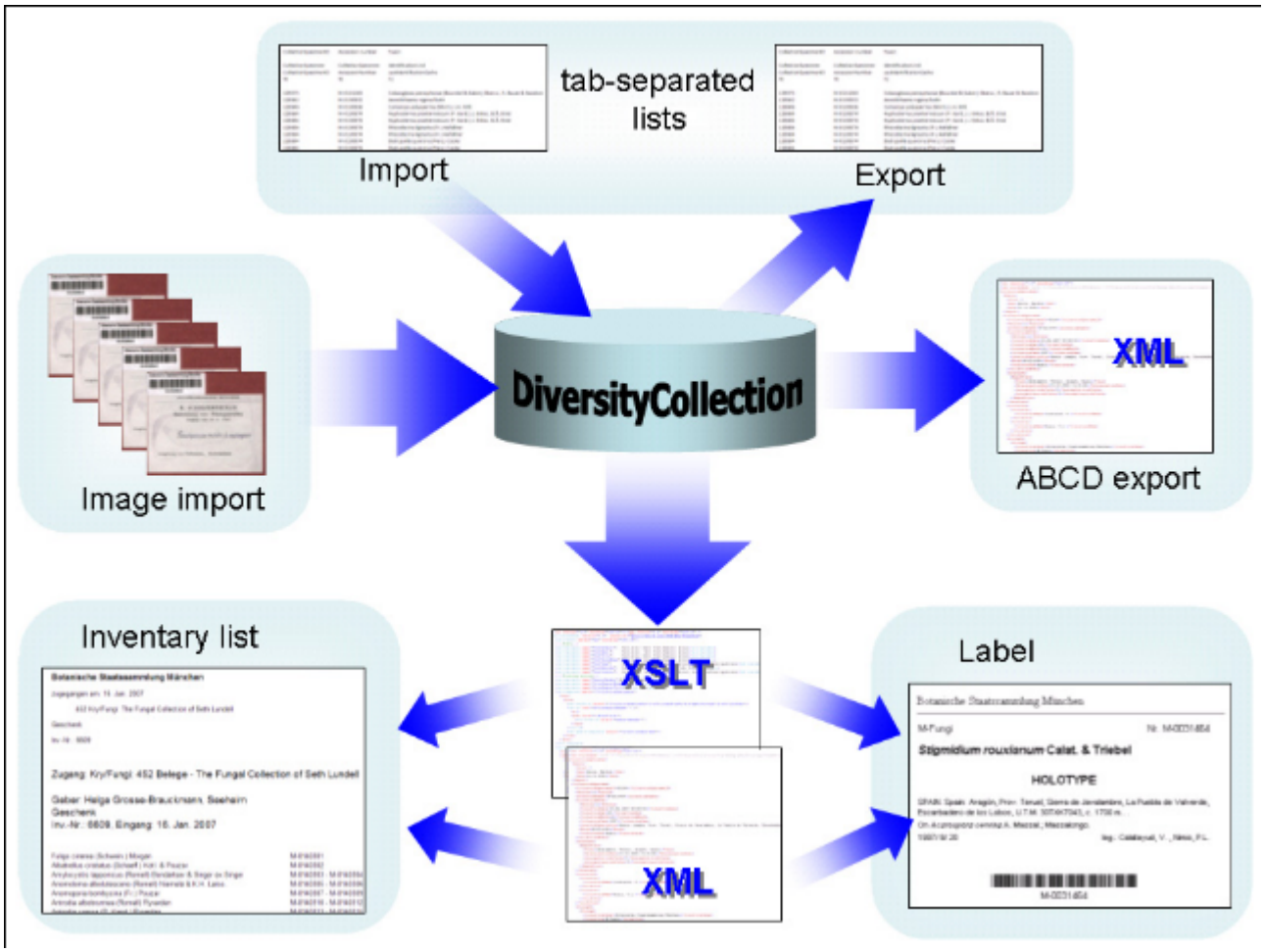
Import of images: Images that are named according to the accession number can be imported together with default informations to the whole batch.

Import and export of tab separated list: To import data from foreign soures and to export data e.g. for reimport after extern editing.

Export of XML data according to ABCD.

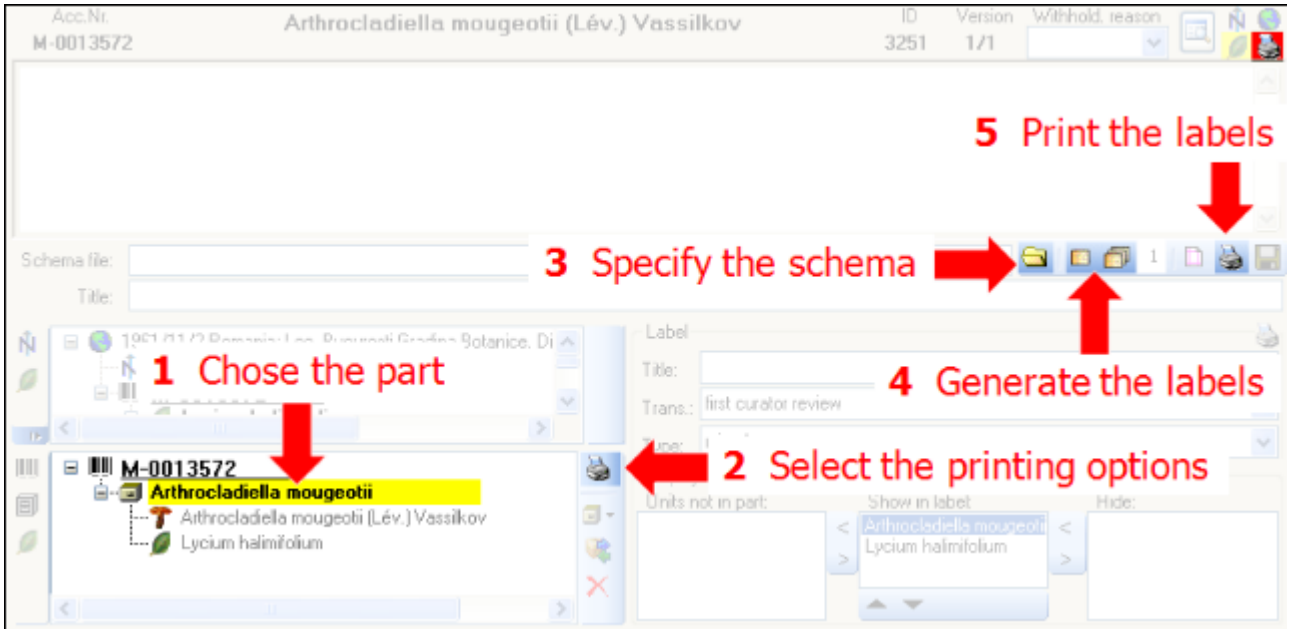
Generation of labels.

Generation of inventary list.

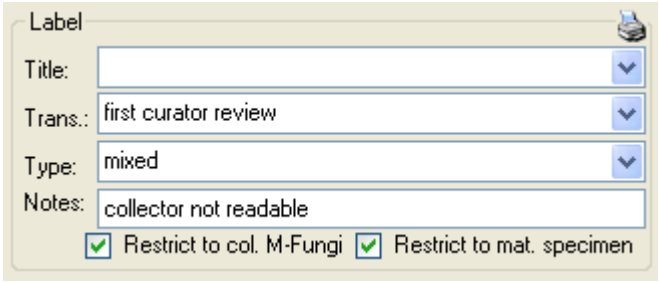


Label

If there is only one part in your specimen click on the printer symbol in the button to switch to the printing mode. If your specimen contains more then one part, choose the part of the specimen for which the label should be generated and click on the button in the right panel. The image area will switch to the label view and in the data area the details related to the label will be accessible. The sequence is shown in the image below.



If you want to print label for all the specimen in the specimen list you can restrict these to the collection and the material category of the current specimen part. These options are active by default as shown in the image below.



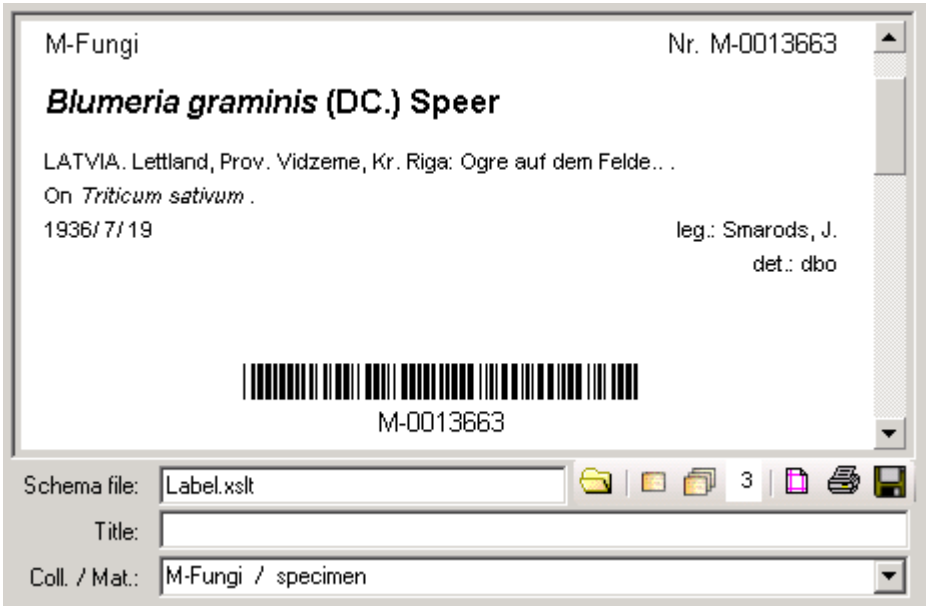
Informations about an existing label are entered in this label section. The data are stored in the table CollectionSpecimen.

label types:
handwriting = all data in handwriting
mixed = data typed with handwriting comments
typed = all data typed

The organisms of a specimen are printed on a label according to the display order.

To print a label for a specimen you have to select a schema file. There are default schema files available in the folder LabelPrinting/Schemas in your application directory. Click on the

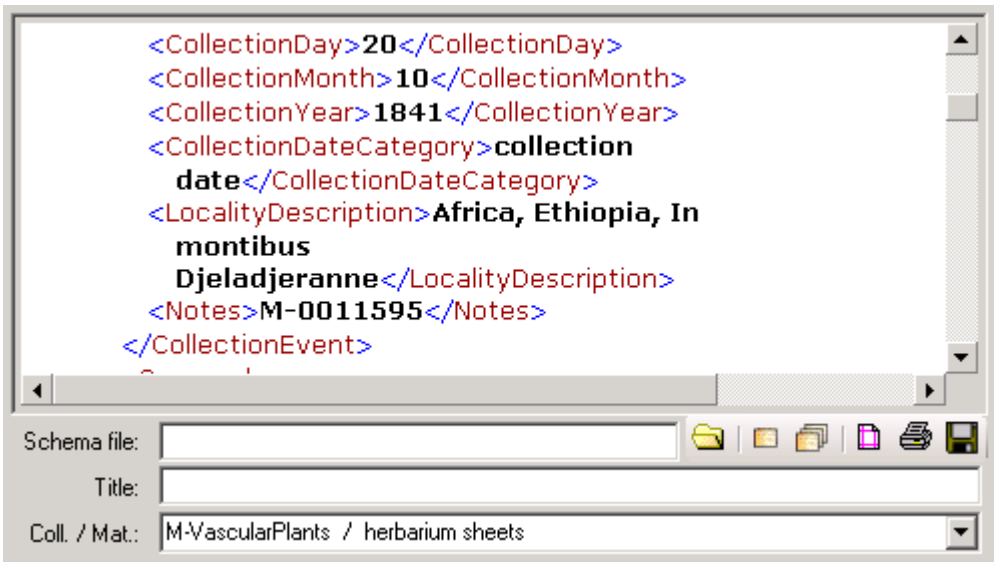
button to open the directory. You will find several prepared schema files among which you can choose resp. change to your own needs or create new ones. The schema file LabelTemplates.xslt provides templates for the other schema files. You may give a title for the print in the field Title. From the Collection and MaterialCategory available for the specimen choose one from the list (Coll./Mat.). To generate the label for the current specimen click on the button. To generate labels for all specimen in your query click on the button. If you need duplicates of your labels change the number in the duplicates box 3 to the desired value. You can print 1 - 99 duplicates of a label. If there are more than 20 specimen in the list, you get a warning if you really want to create all these labels, as this could be somewhat time consuming. The labels are generated as XML files with XSLT-Schema files and depicted in a browser. To print the label click on the button. If you use a different browser than that included in DiversityCollection make sure, that the browser can interpret XSLT/XML (e.g. MS-Internet Explorer).



If you want to save the generated files for later printing click on the button to do this. Note that the program will by default create a file Label.xml in the LabelPrinting directory that will be overwritten everytime you generate a new label. So you have to save the file under a different name or in a different directory to prevent the program to erase these data.

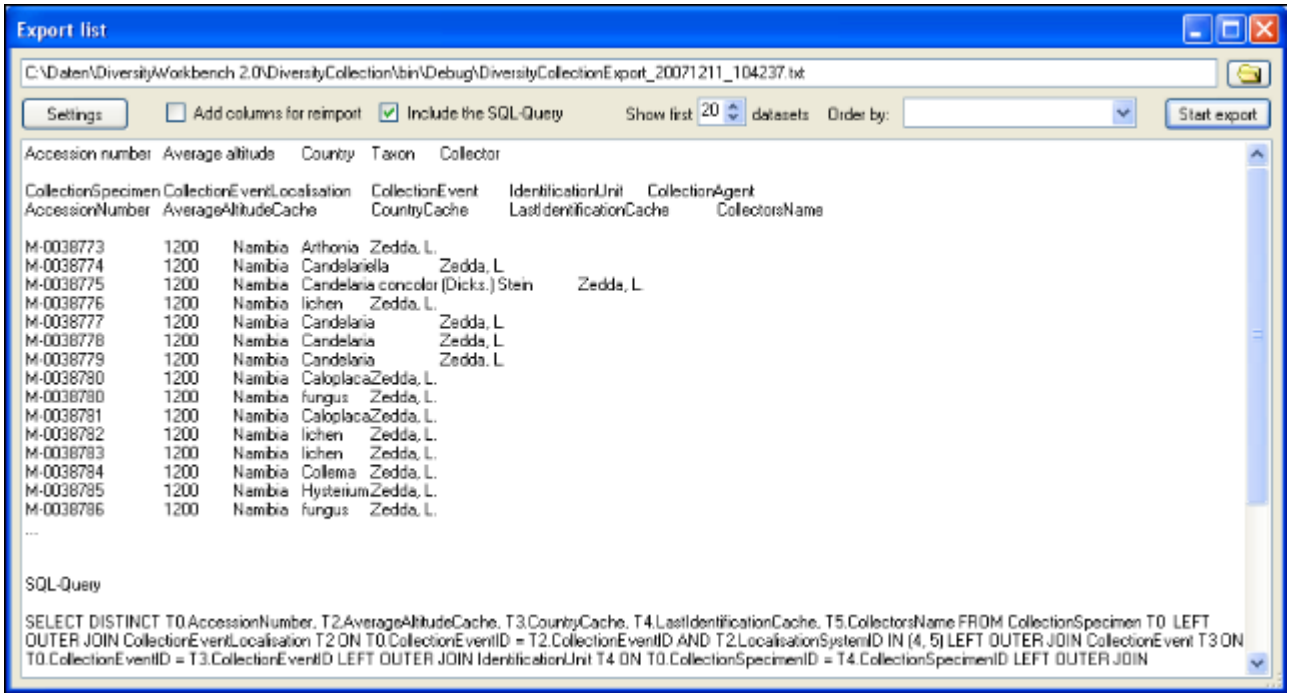
If you use Code 39 for your labels and want to print the barecodes on the labels you need the font code39.ttf , which is included in the DiversityCollection packet. Place this font in the folder where your fonts are stored (e.g.: C:\WINNT\Fonts). If the font is not available, the Barcode will appear as the accession number between two '*' signs.

If you omit to give a Schema file, you will see the generated XML-file as shown in the image below. The XML file is the base for all label types you want to generate. To create your own labels just design your own XSLT-schema file. See e.g. http://www.w3.org/TR/xslt for further informations about schema files.


<CollectionDay>20</CollectionDay>
<CollectionMonth>10</CollectionMonth>
<CollectionYear>1841</CollectionYear>
<CollectionDateCategory>collection date</CollectionDateCategory>
<LocalityDescription>Africa, Ethiopia, In montibus Djeladjeranne</LocalityDescription>
<Notes>M-0011595</Notes>
</CollectionEvent>
Schema file:
Title:
Coll. / Mat.:
M-VascularPlants / herbarium sheets

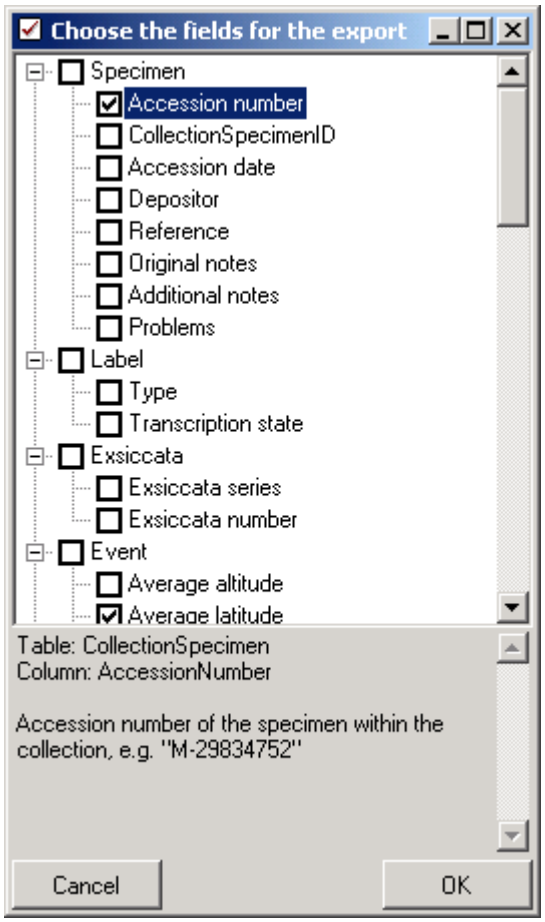
Export

For an analysis of the data within the database, you can export the contents of the datasets listed in the specimen list into a tab separated list. Choose **Data - Export - Export list...** from the menu. A form as shown below will open. To change the preset path of the export file use the button.



To start the export click on the **[Start export]** button. A file will be created in your application directory, containing the exported data. A preview of the data is shown in the lower part of the form. If you check the **include columns for reimport** option, the header will contain an additional line for table names you can use for reimporting the data. If you check the **include the SQL-Query** option, the Text of the command for selecting the datasets will be attached at the end of your report. The form will show the first lines as set in the Show first ... datasets (range: 1 - 99) as a preview. During the first export, the list for the fields for ordering the results **[Order by:]** will be filled. So if you need your results ordered just restart the export after selecting the field after the first export.

To choose the fields you want to see in the export click on the Settings button. A form will open as shown below, where you can choose the fields you want to export.

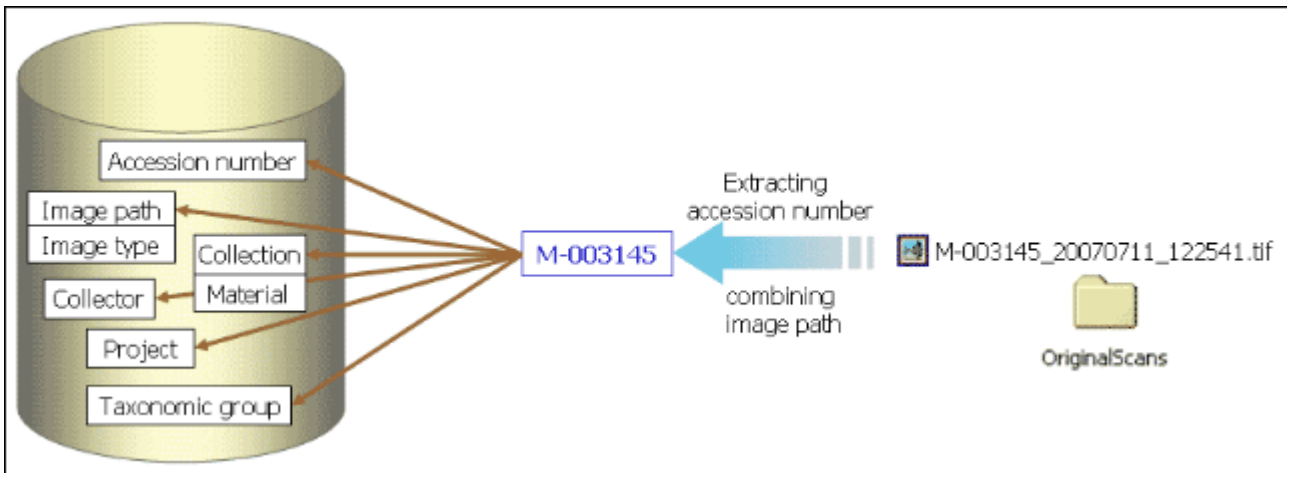


Please be aware, that you will get "cross joins" between the data in your database as in the example shown below. If for example you start an export containing taxa and collectors and you have 3 taxa and 2 collectors in a dataset this will result in 6 lines in the export: every taxon combined with every collector (3 x 2 = 6). To evaluate the data you have to group the results accordingly. The first line in the file corresponds to the description as shown in the form above for the selection of the export fields. The second line shows the tables and the third line the column within the database. The following lines contain the data.

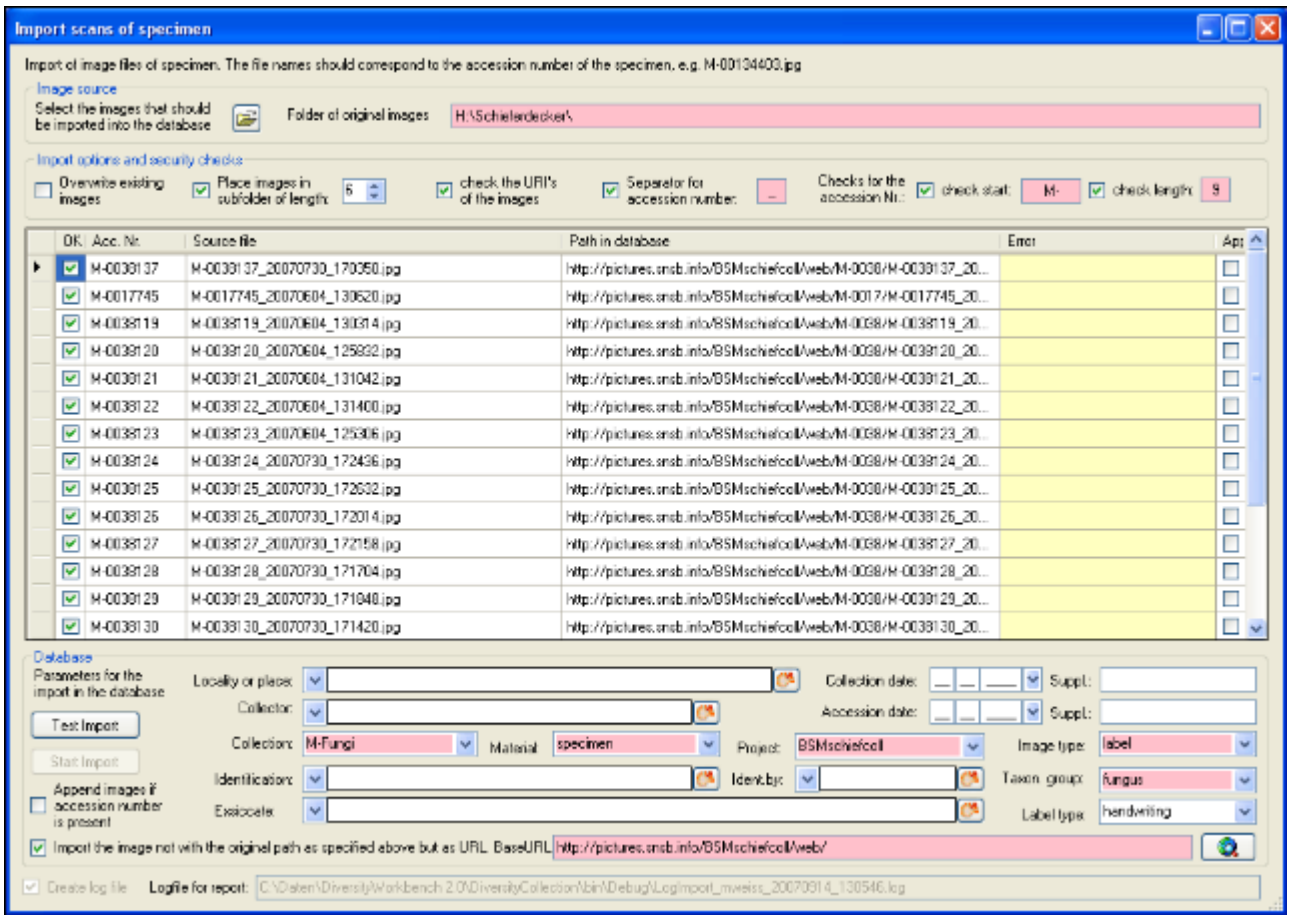
Accession number	**Taxon**
CollectionSpecimen	**IdentificationUnit**
AccessionNumber	**LastIdentificationCache**
M-0013579	Arthrocladiella mougeotii (Lev.) Vassilkov
M-0013579	Arthrocladiella mougeotii (Lev.) Vassilkov
M-0013579	Blumeria graminis (DC.) Speer
M-0013579	Blumeria graminis (DC.) Speer
M-0013579	Lycium barbarum
M-0013579	Lycium barbarum

Import specimen scans

With this import routine, you can import new datasets along with new images into the database. To achieve this the image files should be named corresponding to the accession numbers of the datasets that should be imported and an optional trailing identifier. The image below shows an overview of the whole import process.



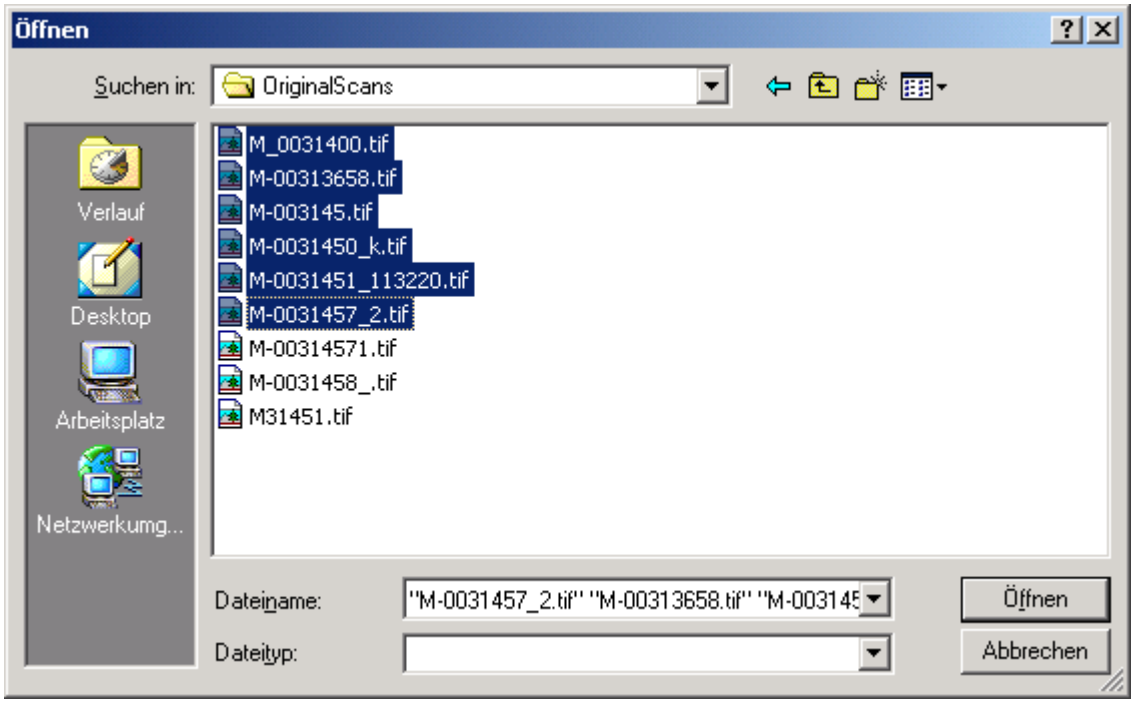
To import images like scanned label from specimen choose **Data - Import - Specimen scans...** from the menu. A form for the import of images will open.



The fields marked with red are obligate.

Image source

To select the images you want to import in the database click on the button. A dialog will open, where you can select the images that you want to import into your Database.



This will enter the selection of the images in the list in the form.

Import options and security checks

If you want to replace datasets for images already in the database, check the **Overwrite existing images** checkbox. If the images are located in a subfolder, that is named according to the first characters of the AccessionNumber, check the **Place images in subfolder of length** checkbox and specify the length of the name of the folder. If you want to check the URI's of the images check the corresponding checkbox. The datasets will than only be imported when the images are available on the web. If the filename next to the accession number contains a trailing identifier, this must be separated by a unique character. Check the corresponding box and enter the separating character. If you want to check if the accession numbers are correct you can check the start and the length of the accession number by checking the appropriate checkboxes.

Image list

The list shows the images selected for the import. If the file name passed the specified checks, the OK field for the image will be checked. Otherwise an explanation for not passing the checks will be shown in the Error section. The list shows the accession number extracted from the file name, the file name of the original image file and if selected the file names for the web and preview images. If a dataset for an image already exists in the database, the checkbox **Append** at the right end of the table will be checked.

Database

You can set several obligate and optional values that will be written in the database for all imported data. The fields marked with red are obligate. These are the collection in which the specimen are located, the material of the specimen, the project, the image type of the scans and the taxonomic group of the main organism in the specimen. If you want to refer to a web source for the label instead of a local file you must give the base URL as well. Optional field are a place according to the DiversityGazetteer and the date of the collection, the collector,

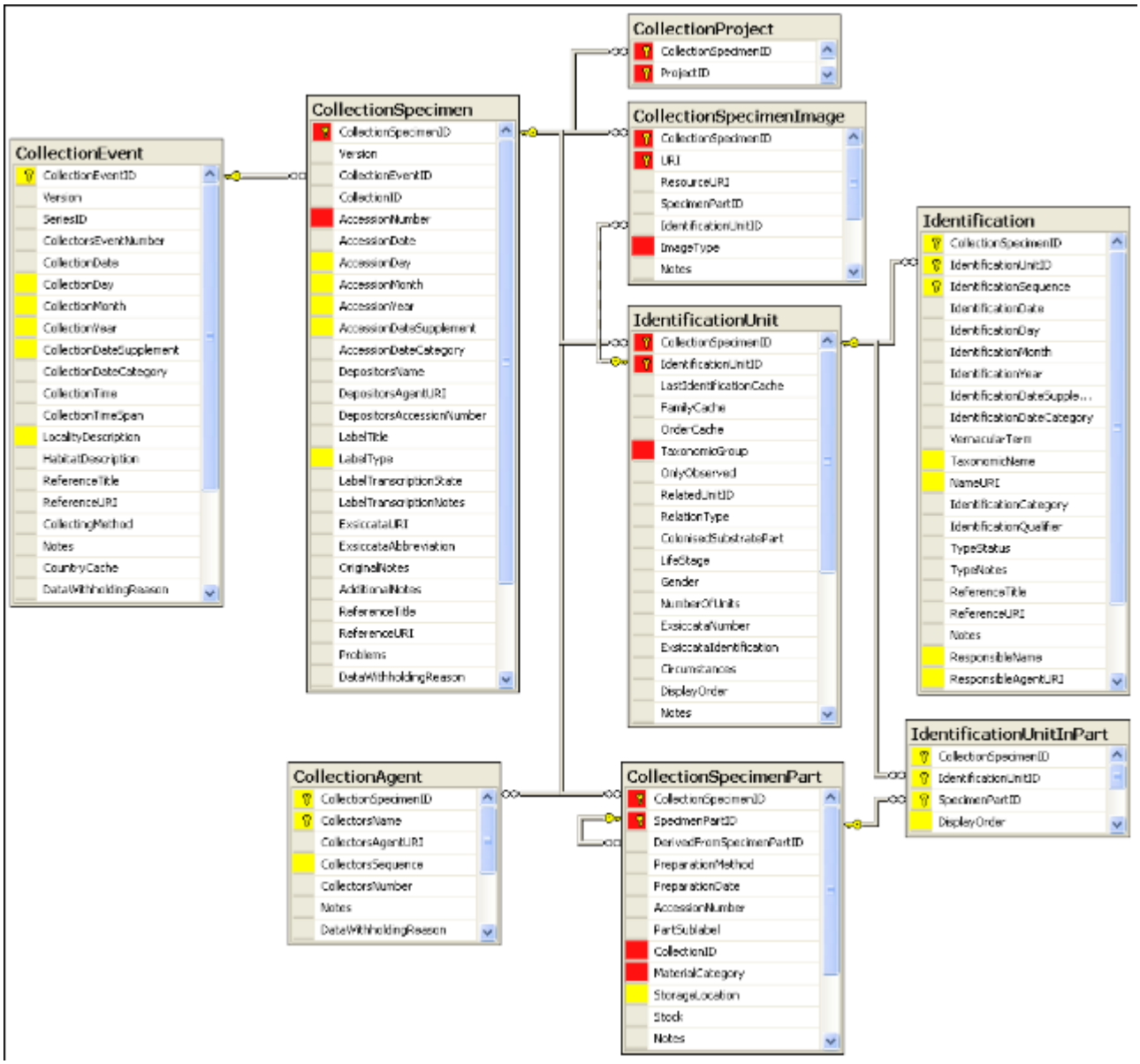
the accession date, an identification, an exsiccatal series and the type of the label. For explanation of the buttons see module related entries .

To test the import and if the images are present in the database click the Test Import button. If everything is fine click the Start Import button to start the import. **Logfile**: To log the list of imported images and any errors during the import, check the Create log file checkbox. This will create a log file with your name and the date and time of the import in the directory where the image files are archived.

The program will check, if an accession number is already present in the database. It will then only import data, when the option Append images if accession number is present is checked. Otherwise these data will not be imported.

If the images will be provided by a webserver, check the appropriate checkbox and specify the folder or click the button of search for the website.

In the overview below the obligate fields are marked with red, the optional fields with yellow.



Import tab-separated lists

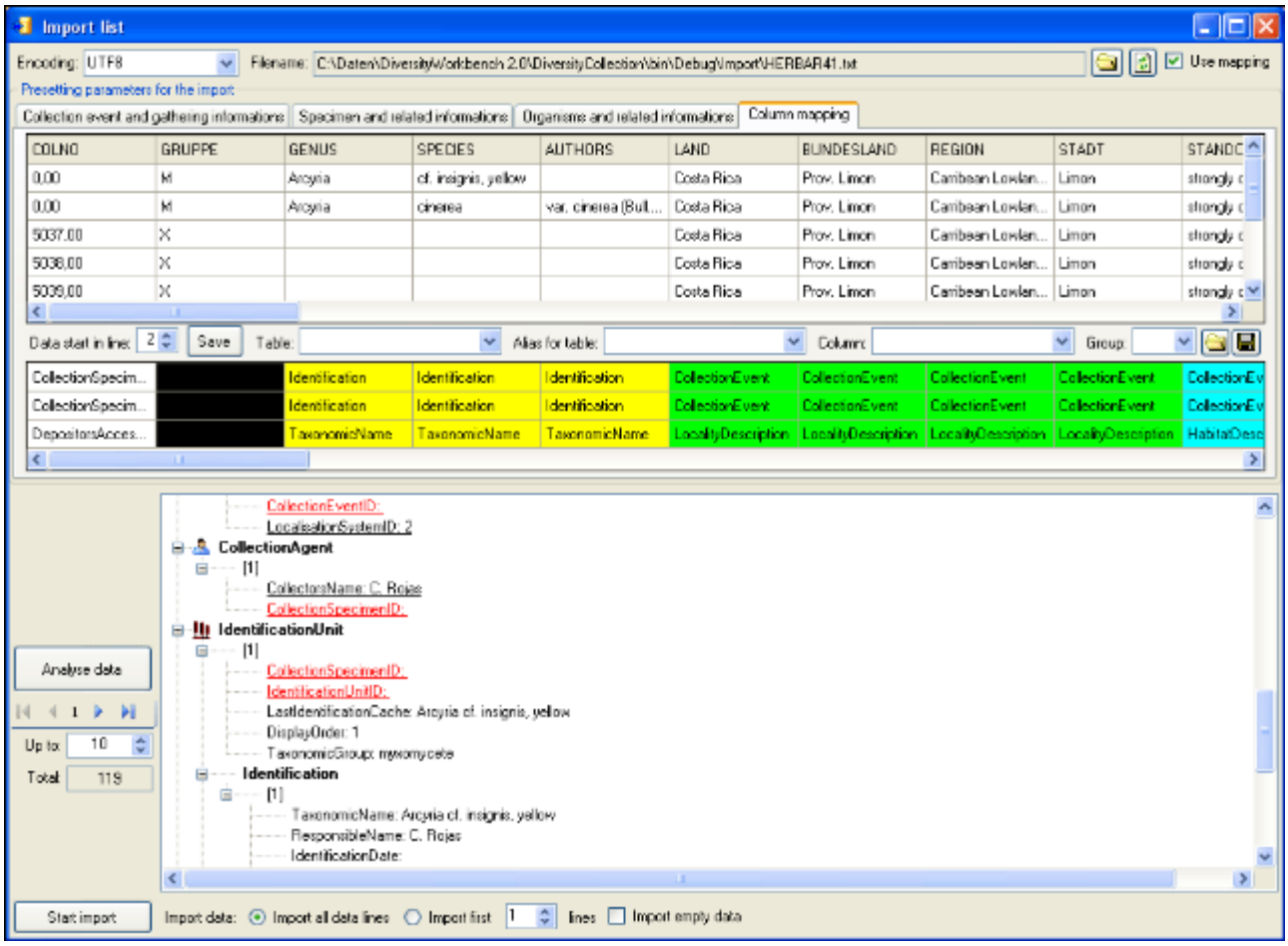
With this import routine, you can import data in tab-separated lists into the database. Choose **Data -> Import -> Import list...** from the menu to open the form for the import. In the form click on the button to select the file with the data you want to import. Use the proper **[encoding]** to ensure that all special characters will be transferred correctly. If you change the encoding after opening the file, click on the button to reload the file with the new encoding.

The content of the file will be shown in the upper part of the **[Column mapping]** tab page. Use the **[Data start in line]** to set the area for the import. Preceeding lines will be ignored and depicted with a gray background as shown below. Every column in your file must be either ignored of mapped to a column in the database. The lower part of the mapping section shows your mapping, while the upper part contains the first lines of your file. You can either use a prepared column mapping or create a new one. To import a previous column mapping, click on the button and choose one of the XML-files.

Column Mapping

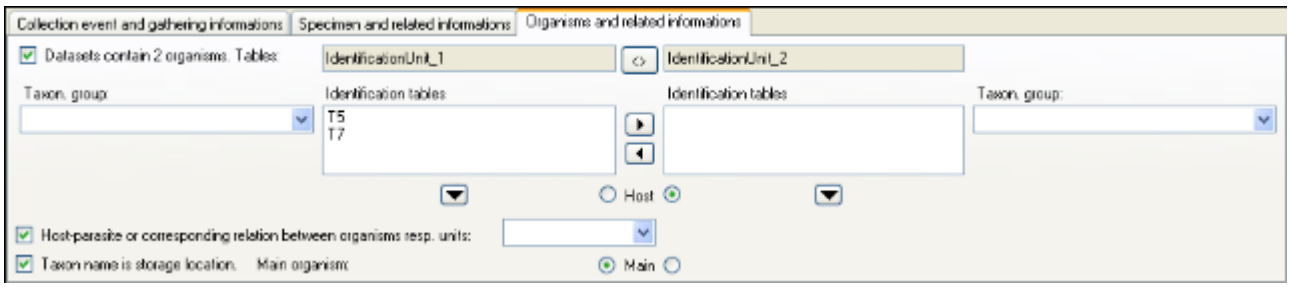
For a column that should not be imported, choose the black color from the Group as shown for the second column in the image below. These columns will be ignored for the analysis and the import. For all other columns choose the **[Table]** and the **[Column]** your data correspond to. If you want to import two different datasets in the same table, you have to change the **[Alias for table]** to a unique value for this dataset.

If some columns should be transfered into one field, you have to choose a group for these column as shown below for columns 3 to 5, 6 to 9, ... To do this, just choose a color from the Group combobox other than white or black. To save the current mapping, click on the button.



To analyse the data in the file click on the **[Analyse]** button. During the analysis you may be asked to give additional information, like the taxonomic groups of the imported organisms. In the upper part of the form you can preset diverse informations that should be imported together with your data from the file. The available options depend on the structure of your data.

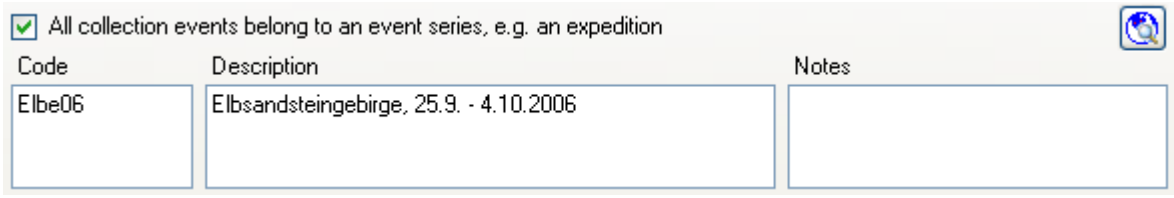
The image below shows the tab page where you can define the relations between two organisms in your data if there is e.g. a parasite and a host. The data tables that are related to an identification are listed with their aliases in the lists for the identification tables. Use the ▶ and ◀ buttons to move them between the lists. Use the ▼ buttons to change the sequence within a list. With the **[Host]** option ○ Host ◉ you define the host and the **[Main]** option defines which organism will be choosen as the first to be printed e.g. on a label.



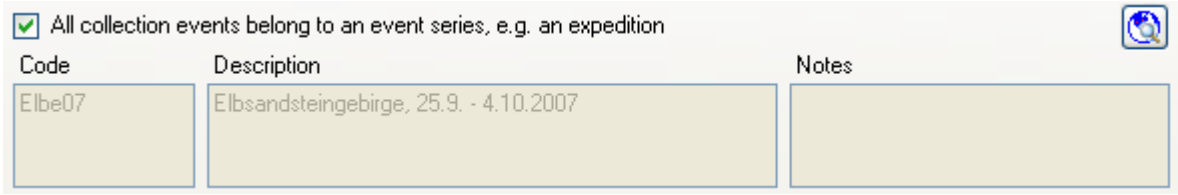
By default for every dataset one collection event will be created.

If the events should be taken together on the bases of the entries in the file, you can choose the second option as shown below. The import will create a new event if there is a change in any of the data related to the collection event, like the description of the locality, the altitude, the coordinates etc.

If all the dataset belong to one event series, e.g. an expedition, you can choose the option as shown below and enter the code and description of the event series in the corresponding fields.



If you want to use an existing event series, click on the button and select a series from the form. The code and description of the selected series will be shown as in the image below.



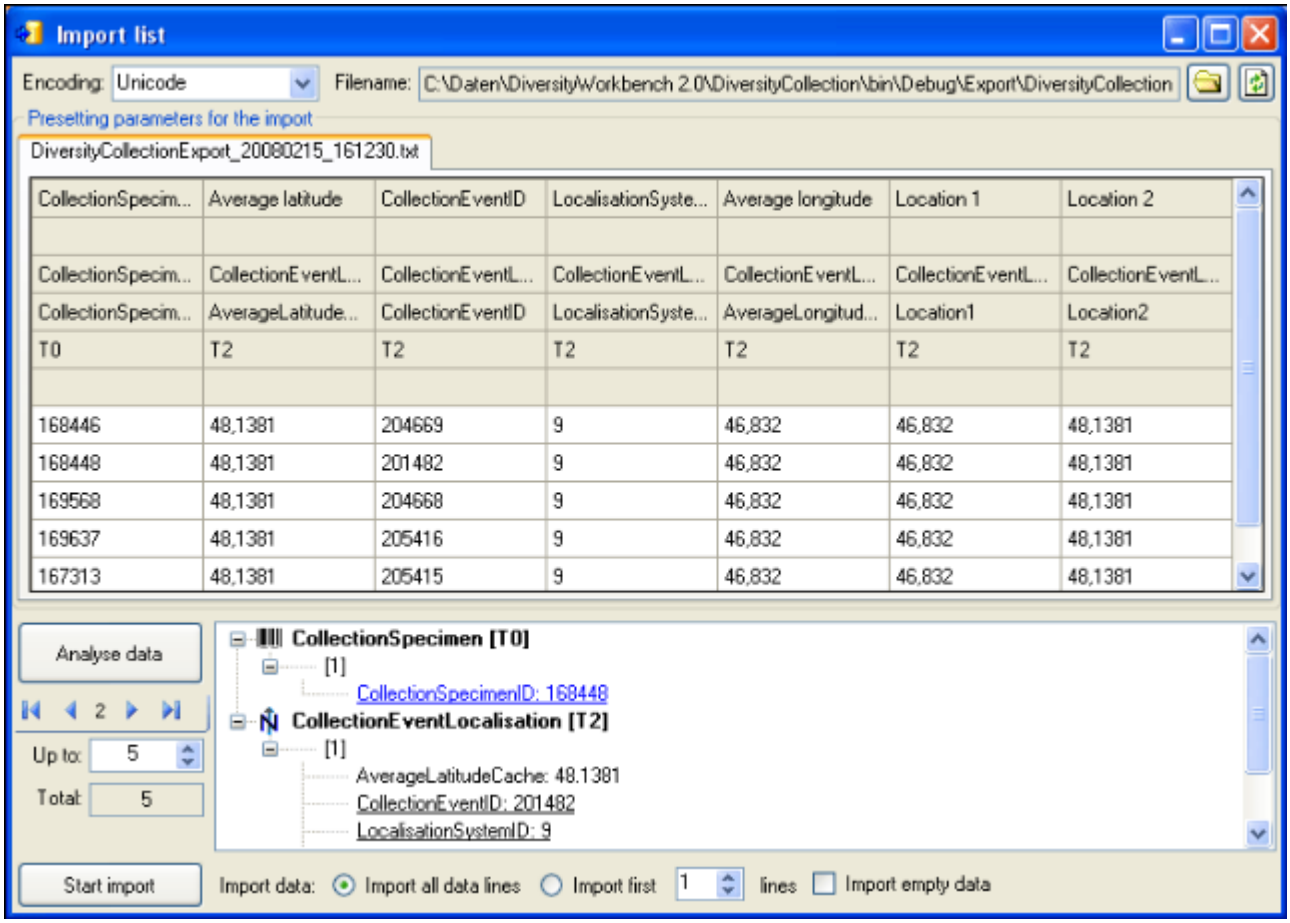
After a successfull analysis, the form will show you the result as shown in the image below. During the analysis, the program will add missing tables and columns necessary for the import. The first dataset will be shown in the form. If you want to check more datasets, set the number of the last position you want to check and click on the **[Analyse]** button. The data are presented according to the tables in the database where they will be inserted. Underlined columns belong the the primary keys of the respecitve tables and are red if missing. These missing values will be generated during the import. If a dataset contains no values and will therefore not be imported into the database, the colums will be shown in blue. To import these entries in any case, choose the **[import empty values]** option. To browse through the datasets in the preview use the 2 buttons.

To import your data, click on the **[Start import]** button. If you want to import just a part of the data, e.g. for a test, choose the **[Import first ... lines]** option.

Reimport tab-separated lists

With this import routine, you can reimport data in tab-separated lists that had been exported from the database into the database. Choose **Data -> Import -> Reimport list...** from the menu to open the form for the reimport. To reimport data, these must have been exported using the option **[add columns for reimport]** to ensure correct header lines in your file. As there is no possibility to change data on the bases of lists, you may export the respective data to a tab separated list, change your values and than reimport it.

In the form click on the button to select the file with the data you want to import. Use the proper **[encoding]** to ensure that all special characters will be transferred correctly. If you change the encoding after opening the file, click on the button to reload the file with the new encoding. The content of the file will be visible as shown below. For a reimport, the data will allways start in line 7, indicated by a gray background of the header lines.



To analyse the data in the file click on the **[Analyse]** button. After a successfull analysis, the form will show you the result as shown in the image below. The first dataset will be shown in the form. If you want to check more datasets, set the number of the last position you want to check and click on the **[Analyse]** button. The data are presented according to the tables in the database where they will be inserted. Underlined columns belong to the primary key of the respecitve tables and are red if missing. These missing values will be generated during the import. If a dataset contains no values and will therefore not be imported into the database respectively updated, the colums will be shown in blue. To import these entries in any case, choose the **[import empty values]** option. To browse through the datasets in the preview use the 2 buttons.

To import your data, click on the **[Start import]** button. If you want to import just a part of the data, e.g. for a test, choose the **[Import first ... lines]** option.

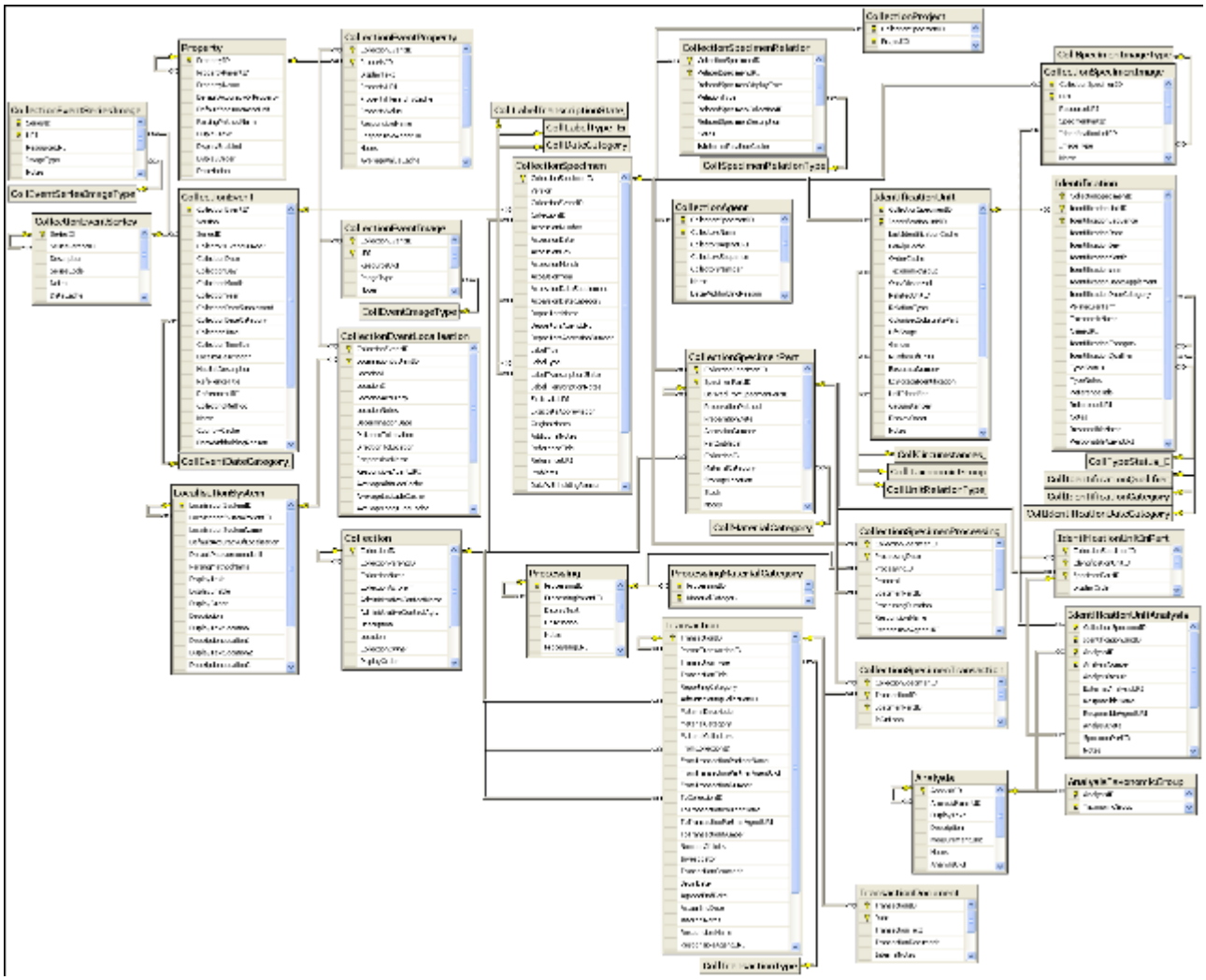
Database

The database for DiversityCollection is based on Microsoft SQL-Server 2005.

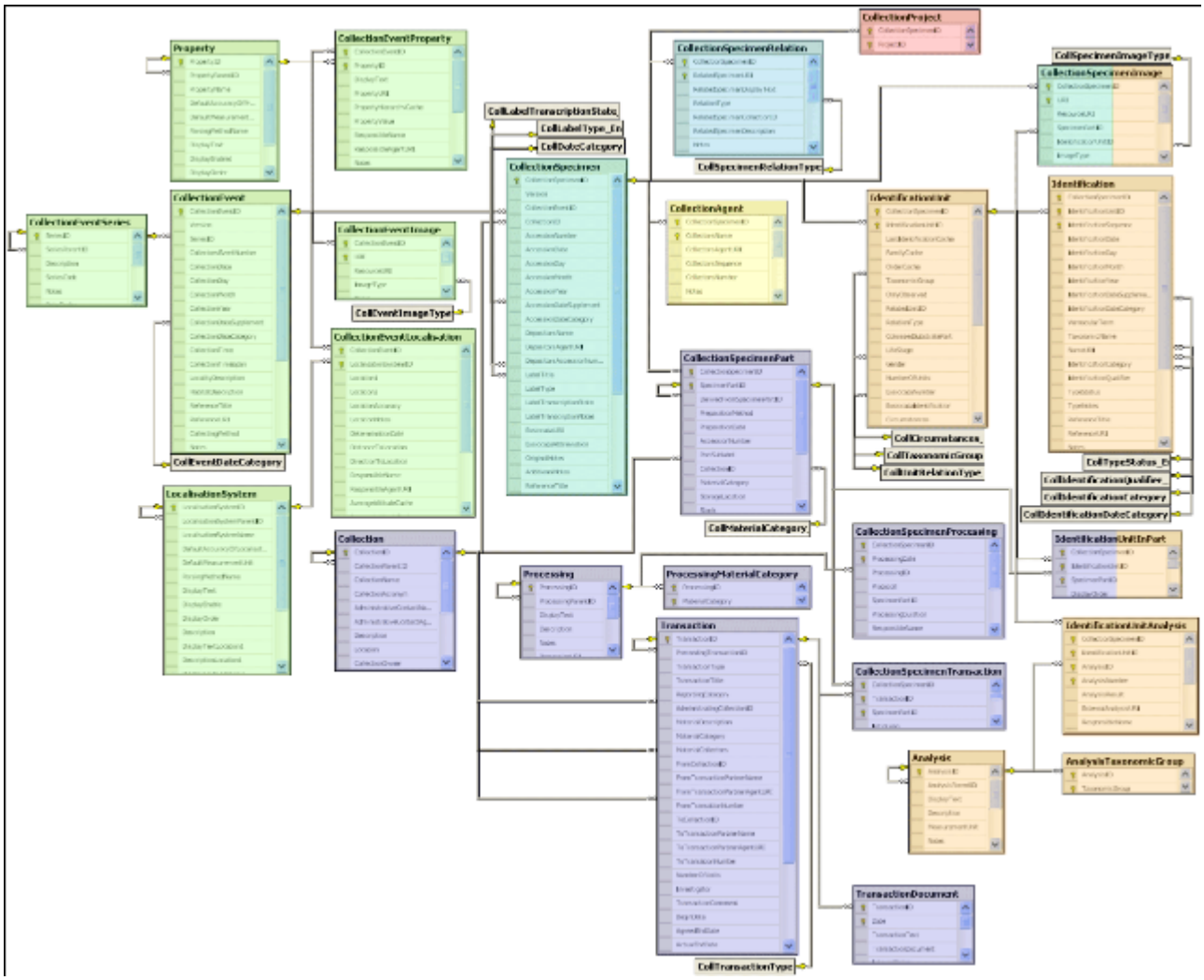
Organisation of the data

The main tables of the database are CollectionEvent corresponding to the event of the collection and CollectionSpecimen holding the collected specimen. Connected to these tables you find tables for holding additional information.

The structure of the whole database is shown in the image below.



In the graphic below the tables of the database are marked according to their logical groups. The central is the table CollectionSpecimen holding the data of the specimen like e.g. the accession number. In the left part you find the tables containing data related to the collection event, and in the right part tables related to the organisms and their identifications. Further logical groups are relation, collectors, projects and storage.



For details about the tables see the table groups about analysis, event, exchange, geography, identification, specimen, storage, .

Further details: application tables, access .

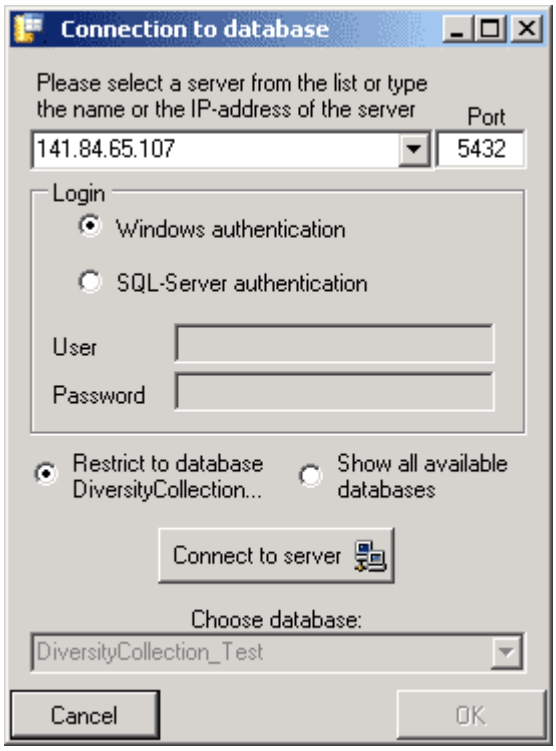
Database - access

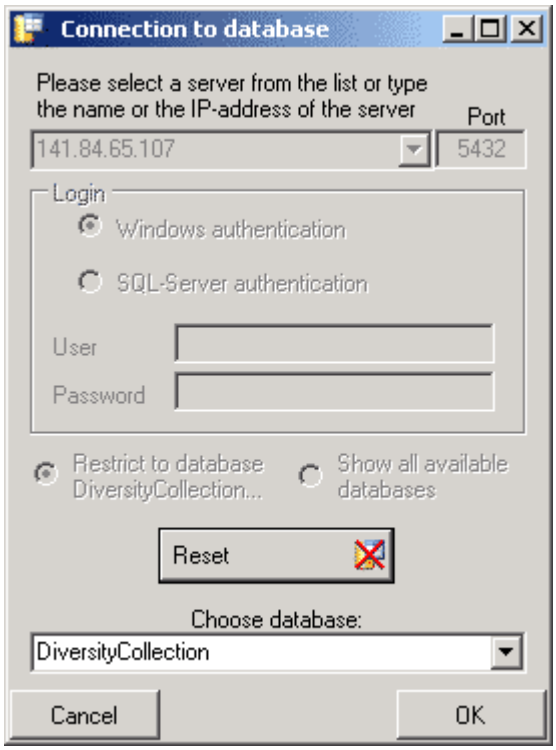
The database for DiversityCollection is based on Microsoft SQL-Server 2005.

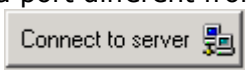
If you are connected to a database this is indicated by the icon of the connection button in left upper corner. If you are not connected this will be indicated by the icon . To access any database, you must specify the server where the database is located. For the configuration of this connection choose **Connection**, **Database...** from the menu or click on the button.

Database name, IP-address and Port

A dialog will open, where you can specify the name or IP-adress and port number of the server and the authentication mode. You can either choose Windows authentication (see left image below) or SQL-Server authentication (see middle image below).





The standard port number for SQL-Server is 1433 and will be set as a default. If the database server is configured using a port different from that port, you must give the port number in the field Port. Click on the 

button to connect to the server. If the connection informations are valid, you can choose a database from the server from the combobox at the base of the form (see right image above). To restart the connecting process click on the

button.

Index

- Analysis
- AnalysisTaxonomicGroup
- Collection
- CollectionAgent
- CollectionCurator
- CollectionEvent
- CollectionEventImage
- CollectionEventLocalisation
- CollectionEventProperty
- CollectionEventSeries
- CollectionEventSeriesImage
- CollectionProject
- CollectionSpecimen
- CollectionSpecimenImage
- CollectionSpecimenPart
- CollectionSpecimenProcessing
- CollectionSpecimenRelation
- CollectionSpecimenTransaction
- Identification
- IdentificationUnit
- IdentificationUnitAnalysis
- IdentificationUnitInPart
- LocalisationSystem
- Processing
- ProcessingMaterialCategory

- Property
- Transaction
- TransactionDocument

Table Analysis

Analysis types used within the database

Column	Data type	Description
AnalysisID	int	ID of the analysis (Primary key)
AnalysisParentID	int	Analysis ID of the parent analysis if it belongs to a certain type documented in this table
DisplayText	nvarchar (50)	Name of the analysis as e.g. shown in user interface
Description	nvarchar (MAX)	Description of the analysis
MeasurementUnit	nvarchar (50)	The measurement unit used for the analysis, e.g. mm, µmol, kg
Notes	nvarchar (MAX)	Notes concerning this analysis
AnalysisURI	varchar (255)	URI referring to an external documentation of the analysis
LogCreatedWhen	datetime	The time when this dataset was created *Default value: getdate()*
LogCreatedBy	nvarchar (50)	Who created this dataset *Default value: user_name()*
LogUpdatedWhen	datetime	The last time when this dataset was updated *Default value: getdate()*
LogUpdatedBy	nvarchar (50)	Who was the last to update this dataset *Default value: user_name()*

Table AnalysisTaxonomicGroup

The types of analysis that are available for a taxonomic group

Column	Data type	Description
AnalysisID	int	Analysis ID, foreign key of table Analysis.
TaxonomicGroup	nvarchar (50)	Taxonomic group the organism identified by this unit belongs to. Groups listed in table

		CollTaxonomicGroup_Enum (= foreign key)
LogUpdatedWhen	datetime	The last time when this dataset was updated *Default value: getdate()*
LogUpdatedBy	nvarchar (50)	Who was the last to update this dataset *Default value: user_name()*

Table Collection

The collections where the specimen are stored

Column	Data type	Description
CollectionID	int	Unique reference ID for the collection (= Primary key)
CollectionParentID	int	For a subcollection within another collection: CollectionID of the collection to which the subcollection belongs. Empty for an independent collection
CollectionName	nvarchar (255)	Name of the collection (e. g. 'Herbarium Kew') or subcollection (e. g. 'cone collection', 'alcohol preservations'). This text should be kept relatively short, use Description for additional information
CollectionAcronym	nvarchar (10)	A unique code for the Collection, e.g. the herbarium code from Index Herbariorum
AdministrativeContactName	nvarchar (255)	The name of the person or organisation responsible for this collection
AdministrativeContactAgentURI	varchar (255)	The URI of the person or organisation responsible for the Collection e.g. as provided by the module DiversityAgents
Description	nvarchar (MAX)	A short description of the collection
Location	nvarchar (255)	Optionally location of the collection, e.g. the number within a file system or a description of the room(s) housing the (sub)collection
CollectionOwner	nvarchar (255)	The owner of the collection as e.g. printed on a label, should be given if CollectionParentID is null
DisplayOrder	smallint	The order in which the entries are displayed. The order may

		be changed at any time, but all values must be unique.
LogCreatedWhen	datetime	The time when this dataset was created *Default value: getdate()*
LogCreatedBy	nvarchar (50)	Who created this dataset *Default value: user_name()*
LogUpdatedWhen	datetime	The last time when this dataset was updated *Default value: getdate()*
LogUpdatedBy	nvarchar (50)	Who was the last to update this dataset *Default value: user_name()*

Table CollectionAgent

The collector(s) of collection specimens

Column	Data type	Description
CollectionSpecimenID	int	Refers to ID of CollectionEvent (= Foreign key and part of primary key)
CollectorsName	nvarchar (255)	Name of the Collector
CollectorsAgentURI	varchar (255)	The URI of the Agent, e.g. as stored within the module DiversityAgents
CollectorsSequence	datetime	The order of collectors in a team. Automatically set by the database system *Default value: getdate()*
CollectorsNumber	nvarchar (50)	Number assigned to a specimen or a batch of specimens by the collector during the collection event (= 'field number')
Notes	nvarchar (MAX)	Notes about the collector, e.g. if the name is uncertain
DataWithholdingReason	nvarchar (255)	If the dataset is withhold, the reason for withholding the data, otherwise null
LogCreatedWhen	datetime	The time when this dataset was created *Default value: getdate()*
LogCreatedBy	nvarchar (50)	Who created this dataset *Default value: user_name()*
LogUpdatedWhen	datetime	The last time when this dataset was updated *Default value: getdate()*
LogUpdatedBy	nvarchar (50)	Who was the last to update this dataset *Default value: user_name()*

Table CollectionCurator

Curators within DiversityCollection, responsible of specimen transactions

Column	Data type	Description
LoginName	nvarchar (50)	A login name which the user uses for access the DivesityWorkbench, Microsoft domains, etc..
AdministratingCollectionID	int	ID for the collection for which the curator has the right to administrate the transaction. Corresponds to AdministratingCollectionID in table Transaction.

Table CollectionEvent

The collection event where the specimen was collected

Column	Data type	Description
CollectionEventID	int	Unique ID for the collection event (= Primary key)
Version	int	The version of the dataset. Automatically set by the system. *Default value: (1)*
SeriesID	int	The ID of the related expedition. Relates to the PK of the table CollectionExpedition (Foreign key).
CollectorsEventNumber	nvarchar (50)	Number assigned to a collection event by the collector (= 'field number')
CollectionDate	datetime	The date of the event calulated from the entries in CollectionDay, -Month and -Year.
CollectionDay	tinyint	The day of the date of the event or when the collection event started
CollectionMonth	tinyint	The month of the date of the event or when the collection event started
CollectionYear	smallint	The year of the date of the event or when the collection event started
CollectionDateSupplement	nvarchar (100)	Verbal or additional collection date information, e.g. 'end of summer 1985', 'first quarter',

		'1888-1892'. The end date if the collection event comprises a period. The time of the event if necessary.
CollectionDateCategory	nvarchar (50)	Category of the date of the identification e.g. "system", "estimated" (= foreign key, see in table CollEventDateCategory_Enum)
CollectionTime	varchar (50)	The time of the event or when the collection event started
CollectionTimeSpan	varchar (50)	The time span in seconds of the collection event
LocalityDescription	nvarchar (MAX)	Locality description of the locality, exactly as written on the original label (i.e. without corrections during data entry)
HabitatDescription	nvarchar (MAX)	Geo-ecological description of the locality, exactly as written on the original label (i.e. without corrections during data entry)
ReferenceTitle	nvarchar (255)	The title of the publication where the collection event was published. Note this is only a cached value where ReferenceURI is present
ReferenceURI	varchar (255)	URI (e.g. LSID) of the source publication where the collection event is published, may e.g. refer to the module DiversityReferences
CollectingMethod	nvarchar (MAX)	Description of the method used for collecting the samples, e.g. traps, moist chambers, drag net
Notes	nvarchar (MAX)	Notes about the collection event
CountryCache	nvarchar (50)	The country where the collection event took place. Cached value derived from an entry in CollectionGeography
DataWithholdingReason	nvarchar (255)	If the dataset is withhold, the reason for withholding the data, otherwise null
LogCreatedWhen	datetime	The time when this dataset was created *Default value: getdate()*
LogCreatedBy	nvarchar (50)	Who created this dataset *Default value: user_name()*
LogUpdatedWhen	datetime	The last time when this dataset was updated *Default value: getdate()*

LogUpdatedBy	nvarchar (50)	Who was the last to update this dataset *Default value: user_name()*

Table CollectionEventImage

The images showing the site of the collection event

Column	Data type	Description
CollectionEventID	int	Unique ID for the collection event (= Primary key)
URI	varchar (255)	The complete URI address of the image. This is only a cached value if ResourceID is available referring to the module DiversityResources
ResourceURI	varchar (255)	The URI of the resource (e.g. see module DiversityResources)
ImageType	nvarchar (50)	Type of the image, e.g. map
Notes	nvarchar (MAX)	Notes to this image concerning the collection event
LogCreatedWhen	datetime	The time when this dataset was created *Default value: getdate()*
LogCreatedBy	nvarchar (50)	Who created this dataset *Default value: user_name()*
LogUpdatedWhen	datetime	The last time when this dataset was updated *Default value: getdate()*
LogUpdatedBy	nvarchar (50)	Who was the last to update this dataset *Default value: user_name()*

Table CollectionEventLocalisation

The geographic localisation of a collection event

Column	Data type	Description
CollectionEventID	int	Refers to the ID of CollectionEvent (= Foreign key and part of primary key)
LocalisationSystemID	int	Refers to the ID of LocalisationSystem (= Foreign key and part of primary key)
Location1	nvarchar (255)	Either a named location selected from a thesaurus (e. g. 'Germany, Bavaria, Kleindingharting') or altitude

		range or other values (e. g. 100-200 m)
Location2	nvarchar (255)	Corresponding value to Location1 e.g. ID or URI of gazetteer or thesaurus
LocationAccuracy	nvarchar (50)	The accuracy of the determination of this locality
LocationNotes	nvarchar (MAX)	Notes on the location
DeterminationDate	smalldatetime	Date of the determination of the geographical localisation
DistanceToLocation	varchar (50)	Distance from the specified place to the real location of the collection event (m)
DirectionToLocation	varchar (50)	Direction from the specified place to the real location of the collection event (Degrees rel. to north)
ResponsibleName	nvarchar (255)	The name of the agent (person or organization) responsible for this entry. *Default value: [dbo].[CurrentUserName]()*
ResponsibleAgentURI	varchar (255)	URI of the person or organisation responsible for the data (see e.g. module DiversityAgents)
AverageAltitudeCache	real	Calculated altitude as parsed from the location fields
AverageLatitudeCache	real	Calculated latitude as parsed from the location fields
AverageLongitudeCache	real	Calculated longitude as parsed from the location fields
LogCreatedWhen	datetime	The time when this dataset was created *Default value: getdate()*
LogCreatedBy	nvarchar (50)	Who created this dataset *Default value: user_name()*
LogUpdatedWhen	datetime	The last time when this dataset was updated *Default value: getdate()*
LogUpdatedBy	nvarchar (50)	Who was the last to update this dataset *Default value: user_name()*

Table CollectionEventProperty

A property of a collection site, e.g. exposition, slope, vegetation. May refer to Diversity Workbench module DiversityScientificTerms

Column	Data type	Description
CollectionEventID	int	Refers to the ID of

		CollectionEvent (= Foreign key and part of primary key)
PropertyID	int	The ID of the descriptor of the collection event, foreign key, see table Descriptor
DisplayText	nvarchar (255)	The text for the property as shown e.g. in a user interface
PropertyURI	varchar (255)	URI referring to an external datasource e.g. DiversityTerminology
PropertyHierarchyCache	nvarchar (MAX)	A cached text of the complete name of the descriptor including superior categories if present
PropertyValue	nvarchar (255)	The value of a captured feature e.g. temperature, pH, vegetation etc. If there is a range this is the lower or first value
ResponsibleName	nvarchar (255)	The name of the agent (person or organization) responsible for this entry. *Default value: [dbo].[CurrentUserName]()*
ResponsibleAgentURI	varchar (255)	URI of the person or organisation responsible for the data (see e.g. module DiversityAgents)
Notes	nvarchar (MAX)	Notes about the property of the colletion site.
AverageValueCache	float	For numeric values - a cached average value according to the
LogCreatedWhen	datetime	The time when this dataset was created *Default value: getdate()*
LogCreatedBy	nvarchar (50)	Who created this dataset *Default value: user_name()*
LogUpdatedWhen	datetime	The last time when this dataset was updated *Default value: getdate()*
LogUpdatedBy	nvarchar (50)	Who was the last to update this dataset *Default value: user_name()*

Table CollectionEventSeries

The series whithin which collection events take place

Column	Data type	Description
SeriesID	int	Primary key. The ID for this expedition (= Primary key)

SeriesParentID	int	The ID of the superior expedition
Description	nvarchar (MAX)	The description of the expedition as it will be printed on e.g. the label
SeriesCode	nvarchar (50)	The user defined code for an expedition
Notes	nvarchar (MAX)	Notes about this expedition
DateCache	datetime	The first date of the depending events, used for sorting the expeditions [controlled by the database]
LogCreatedWhen	datetime	The time when this dataset was created *Default value: getdate()*
LogCreatedBy	nvarchar (50)	Who created this dataset *Default value: user_name()*
LogUpdatedWhen	datetime	The last time when this dataset was updated *Default value: getdate()*
LogUpdatedBy	nvarchar (50)	Who was the last to update this dataset *Default value: user_name()*

Table CollectionEventSeriesImage

The images showing the sites of a collection event series, e.g. an expedition or sampling plot

Column	Data type	Description
SeriesID	int	Unique ID for the collection event series (= Foreign key and part of primary key)
URI	varchar (255)	The complete URI address of the image. This is only a cached value if ResourceID is available referring to the module DiversityResources
ResourceURI	varchar (255)	The URI of the resource (e.g. see module DiversityResources)
ImageType	nvarchar (50)	Type of the image, e.g. map
Notes	nvarchar (MAX)	Notes to this image concerning the collection event
LogCreatedWhen	datetime	The time when this dataset was created *Default value: getdate()*
LogCreatedBy	nvarchar (50)	Who created this dataset *Default value: user_name()*
LogUpdatedWhen	datetime	The last time when this

		dataset was updated *Default value: getdate()*
LogUpdatedBy	nvarchar (50)	Who was the last to update this dataset *Default value: user_name()*

Table CollectionProject

The projects within which the collection specimen were placed

Column	Data type	Description
CollectionSpecimenID	int	Refers to the ID of CollectionSpecimen (= Foreign key and part of primary key)
ProjectID	int	ID of the project to which the specimen belongs (Projects are defined in DiversityProjects)
LogCreatedWhen	datetime	The time when this dataset was created *Default value: getdate()*
LogCreatedBy	nvarchar (50)	Who created this dataset *Default value: user_name()*
LogUpdatedWhen	datetime	The last time when this dataset was updated *Default value: getdate()*
LogUpdatedBy	nvarchar (50)	Who was the last to update this dataset *Default value: user_name()*

Table CollectionSpecimen

The data directly attributed to the collection specimen

Column	Data type	Description
CollectionSpecimenID	int	Unique reference ID for the collection specimen record (primary key)
Version	int	The version of the dataset *Default value: (1)*
CollectionEventID	int	Refers to the ID of CollectionEvent (= Foreign key and part of primary key)
CollectionID	int	ID of the Collection as stored in table Collection (= foreign key, see table Collection)
AccessionNumber	nvarchar (50)	Accession number of the specimen within the collection, e.g. "M-29834752"
AccessionDate	datetime	The date of the accession

		calculated from the entries in AccessionDay, -Month and -Year
AccessionDay	tinyint	The day of the date when the specimen was acquired in the collection
AccessionMonth	tinyint	The month of the date when the specimen was acquired in the collection
AccessionYear	smallint	The year of the date when the specimen was acquired in the collection
AccessionDateSupplement	nvarchar (255)	Verbal or additional accession date information, e.g. 'end of summer 1985', 'first quarter', '1888-1892'
AccessionDateCategory	nvarchar (50)	Category of the date of the identification e.g. "system", "estimated" (= foreign key, see in table xColl_DateCategory_Enum)
DepositorsName	nvarchar (255)	The name of the depositor(s) (person or organization responsible for deposition). Where entire collections are deposited, this should also contain the collection name (e.g. 'Herbarium P. Döbbler')
DepositorsAgentURI	varchar (255)	The URI of the depositor(s) (person or organization responsible for deposition)
DepositorsAccessionNumber	nvarchar (50)	Accession number of the specimen within the previous or original collection, e.g. 'D-23948'
LabelTitle	nvarchar (255)	The title of the label e.g. for printing labels.
LabelType	nvarchar (50)	Printed, typewritten, typewritten with handwriting added, entirely in handwriting, etc.
LabelTranscriptionState	nvarchar (50)	The state of the transcription of a label into the database: 'Not started', 'incomplete', 'complete'
LabelTranscriptionNotes	nvarchar (255)	User defined notes concerning the transcription of the label into the database
ExsiccataURI	varchar (255)	If specimen is an exsiccata: The URI of the Exsiccata series, e.g. as stored within the DiversityExsiccata module
ExsiccataAbbreviation	nvarchar (255)	If specimen is an exsiccata: Standard abbreviation of the

		exsiccata (not necessarily a unique identifier; editors or publication places may change over time)
OriginalNotes	nvarchar (MAX)	Notes found on the label of the specimen, by the original collector or from a later revision
AdditionalNotes	nvarchar (MAX)	Additional notes made by the editor of the specimen record, e. g. 'doubtful identification/locality'
ReferenceTitle	nvarchar (255)	The title of the publication where the specimen was published. Note this is only a cached value where ReferenceURI is present
ReferenceURI	varchar (255)	URI (e.g. LSID) of reference where specimen is published, e.g. referring to the module DiversityReferences
Problems	nvarchar (255)	Description of a problem that occurred during data editing. Typically these entries should be deleted after help has been obtained. Do not enter scientific problems here; use AdditionalNotes for such permanent problems!
DataWithholdingReason	nvarchar (255)	If the dataset is withhold, the reason for withholding the data, otherwise null
LogCreatedWhen	datetime	The time when this dataset was created *Default value: getdate()*
LogCreatedBy	nvarchar (50)	Who created this dataset *Default value: user_name()*
LogUpdatedWhen	datetime	The last time when this dataset was updated *Default value: getdate()*
LogUpdatedBy	nvarchar (50)	Who was the last to update this dataset *Default value: user_name()*

Table CollectionSpecimenImage

The images of a collection specimen or of an identification unit within this specimen

Column	Data type	Description
CollectionSpecimenID	int	Refers to the ID of CollectionSpecimen (= Foreign key and part of primary key)

URI	varchar (255)	The complete URI address of the image. This is only a cached value if ResourceID is available referring to the module DiversityResources
ResourceURI	varchar (255)	The URI of the image, e.g. as stored in the module DiversityResources.
SpecimenPartID	int	Optional: If the dataset is not related to a part of a specimen, the ID of a related part (= foreign key)
IdentificationUnitID	int	If image refers to only on out of several identification units for a specimen, refers to the ID of an IdentificationUnit for a CollectionSpecimen (= foreign key)
ImageType	nvarchar (50)	Type of the image, e.g. label
Notes	nvarchar (MAX)	Notes about the specimen image
LogCreatedWhen	datetime	The time when this dataset was created *Default value: getdate()*
LogCreatedBy	nvarchar (50)	Who created this dataset *Default value: user_name()*
LogUpdatedWhen	datetime	The last time when this dataset was updated *Default value: getdate()*
LogUpdatedBy	nvarchar (50)	Who was the last to update this dataset *Default value: user_name()*

Table CollectionSpecimenPart

Parts of a collection specimen. Includes a possible hierarchy of the parts

Column	Data type	Description
CollectionSpecimenID	int	Refers to the ID of CollectionSpecimen (= Foreign key and part of primary key)
SpecimenPartID	int	ID for a part of a specimen (part of primary key) *Default value: (1)*
DerivedFromSpecimenPartID	int	SpecimenPartID of the specimen from which the current specimen is derived from
PreparationMethod	nvarchar (MAX)	The method used for the preparation of the part of the specimen, e.g. the inoculation method for cultures

PreparationDate	datetime	The date and time when the part was preparated e.g when it was separated from the source object
AccessionNumber	nvarchar (50)	Accession number of the part of the specimen within the collection if it is different from the accession number of the specimen as stored in the table CollectionSpecimen, e.g. "M-29834752"
PartSublabel	nvarchar (50)	The label for a part of a specimen, e.g. if duplicats of a specimen have a separate number
CollectionID	int	ID of the Collection as stored in table Collection (= foreign key, see table Collection)
MaterialCategory	nvarchar (50)	Material category of specimen. Examples: 'herbarium sheets', 'drawings', 'microscopic slides' etc. (= foreign key, see table CollMaterialCategory_Enum) *Default value: N'specimen'*
StorageLocation	nvarchar (255)	A code identifying the place where the specimen is stored within the collection. Frequently the accepted scientific name is used as storage location code.
Stock	tinyint	Number of stock units if the specimen is stored in separated units e.g. several boxes or vessels
Notes	nvarchar (MAX)	Notes concerning the storage of the sample
LogCreatedWhen	datetime	The time when this dataset was created *Default value: getdate()*
LogCreatedBy	nvarchar (50)	Who created this dataset *Default value: user_name()*
LogUpdatedWhen	datetime	The last time when this dataset was updated *Default value: getdate()*
LogUpdatedBy	nvarchar (50)	Who was the last to update this dataset *Default value: user_name()*

Table CollectionSpecimenProcessing

The processing that was applied to a collection specimen

Column	Data type	Description
CollectionSpecimenID	int	Refers to ID of CollectionSpecimen (= Foreign key and part of primary key)
ProcessingDate	datetime	Date and time of the start of the processing *Default value: getdate()*
ProcessingID	int	ID of the processing. Refers to ProcessingID in table Processing (foreign key) *Default value: (1)*
Protocoll	nvarchar (100)	The label of the processing protocoll
SpecimenPartID	int	Optional: If the dataset is related to a part of a specimen, the ID of a related part (= foreign key, see table CollectionSpecimenPart)
ProcessingDuration	varchar (50)	The duration of the processing in seconds
ResponsibleName	nvarchar (255)	Name of the person or institution responsible for the determination *Default value: [dbo].[CurrentUserName]()*
ResponsibleAgentURI	varchar (255)	URI of the person or institution responsible for the determination (= foreign key) as stored in the module DiversityAgents.
Notes	nvarchar (MAX)	Notes about the processing
LogCreatedWhen	datetime	The time when this dataset was created *Default value: getdate()*
LogCreatedBy	nvarchar (50)	Who created this dataset *Default value: user_name()*
LogUpdatedWhen	datetime	The last time when this dataset was updated *Default value: getdate()*
LogUpdatedBy	nvarchar (50)	Who was the last to update this dataset *Default value: user_name()*

Table CollectionSpecimenRelation

The relations of a collection specimen to other collection specimen

Column	Data type	Description
CollectionSpecimenID	int	Unique reference ID for the collection specimen record (primary key)
RelatedSpecimenURI	varchar (255)	URI of the related specimen

RelatedSpecimenDisplayText	varchar (255)	The name of a related specimen as shown e.g. in a user interface
RelationType	nvarchar (50)	Type of the relation between the specimen (= foreign key, see table CollRelationType_Enum)
RelatedSpecimenCollectionID	int	ID of the Collection as stored in table Collection (= foreign key, see table Collection)
RelatedSpecimenDescription	nvarchar (MAX)	Description of the related specimen
Notes	nvarchar (MAX)	Notes on the relation to the specimen
IsInternalRelationCache	bit	If the relation represents a connection between specimen in this database *Default value: (1)*
LogCreatedWhen	datetime	The time when this dataset was created *Default value: getdate()*
LogCreatedBy	nvarchar (50)	Who created this dataset *Default value: user_name()*
LogUpdatedWhen	datetime	The last time when this dataset was updated *Default value: getdate()*
LogUpdatedBy	nvarchar (50)	Who was the last to update this dataset *Default value: user_name()*

Table CollectionSpecimenTransaction

The transactions in which a specimen was involved

Column	Data type	Description
CollectionSpecimenID	int	Refers to ID of CollectionSpecimen (= Foreign key and part of primary key)
TransactionID	int	Unique ID for the transaction (= Foreign key and part of primary key)
SpecimenPartID	int	Optional: If the dataset is related to a part of a specimen, the ID of a related part (= foreign key, see table CollectionSpecimenPart)
IsOnLoan	bit	True if a specimen is on loan
LogInsertedBy	nvarchar (50)	Name of user who first entered (typed or imported) the data. *Default value: user_name()*
LogInsertedWhen	smalldatetime	Date and time when the data

		were first entered (typed or imported) into this database. *Default value: getdate()*
LogUpdatedBy	nvarchar (50)	Name of user who last updated the data. *Default value: user_name()*
LogUpdatedWhen	smalldatetime	Date and time when the data were last updated. *Default value: getdate()*

Table Identification

The identifications of the organisms within a specimen

Column	Data type	Description
CollectionSpecimenID	int	Refers to the ID of CollectionSpecimen (= Foreign key and part of primary key)
IdentificationUnitID	int	Refers to the ID of IdentficationUnit (= foreign key and part of primary key)
IdentificationSequence	smallint	The sequence of the identifications. The last identification (having the highest sequence) is regarded as valid *Default value: (1)*
IdentificationDate	datetime	The date of the identification calculated from the entries in IdentificationDay, -Month and -Year
IdentificationDay	tinyint	The day of the identification event
IdentificationMonth	tinyint	The month of the identification event
IdentificationYear	smallint	The year of the identification event. The year may be empty if only the day or month are known.
IdentificationDateSupplement	nvarchar (255)	Verbal or additional identification date information, e.g. 'end of summer 1985', 'first quarter', '1888-1892'
IdentificationDateCategory	nvarchar (50)	Category of the date of the identification e.g. "system", "estimated" (= foreign key, see in table CollDateCategory_Enum)
VernacularTerm	nvarchar (255)	Name or term other than a taxonomic (= scientific) name, e.g. 'pine', 'limestone', 'conifer', 'hardwood'
TaxonomicName	nvarchar (255)	Valid name of the species

		(including the taxonomic author where available. Example: 'Rosa canina L.'
NameURI	varchar (255)	The URI of the taxonomic name, e.g. as provided by the module DiversityTaxonNames.
IdentificationCategory	nvarchar (50)	Category of the identification e.g. 'determination', 'confirmation', 'absence' (= foreign key, see table CollIdentificationCategory_Enum)
IdentificationQualifier	nvarchar (50)	Qualification of the identification e.g. "cf."," aff.", "sp. nov." (= foreign key, see table CollIdentificationQualifier_Enum)
TypeStatus	nvarchar (50)	If identification unit is type of a taxonomic name: holotype, syntype, etc. (= foreign key, see table CollTypeStatus_Enum)
TypeNotes	nvarchar (MAX)	Notes concerning the typification of this specimen
ReferenceTitle	nvarchar (255)	Publications or authoritative opinions of scientist used during the identification process. Example: enter 'Schmeil-Fitschen 1995' if this field flora was used.
ReferenceURI	varchar (255)	The URI of the reference e.g. as provided by the module DiversityReferences
Notes	nvarchar (MAX)	User defined notes, e.g. the reason for a re-determination / change of the name, etc.
ResponsibleName	nvarchar (255)	Name of the person or institution responsible for the determination *Default value: [dbo].[CurrentUserName]()*
ResponsibleAgentURI	varchar (255)	URI of the person or institution responsible for the determination (= foreign key) as stored in the module DiversityAgents.
LogCreatedWhen	datetime	Date and time when the dataset was created *Default value: getdate()*
LogCreatedBy	nvarchar (50)	Login of the user who created the dataset *Default value: user_name()*
LogUpdatedWhen	datetime	Date and time when the dataset was changed

		Default value: getdate()
LogUpdatedBy	nvarchar (50)	Login of the user who changed the dataset *Default value: user_name()*

Table IdentificationUnit

Organism that is present in or on a collection specimen

Column	Data type	Description
CollectionSpecimenID	int	Refers to the ID of CollectionSpecimen (= Foreign key and part of primary key)
IdentificationUnitID	int	ID of the identification unit (= part of Primary key). Usually one of possibly several organisms present on the collection specimen. Example: parasite with hyperparasite on plant leaf = 3 units,
LastIdentificationCache	nvarchar (255)	The last identification as entered in table Identification
FamilyCache	nvarchar (255)	A cached value of the family of the taxon of the last identification. Can be set by the editor if NameURI in table Identification is NULL, otherwise set by the system.
OrderCache	nvarchar (255)	A cached value of the order of the taxon of the last identification. Can be set by the editor if NameURI in table Identification is NULL, otherwise set by the system.
TaxonomicGroup	nvarchar (50)	Taxonomic group the organism identified by this unit belongs to. Groups listed in table CollTaxonomicGroup_Enum (= foreign key)
OnlyObserved	bit	True if the organism was only observed rather than collected. It is therefore not present on the preserved specimen. Example: Tree under which the collected mycorrhizal fungus grew. *Default value: (0)*
RelatedUnitID	int	The IdentificationUnitID of the organism or substrate, on which this organism is growing (= foreign key)
RelationType	nvarchar (50)	The relation of an unit to its substrate, e.g. parasitism, symbiosis etc. as stored in

		CollRelationType_Enum (= foreign key)
ColonisedSubstratePart	nvarchar (255)	If a substrate association exists: part of the substrate that is affected in the interaction (e.g. 'leaves' if a fungus is growing on the leaves of an infected plant)
LifeStage	nvarchar (255)	Examples: 'II, III' for spore generations of rusts or 'seed', 'seedling' etc. for higher plants
Gender	nvarchar (50)	The gender of the identification unit, e.g. 'male'
NumberOfUnits	smallint	The number of units of this identification unit, e.g. 400 beetle in a bottle
ExsiccataNumber	nvarchar (50)	If specimen is an exsiccata: Number of current specimen within the exsiccata series
ExsiccataIdentification	smallint	Refers to the IdentificationSequence in Identification (= foreign key). The name under which the collection specimen resp. this unit is published within an exsiccate.
UnitIdentifier	nvarchar (50)	An identifier for the identification of the unit e.g. a number painted on a tree within an experimental plot
UnitDescription	nvarchar (50)	Description of the unit, esp. if not the an organism but parts or remnants of it were present or observed, e.g. a nest of an insect or a song of a bird
Circumstances	nvarchar (50)	Circumstances of the occurence of the organism
DisplayOrder	smallint	The sequence in which the units within this specimen will appear on e.g. a label where the first unit may be printed in the header and others in the text below. 0 means the unit should not appear on a label. *Default value: (1)*
Notes	nvarchar (MAX)	Further information on the identification unit or interaction, e. g. infection symptoms like 'producing galls'
LogCreatedWhen	datetime	The time when this dataset was created *Default value: getdate()*
LogCreatedBy	nvarchar (50)	Who created this dataset *Default value: user_name()*

LogUpdatedWhen	datetime	The last time when this dataset was updated *Default value: getdate()*
LogUpdatedBy	nvarchar (50)	Who was the last to update this dataset *Default value: user_name()*

Table IdentificationUnitAnalysis

The analysis values taken from an identification unit

Column	Data type	Description
CollectionSpecimenID	int	Refers to the ID of CollectionSpecimen (= Foreign key and part of primary key)
IdentificationUnitID	int	Refers to the ID of IdentficationUnit (= foreign key and part of primary key)
AnalysisID	int	Analysis ID, foreign key of table Analysis.
AnalysisNumber	nvarchar (50)	Number of the analysis
AnalysisResult	nvarchar (MAX)	The result of the analysis
ExternalAnalysisURI	varchar (255)	An URI for an analysis as defined in an external datasoure
ResponsibleName	nvarchar (255)	Name of the person or institution responsible for the determination *Default value: [dbo].[CurrentUserName]()*
ResponsibleAgentURI	varchar (255)	URI of the person or institution responsible for the determination (= foreign key) as stored in the module DiversityAgents.
AnalysisDate	datetime	The date of the analysis
SpecimenPartID	int	ID of the part of a specimen (optional, Foreign key) if the analysis was done with a part of the specimen (see table CollectionSpecimenPart).
Notes	nvarchar (MAX)	Notes concerning this analysis
LogCreatedWhen	datetime	The time when this dataset was created *Default value: getdate()*
LogCreatedBy	nvarchar (50)	Who created this dataset *Default value: user_name()*
LogUpdatedWhen	datetime	The last time when this dataset was updated *Default value: getdate()*
LogUpdatedBy	nvarchar (50)	Who was the last to update this dataset

		Default value: user_name()

Table IdentificationUnitInPart

The list of the organisms that are found in a part of the specimen

Column	Data type	Description
CollectionSpecimenID	int	Refers to the ID of CollectionSpecimen (= Foreign key and part of primary key)
IdentificationUnitID	int	ID of the identification unit (= part of Primary key). Usually one of possibly several organisms present on the collection specimen. Example: parasite with hyperparasite on plant leaf = 3 units,
SpecimenPartID	int	ID of the part of a specimen (optional, Foreign key) if the identification unit is located on a part of the specimen (see table CollectionSpecimenPart).
DisplayOrder	smallint	The sequence in which the units within this part will appear on e.g. a label where the first unit may be printed in the header and others in the text below. 0 means the unit should not appear on a label. *Default value: (1)*
Description	nvarchar (500)	A description of the unit, esp. if not a whole unit but e.g. parts of it are stored in the collection, e.g. a nest of a bird
LogInsertedBy	nvarchar (50)	Name of user who first entered (typed or imported) the data. *Default value: user_name()*
LogInsertedWhen	smalldatetime	Date and time when the data were first entered (typed or imported) into this database. *Default value: getdate()*
LogUpdatedBy	nvarchar (50)	Name of user who last updated the data. *Default value: user_name()*
LogUpdatedWhen	smalldatetime	Date and time when the data were last updated. *Default value: getdate()*

Table LocalisationSystem

The geographic localisation systems, e.g. coordinates

Column	Data type	Description
LocalisationSystemID	int	Unique ID for the localisation system (= Primary key)
LocalisationSystemParentID	int	LocalisationSystemID of the superior LocalisationSystem
LocalisationSystemName	nvarchar (100)	Name of the system used for the determination of the place of the collection, e. g. Gauss-Krüger, MTB, GIS
DefaultAccuracyOfLocalisation	nvarchar (50)	The default for the accuracy of values that can be reached with this method
DefaultMeasurementUnit	nvarchar (50)	The default measurement unit for the localisation system, e.g. m, geograpic coordinates
ParsingMethodName	nvarchar (50)	Internal value, specifying a programming method used for parsing text in fields Location1/Location2 in table CollectionLocalisation
DisplayText	nvarchar (50)	Short abbreviated description of the localisation system as displayed in the user interface
DisplayEnable	bit	Specifies if this item is enabled to be used within the database. LocalisationSystems can be disabled to avoid seeing them, but to keep the definition for the future.
DisplayOrder	smallint	The order in which the entries are displayed. The order may be changed at any time, but all values must be unique.
Description	nvarchar (255)	Description of the localisation method
DisplayTextLocation1	nvarchar (50)	Short abbreviated description of the attribute Location1 in the table CollectionGeography as displayed in the user interface
DescriptionLocation1	nvarchar (255)	Description of the attribute Location1 in the table CollectionGeography as displayed in the user interface
DisplayTextLocation2	nvarchar (50)	Short abbreviated description of the attribute Location2 in the table CollectionGeography as displayed in the user interface

DescriptionLocation2	nvarchar (255)	Description of the attribute Location2 in the table CollectionGeography as displayed in the user interface

Table Processing

The processings of the specimen

Column	Data type	Description
ProcessingID	int	ID of the processing (Primary key)
ProcessingParentID	int	The ID of the superior type of the processing
DisplayText	nvarchar (50)	The display text of the processing as shown e.g. in a user interface
Description	nvarchar (MAX)	Description of the processing
Notes	nvarchar (MAX)	Notes about the processing
ProcessingURI	varchar (255)	An URI for a processing as defined in an external datasource
LogCreatedWhen	datetime	The time when this dataset was created *Default value: getdate()*
LogCreatedBy	nvarchar (50)	Who created this dataset *Default value: user_name()*
LogUpdatedWhen	datetime	The last time when this dataset was updated *Default value: getdate()*
LogUpdatedBy	nvarchar (50)	Who was the last to update this dataset *Default value: user_name()*

Table ProcessingMaterialCategory

The processings that are possible for a certain material category

Column	Data type	Description
ProcessingID	int	ID of the processing. Refers to ProcessingID in table Processing (foreign key) *Default value: (1)*
MaterialCategory	nvarchar (50)	Material category of specimen. Examples: 'herbarium sheets', 'drawings', 'microscopic slides' etc. *Default value: N'specimen'*
LogUpdatedWhen	datetime	The last time when this dataset was updated

		Default value: getdate()
LogUpdatedBy	nvarchar (50)	Who was the last to update this dataset *Default value: user_name()*

Table Property

The list of the properties that can be specified for the collection site

Column	Data type	Description
PropertyID	int	Unique ID for the localisation system (= Primary key)
PropertyParentID	int	LocalisationSystemID of the superior LocalisationSystem
PropertyName	nvarchar (100)	Name of the system used for the determination of the place of the collection, e. g. Gauss-Krüger, MTB, GIS
DefaultAccuracyOfProperty	nvarchar (50)	The default for the accuracy of values that can be reached with this method
DefaultMeasurementUnit	nvarchar (50)	-
ParsingMethodName	nvarchar (50)	Internal value, specifying a programming method used for parsing text in fields Location1/Location2 in table CollectionLocalisation
DisplayText	nvarchar (50)	Short abbreviated description of the localisation system as displayed in the user interface
DisplayEnabled	bit	Specifies if this item is enabled to be used within the database. LocalisationSystems can be disabled to avoid seeing them, but to keep the definition for the future.
DisplayOrder	smallint	The order in which the entries are displayed. The order may be changed at any time, but all values must be unique.
Description	nvarchar (255)	Description of the localisation method

Table Transaction

Transactions like loan, borrow, gift, exchange etc. of specimen if they are e.g. permanently or temporary transfered from one collection to another

Column	Data type	Description

TransactionID	int	Unique ID for the transaction (= Primary key)
ParentTransactionID	int	The ID of a preceeding transaction of a superior transaction if transactions are organized in a hierarchy
TransactionType	nvarchar (50)	Type of the transaction e.g. gift in or out, exchange in or out, purchase in or out *Default value: N'exchange'*
TransactionTitle	nvarchar (200)	The title of the transaction as e.g. shown in an user interface
ReportingCategory	nvarchar (50)	A group defined for the transaction, e. g. a taxonomic group as used for exchange balancing
AdministratingCollectionID	int	ID of the collection thas is responsible for the administration of the transaction.
MaterialDescription	nvarchar (MAX)	ID of the project to which the transaction belongs (Projects are defined in DiversityProjects) *Default value: ''*
MaterialCategory	nvarchar (50)	Material category of specimen. Examples: 'herbarium sheets', 'drawings', 'microscopic slides' etc. *Default value: N'specimen'*
MaterialCollectors	nvarchar (MAX)	The collectors of the material
FromCollectionID	int	The ID of the collection from which the specimen were transfered, e.g. the donating collection of a gift
FromTransactionPartnerName	nvarchar (255)	Name of the person or institution from which the specimen were transfered, e.g. the donator of a gift
FromTransactionPartnerAgentURI	varchar (255)	The URI of the transaction partner (see e.g. module DiversityAgents)
FromTransactionNumber	nvarchar (50)	Number or code by which a transaction may be recorded by the administration of the source of the specimen, e.g. the donating collection of a gift
ToCollectionID	int	The ID of the collection to which the specimen were transfered, e.g. the receiver of a gift
ToTransactionPartnerName	nvarchar (255)	Name of the person or

		institution to which the specimen were transfered, e.g. the receiver of a gift
ToTransactionPartnerAgentURI	varchar (255)	The URI of the transaction partner (see e.g. module DiversityAgents)
ToTransactionNumber	nvarchar (50)	Number or code by which a transaction may be recorded by the administration of the destination of the specimen, e.g. the receiving collection of a gift
NumberOfUnits	smallint	The number of units that were (initially) included in the transaction
Investigator	nvarchar (50)	The investigator for whose study a transacted material was sent
TransactionComment	nvarchar (MAX)	Comments about the exchanged material addressed to the transaction partner
BeginDate	datetime	Date when the transaction started
AgreedEndDate	datetime	End of the transaction period, e.g. if the time for borrowing the specimen is restricted
ActualEndDate	datetime	Actual end of the transaction when e.g. the borrowed specimen were returned to the owner
InternalNotes	nvarchar (MAX)	Internal notes about this transaction, not to be published e.g. on a web page
ResponsibleName	nvarchar (255)	The person responsible for this transaction
ResponsibleAgentURI	varchar (255)	The URI of the person, team or organisation responsible for the data (see e.g. module DiversityAgents)
LogCreatedWhen	datetime	The time when this dataset was created *Default value: getdate()*
LogCreatedBy	nvarchar (50)	Who created this dataset *Default value: user_name()*
LogUpdatedWhen	datetime	The last time when this dataset was updated *Default value: getdate()*
LogUpdatedBy	nvarchar (50)	Who was the last to update this dataset *Default value: user_name()*

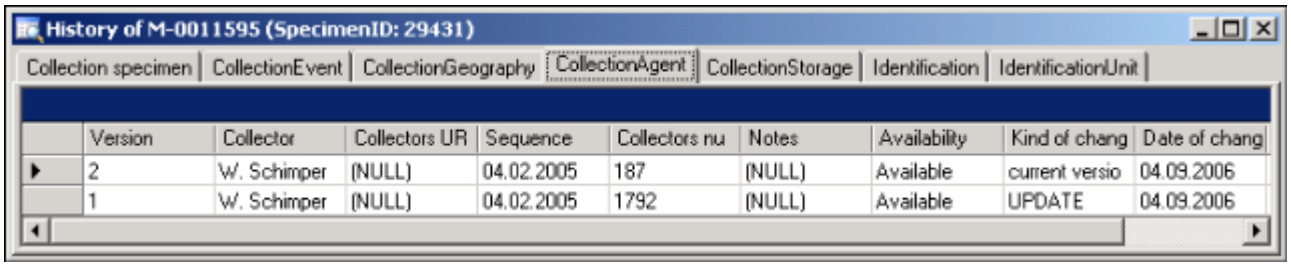
Table TransactionDocument

The history of transactions resp. the documents connected to the transactions

Column	Data type	Description
TransactionID	int	Unique ID for the Transaction, refers to table Transaction (= Part of primary key and foreign key)
Date	datetime	The date of the event of a transaction
TransactionText	nvarchar (MAX)	The text of a transaction document
TransactionDocument	image (2147483647)	A scanned document connected to this transaction event
InternalNotes	nvarchar (MAX)	Internal notes about this transaction event
LogCreatedWhen	datetime	The time when this dataset was created *Default value: getdate()*
LogCreatedBy	nvarchar (50)	Who created this dataset *Default value: user_name()*
LogUpdatedWhen	datetime	The last time when this dataset was updated *Default value: getdate()*
LogUpdatedBy	nvarchar (50)	Who was the last to update this dataset *Default value: user_name()*

History

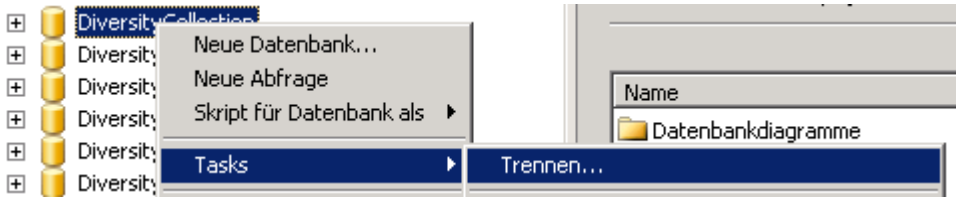
To inspect the history of a dataset click on the button. A form will open, showing all former states of the data in the tables with the current dataset at the top. The database DiversityCollection handles 2 different histories - one for the collection specimen and one for the collection event. The version shown in the header of the main form refers to these histories. The first number refers to the version of the specimen. If an event was defined a second number is shown, referring to the version of the collection event, e.g. 2 / 1 means version 2 of the specimen and version 1 of the collection event.



The version will be set automatically. If a dataset is changed the version will be increased if the last changes were done by a different user or the last change is more than 24 hours ago (for further details see topic Logging).

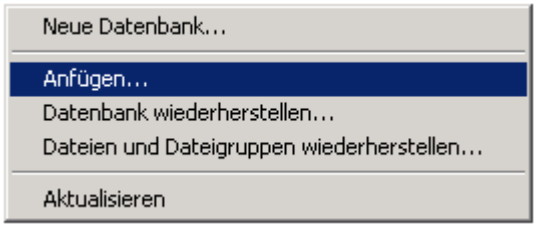
Backup

If you need to backup your database, you have to use the functionality provided by SQL-Server. To do this, you need administration rights in the database you want to create a backup. Open the Enterprise Manager for SQL-Server, choose the database and detach it from the server as shown in the image below.



After detaching the database, you can store a copy of the ..._Data.MDF File to keep it as a backup.

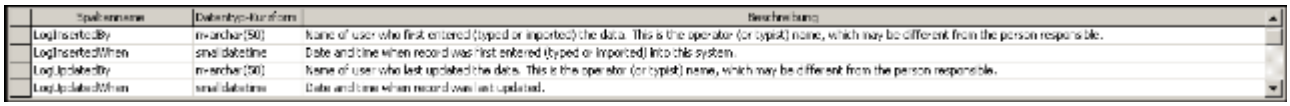
After storing the backup you have to attach the database.



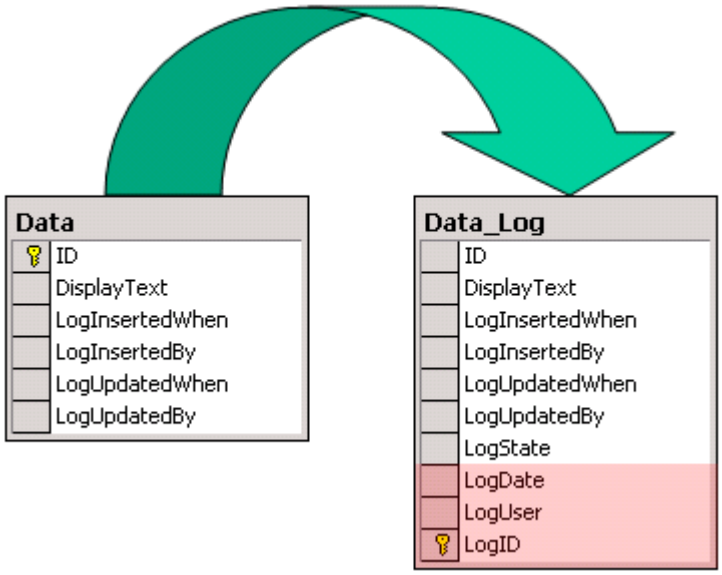
A dialog will appear where you have to select the original database file in your directory.

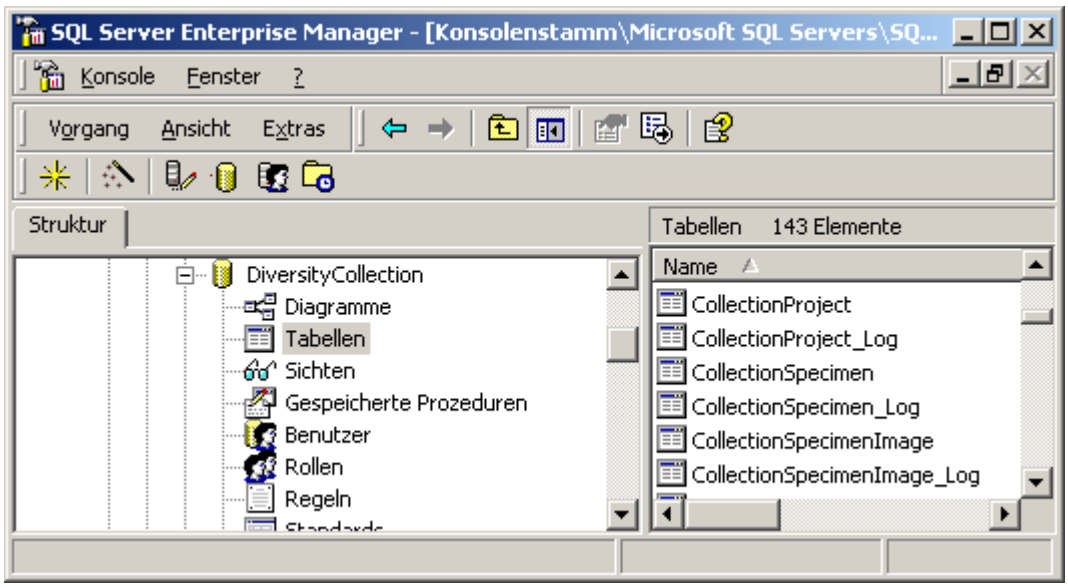
Logging

Changes within the database will be documented for each dataset together with the time and the responsible user in the columns shown in the image below.



All main tables have a corresponding logging table. If you change or delete a dataset the orignial dataset will be stored in this logging table together with informations about who has done the changes and when it happend.





Version of datasets

The **Version** of a dataset consists of two parts, e.g. 3 / 1 as shown in the header of the specimen in the main form. The first part of the number refers to the version of the dataset as stored in table CollectionSpecimen. The second part refers to the version as stored in the table CollectionEvent. Both versions will be set to a higher number if the data in the tables themselves or in dependent tables are changed, e.g. the insertion of a new identification in the table Identification will increase the specimen part of the version from 3 to 4 resulting in a version 4 / 1 for the whole dataset. The changes in the version will only occur if the last changes in the data were more than **24 hours** ago or a **different user** is changing the data. This ensures, that a user can change several parts in a dataset within 24 hours and the version will only be increased by 1.

The data are stored in the tables CollectionSpecimen and CollectionEvent.

Database - Application tables

Informations concerning the application are stored in the application tables. The table ApplicationSearchSelectionStrings hold the SQL-strings of the predefined queries for the users. The table ApplicationEntityDescription holds the description of the database entities, if these are different from the description in the database, especially if other languages then English are to be supported.

Index

- ApplicationEntityDescription
- ApplicationSearchSelectionStrings
- ProjectProxy
- ProjectUser
- UserProxy

Table ApplicationEntityDescription

The description of the columns in different languages

Column	Data type	Description
TableName	varchar (50)	The name of the table within the database
ColumnName	varchar (50)	The name of the column of the table within the database
DisplayText	nvarchar (50)	The text for the column as shown e.g. in a user interface
Description	nvarchar (MAX)	The description of the content of the column
LanguageCode	nvarchar (50)	ISO 639: 2-letter codes for the language of DisplayText and Description
LogCreatedWhen	datetime	The time when this dataset was created *Default value: getdate()*
LogCreatedBy	nvarchar (50)	Who created this dataset *Default value: user_name()*
LogUpdatedWhen	datetime	The last time when this dataset was updated *Default value: getdate()*
LogUpdatedBy	nvarchar (50)	Who was the last to update this dataset *Default value: user_name()*

Table ApplicationSearchSelectionStrings

Selection strings for customized queries for users

Column	Data type	Description
UserName	varchar (50)	The name of the user who created this SQL string *Default value: user_name()*
SQLStringIdentifier	varchar (50)	The identifier for the selection string as shown in user interface

ItemTable	varchar (50)	The main table from which the datasets should be selected *Default value: 'TaxonName'*
SQLString	varchar (MAX)	SQL string for selecting datasets from the database
Description	nvarchar (MAX)	Description of the resultset and the purpose of the query
LogCreatedWhen	datetime	The time when this dataset was created *Default value: getdate()*
LogCreatedBy	nvarchar (50)	Who created this dataset *Default value: user_name()*
LogUpdatedWhen	datetime	The last time when this dataset was updated *Default value: getdate()*
LogUpdatedBy	nvarchar (50)	Who was the last to update this dataset *Default value: user_name()*

Table ProjectProxy

The projects as stored in the module DiversityProjects

Column	Data type	Description
ProjectID	int	ID of the project to which the specimen belongs (Projects are defined in DiversityProjects)
Project	nvarchar (50)	The name or title of the project as shown in a user interface (Projects are defined in DiversityProjects)

Table ProjectUser

The projects that a user can access

Column	Data type	Description
LoginName	nvarchar (50)	A login name which the user uses for access the DivesityWorkbench, Microsoft domains, etc..
ProjectID	int	ID of the project to which the specimen belongs (Projects are defined in DiversityProjects)

Table UserProxy

The user as stored in the module DiversityUsers

Column	Data type	Description
LoginName	nvarchar (50)	A login name which the user uses for access the DivesityWorkbench, Microsoft domains, etc..
CombinedNameCache	nvarchar (255)	The short name of the user, e.g. P. Smith *Default value: NULL*
UserURI	varchar (255)	URI of a user in a remote module, e.g. refering to UserInfo.UserID in database DiversityUsers

Diversity Workbench

The Diversity Workbench is composed of components for building and managing biodiversity information, each of which focuses on a particular domain. Each component can provide services to the other components. DiversityCollection can link data to the modules DiversityProjects, DiversityResources, DiversityExsiccatae, DiversityGazetteer, DiversityScientificTerms, DiversityTaxonNames, DiversityAgents, DiversityUsers and DiversityReferences as illustrated in the image below.

The modules communicate with each other to provide their services for the other modules.

Modules

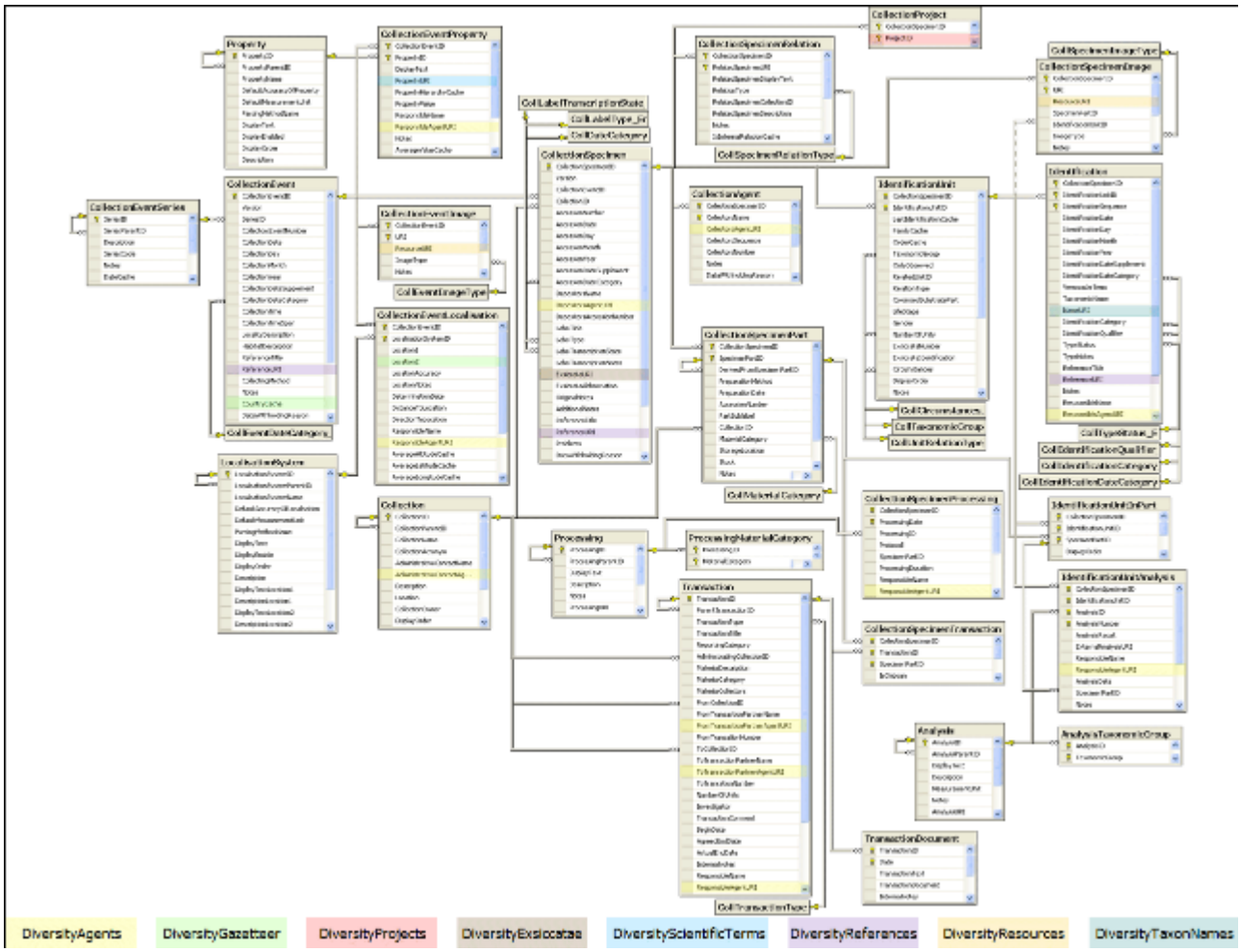
The Diversity Workbench is a set of components for building and managing biodiversity information, each of which focuses on a particular domain.

Diversity Agents	Administration of the agents, i.e. people and institutions which should be documented with e.g. their addresses
Diversity Collection	Administration of the scientific collections and specimens within these collections
Diversity Descriptions	Administration of descriptive data
Diversity Exsiccatae	Administration of exsiccatal series
Diversity Gazetteer	A data collection to enable the linking of geographical records with the Getty Thesaurus of Geographical Names (TGN), the Germ geographical names.
Diversity ScientificTerms	Data collections of scientific terms from foreign sources like vegetation, stratigraphy, soil science etc.
Diversity Projects	Administration of projects within the Diversity Workbench
Diversity References	Administration of references
Diversity Resources	Administration of resources like images, etc.
Diversity TaxonNames	Administration of taxonomic names, their synonyms and hierarchical position
Diversity Users	Administration of the users and their permissions within the Diversity Workbench

Each module provides services for the other modules. To use the service of a module, you need access to the database of the module and optionally the module application placed in

your application directory.

In the graphic below the connections to other modules within DiversityCollection are indicated.



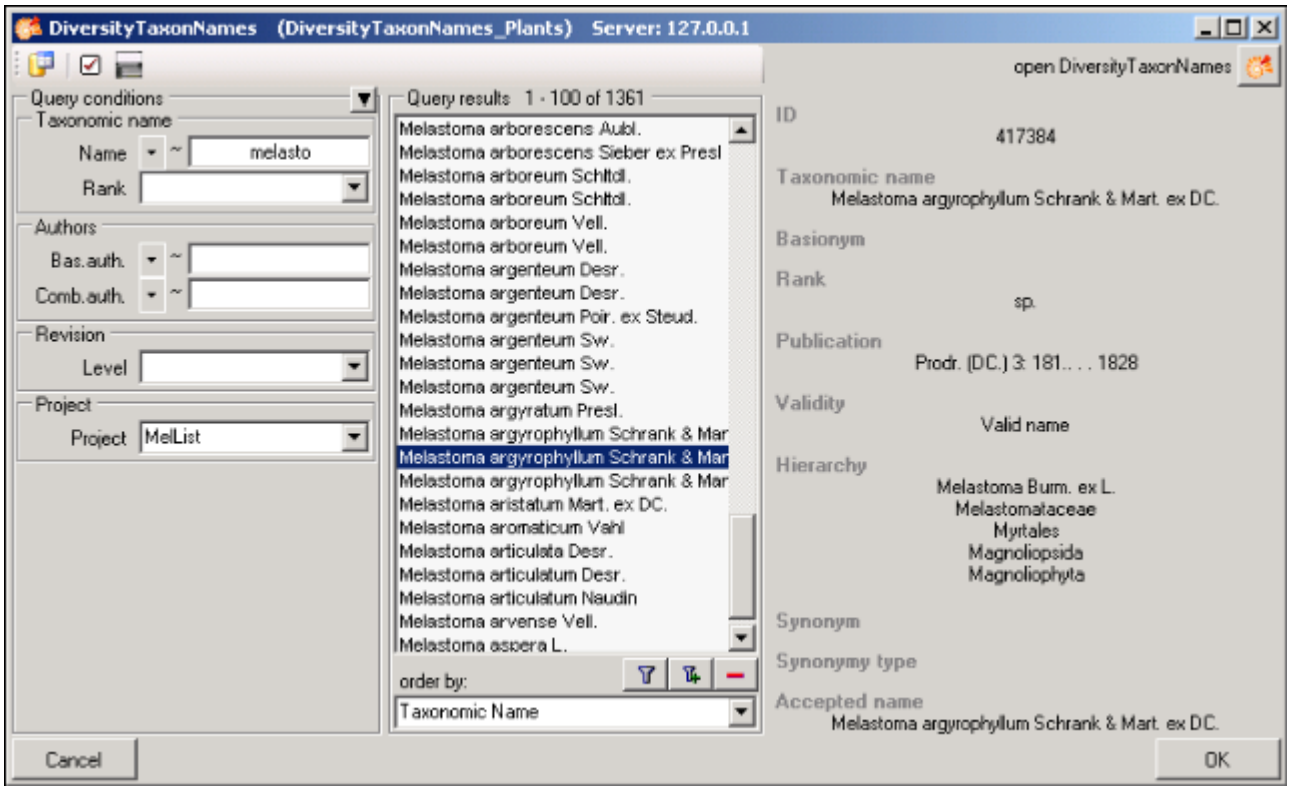
In the form a connection to a module of [Diversity Workbench] is a set of components for building and managing biodiversity information, each of which focuses on a particular domain.

Module related entry

The Diversity Workbench is a set of components for building and managing biodiversity information, each of which focuses on a particular domain. Each module provides services for the other modules. To use the service of a module, you need access to the database of the module and optionally the module application placed in your application directory. Entries related to an external module have a standard interface in the main form. There are 2 states of this interface:

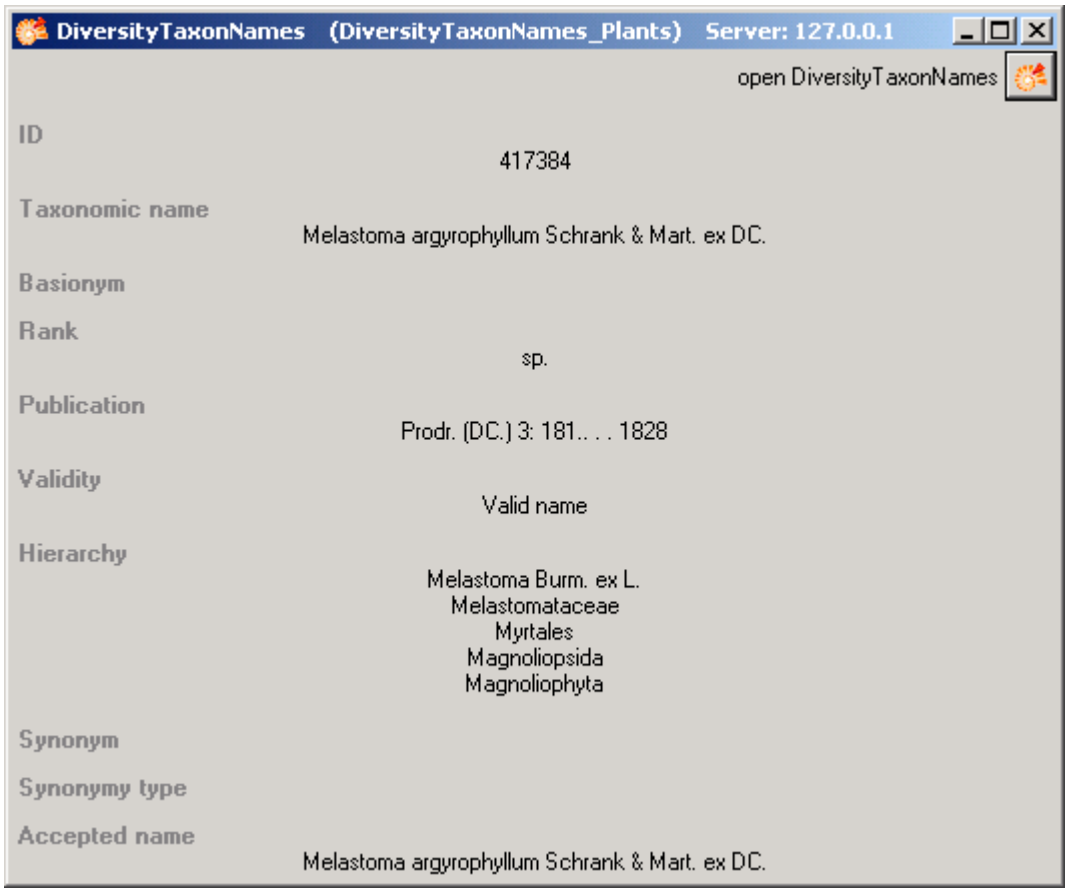
1 - the value is only set in the local database with no connection to the remote module

In this state, you can either type the value or select it from the values that are already available in the database. To get a list of the available values type the beginning of the value (you may use wildcards) and click on the button. If you want to set a relation to the remote module, click on the button. A form will open where you may select an entry from the foreign database.



2 - the value is related to the remote module

If the value has a relation to the remote module, the interface will appear as shown above. To release the connection to the remote module click the button. If you need further information about the value, click the button. This will open a form, showing an overview of the related value.



If the client application of the module is available you can inspect the details of the entry.

To start the client application of the remote module, just click on the button.

3 - relation to a webservice

Some modules provide the possibility to link your data to an external webservice. For example DiversityTaxonNames gives you access to the taxonomic names of IndexFungorum.

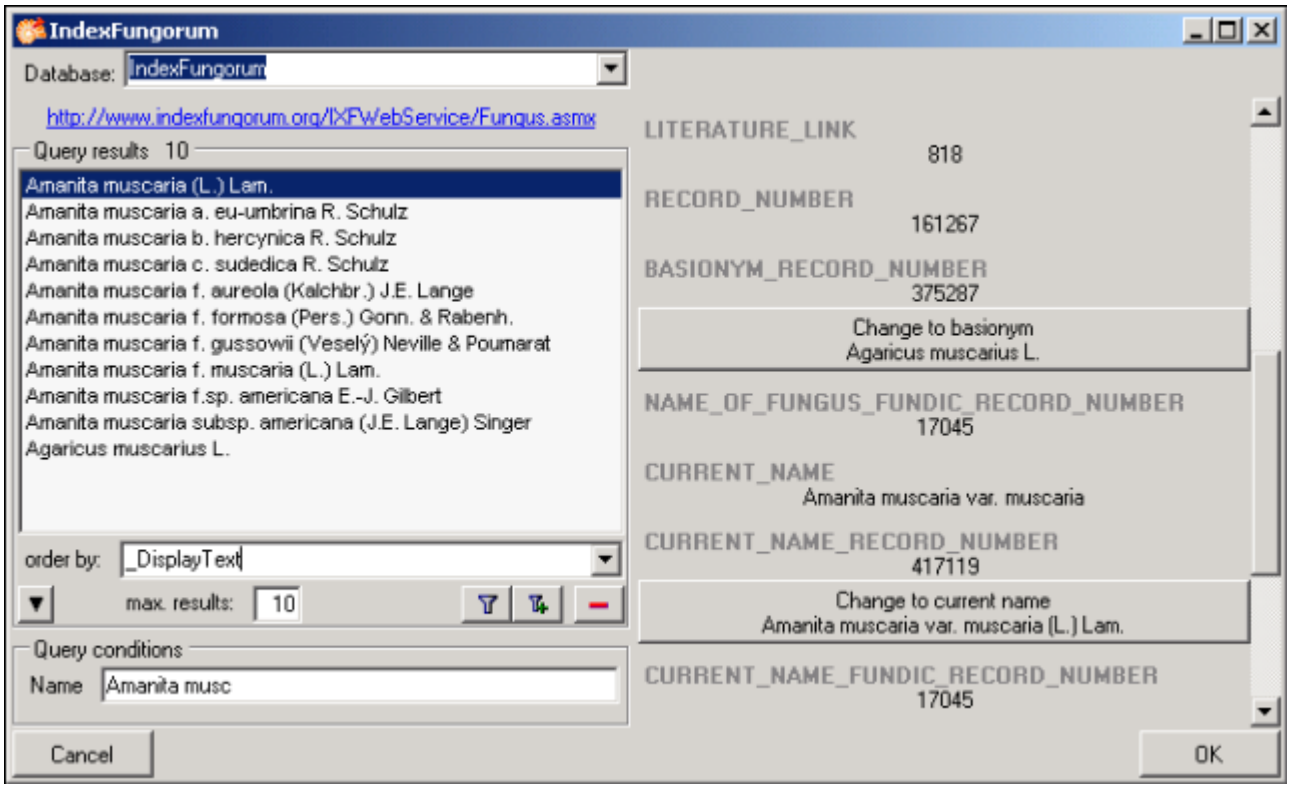
To establish a connection to an external webservice, click on the button. As with the link to modules within the Diversity Workbench a form will open where you can choose from either Diversity Workbench modules or external Webservices. See Webservice for further details

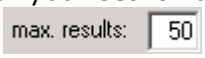
4 - relation to a local service

Some modules provide the possibility to link your data to a local installed service. For example DiversityTaxonNames gives you access to the taxonomic names of the Catalogue of life. See Catalogue of Life for further details

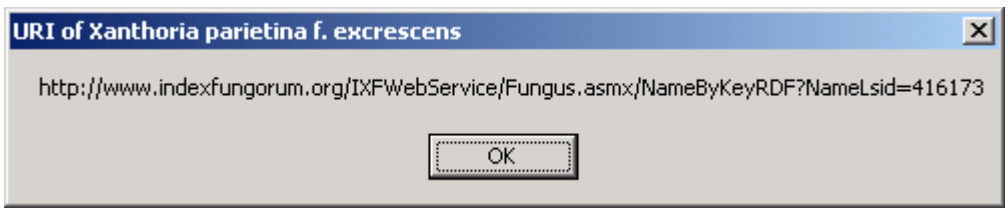
Webservice - foreign sources

Some modules within the Diversity Workbench provide the possibility to link your data to an external webservice. For example DiversityTaxonNames gives you access to the taxonomic names of IndexFungorum. To establish a connection to an external webservice, click on the button. A form will open where you can choose from either Diversity Workbench modules or external Webservices.



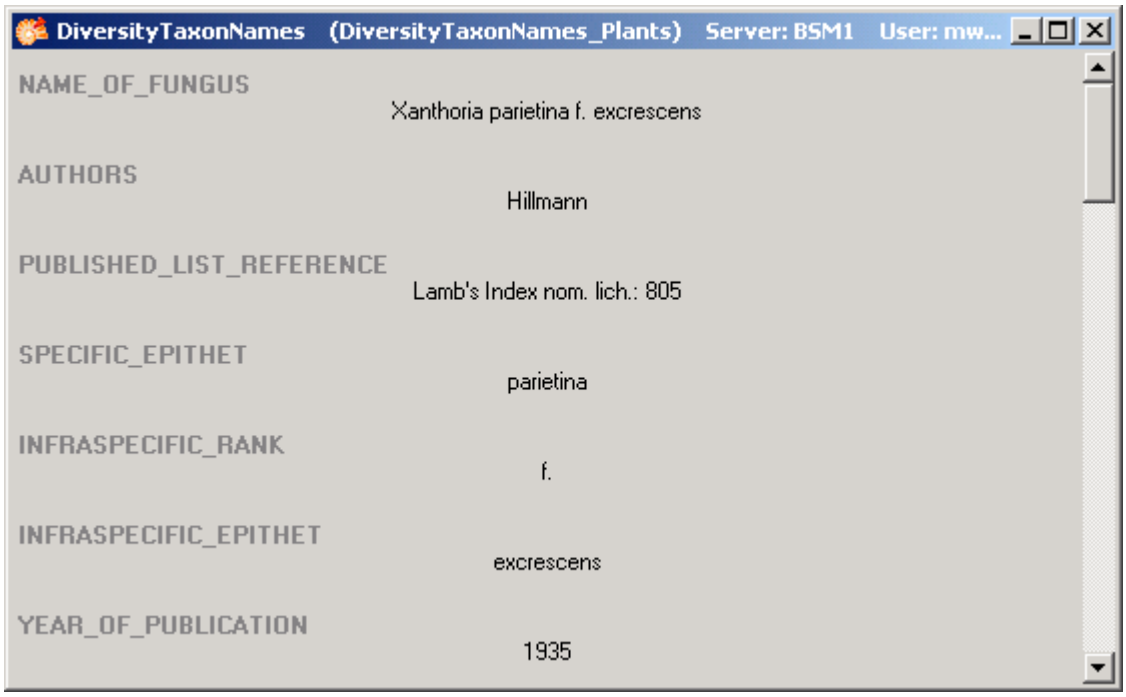
If you choose a webservice the form will change as shown in the above image. Enter the query condition for your search and the maximal number of records you want to get in the max. results field  (choose a low number if you have a slow connection to the internet). Then click on the search button to start the query. In the right part of the form additional information is shown as provided by the webservice. For certain entries buttons will appear, as e.g. shown above for the basionym and the current name of a scientific name. Click on these buttons if you want to change to one of these related datasets from the webservice. To take the link from the webservice into your database choose one of the entries and click OK. The entry will change as shown below.

If you double-click on the link area a window will open, providing you with the retrieval information of the webservice.



To get the whole information related to this entry as provided by the webservice, click on the

button. A window will open as shown below where the informations of the webservice are listed.



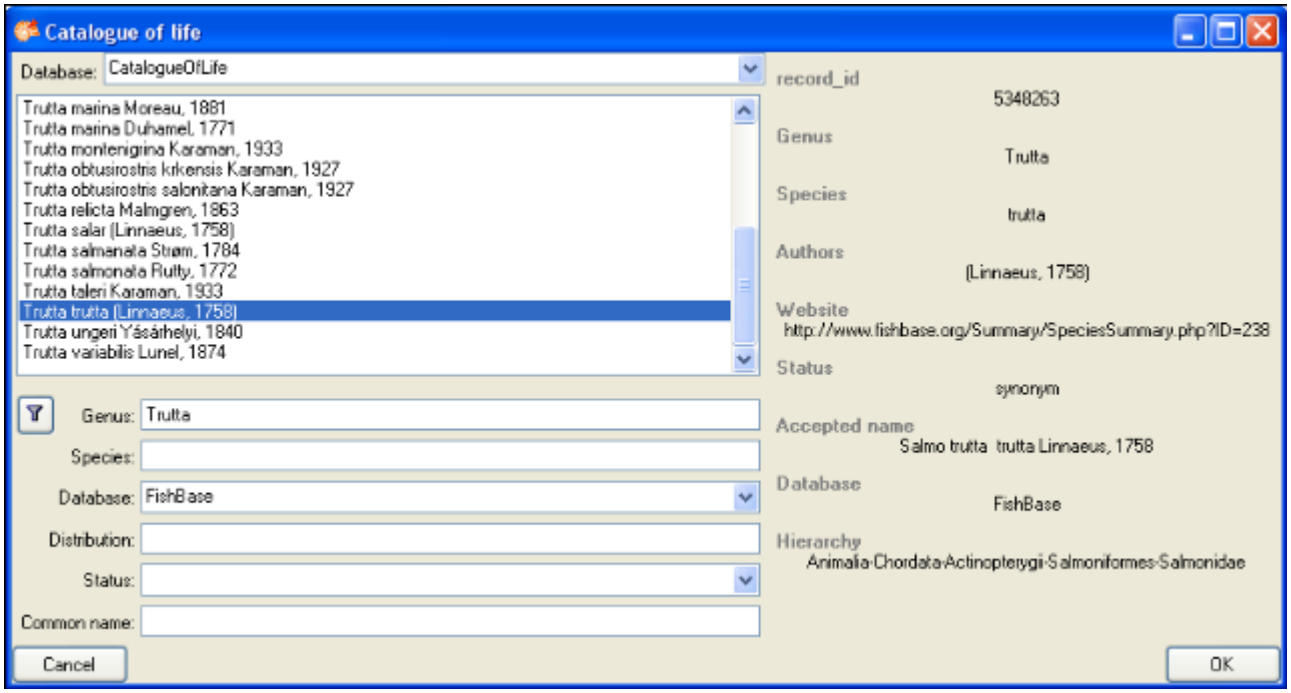
If you want to remove the link to the webservice, click on the button. This will only remove the relation to the webservice, not the cached name.

Catalogue of life

Some modules within the Diversity Workbench provide the possibility to link your data to a foreign source. For example DiversityTaxonNames gives you access to the taxonomic names of the Catalogue of life. To establish a connection to this source, click on the button. As with the link to modules within the Diversity Workbench a form will open where you can choose from either Diversity Workbench modules, external Webservices or foreign sources. If you need the service of the catalogue of life, you have to install this according to the installation instructions provided at http://www.catalogueoflife.org/annual-checklist/2008/info_about_col.php. If you have the catalogue of life installed on your local machine, you have access to this service provided the service is started. Start the service e.g. by choosing **Programs** -> **Catalogue of Life** -> **2008 Annual Checklist** from your Windows Start menu (the exact path depends on your local installation).

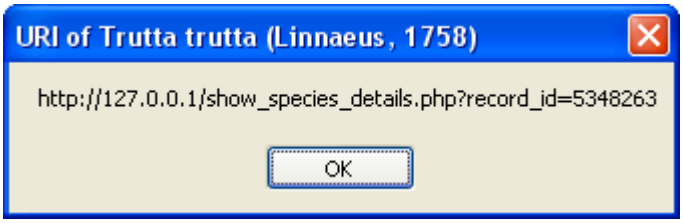
An icon will apear in your Windows Task panel, showing that the MicroWeb server is running. Than choose the Catalogue of life from the list as shown below. If you forget to start this service, the programm will try to start it. To stop this service, right-click on this symbol and choose Exit.

If the service is started you can search the names provided by the Catalogue of life as shown below.

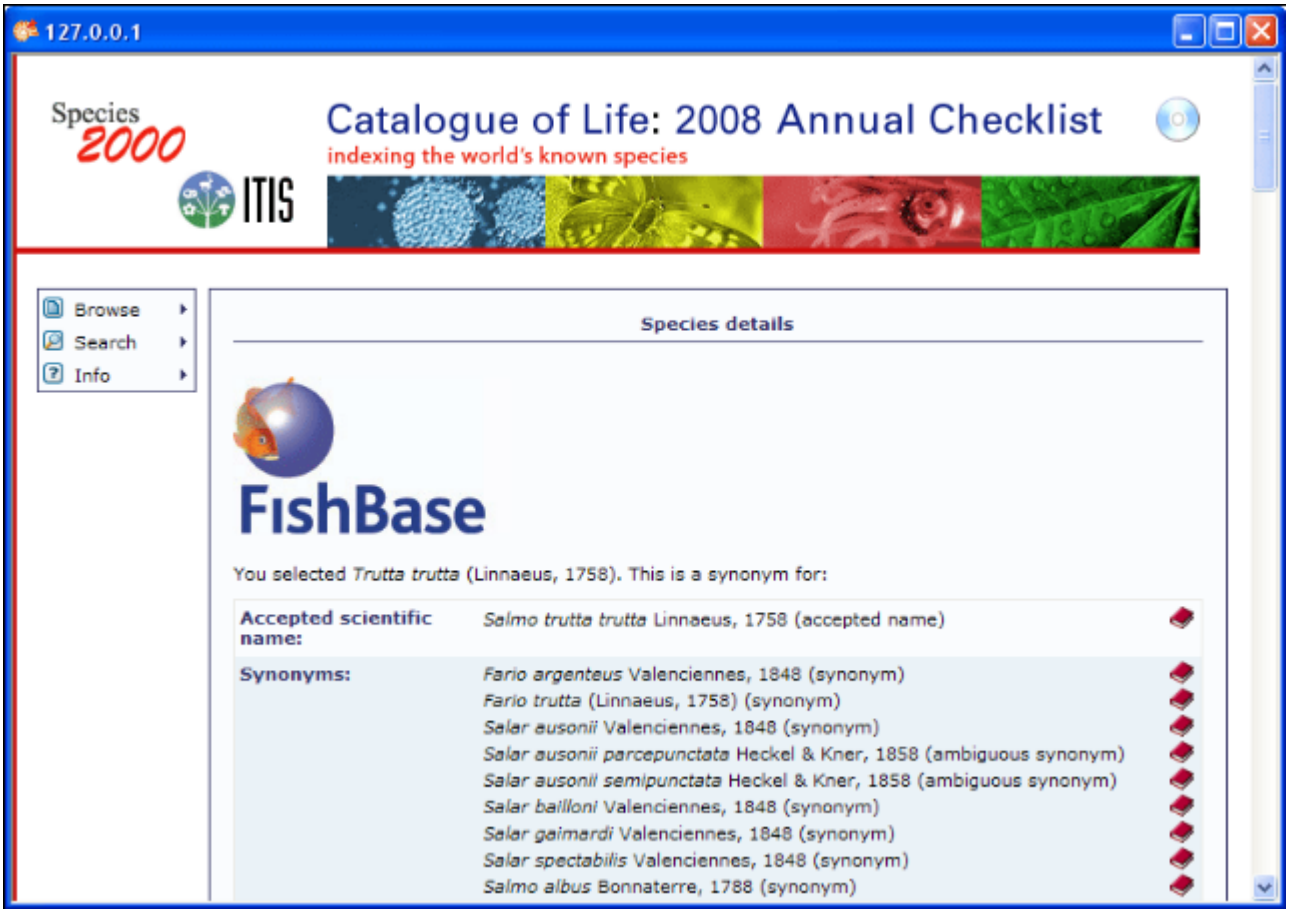


You can choose one of the databases provided by the Catalogue of Life and enter the query conditions for your search. Then click on the search button to start the query. In the right part of the form additional information is shown as provided by the service. To take the link from the service into your database choose one of the entries and click OK. The entry will change as shown below.

If you double-click on the link area a window will open, providing you with the retrieval information of the service.



To get the whole information related to this entry as provided by the service, click on the button. A window will open as shown below where the informations of the service are listed.



If you want to remove the link to the service, click on the button. This will only remove the relation to the service, not the cached name.

Resources

Images for specimen and the collection event can either be stored in DiversityCollection with their file path or in more detail in the module DiversityResources. For directly changing to DiversityResources click on the button.

For direct access to the resources in this module, you need the application **DiversityResources.exe** in your application directory, the database DiversityResources and a valid account in the database DiversityUsers.

Reference

Details about References are stored in the module DiversityReferences. You can choose one of the entries in this module from the picklist. To directly change to DiversityReferences click on .

For access to the references from other modules, you need the application **DiversityReferences.exe** in your application directory. To use the application DiversityReferences.exe you need access to the database DiversityReferences.