BEXIS 2.4.0

Security System

User Guide

Authors

Sven Thiel, Roman Gerlach

Contact

Website: <http://fusion.cs.uni-jena.de/bexis>

Email: bexis-support@uni-jena.de

Phone: +49-(0)3641-948968

Acknowledgement

The development of the BExIS 2.4.0 software would not be possible without the German Science Foundation (DFG) funding the BExIS++ project. BExIS++ is a collaboration of the Friedrich-Schiller-University Jena, Germany (Dept. of Computer Science, Dept. for Geography, Dept. of Ecology) and the Max-Planck-Institute for Biogeochemistry Jena, Germany.

Table of Contents

[Precautions 3](#_Toc393968571)

[Account 4](#_Toc393968572)

[Registration 4](#_Toc393968573)

[LogOn 4](#_Toc393968574)

[Users 6](#_Toc393968575)

[Creation 6](#_Toc393968576)

[Deletion 6](#_Toc393968577)

[Details 7](#_Toc393968578)

[Roles 10](#_Toc393968579)

[Creation 10](#_Toc393968580)

[Deletion 10](#_Toc393968581)

[Details 11](#_Toc393968582)

[Permissions 15](#_Toc393968583)

[Features 15](#_Toc393968584)

[Feature Permissions 16](#_Toc393968585)

[Data 18](#_Toc393968586)

[Data Permissions 19](#_Toc393968587)

# Precautions

**Caution! Do not delete the preset users and roles!**

*Users*. There is just a single user inside the database. The user name is “Admin” and the password is “5yy-Gul”. By default, this user has full access to the whole application.

*Roles*. Similar to *Users*, there is just one role inside the database – and the role name is “Admin”. In addition to that, the only and identical named user “Admin” belongs to that role. By default, this role grants full access to the application.

*Features.* The set of features is fixed and corresponds to the different hierarchical items of the menu bar.

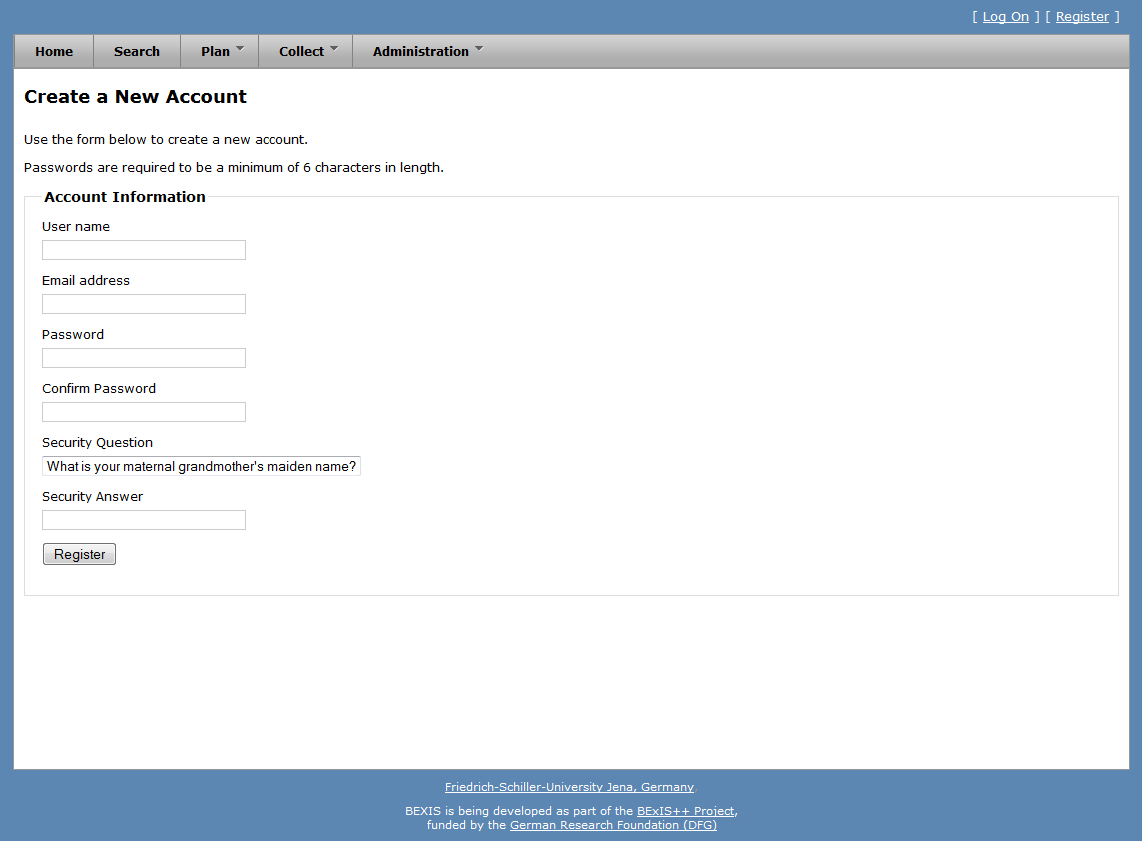
*Permissions*. Within the system, it is possible to add, remove or modify existing feature or dataset permissions.

# Account

By default, only a few components of BExIS are accessible for anonymous users. Any further access to the application is available for authorized users, only. Therefore you need to register and create an account. After a successful registration you are able to logon to BExIS using your account credentials for authentication.

## Registration

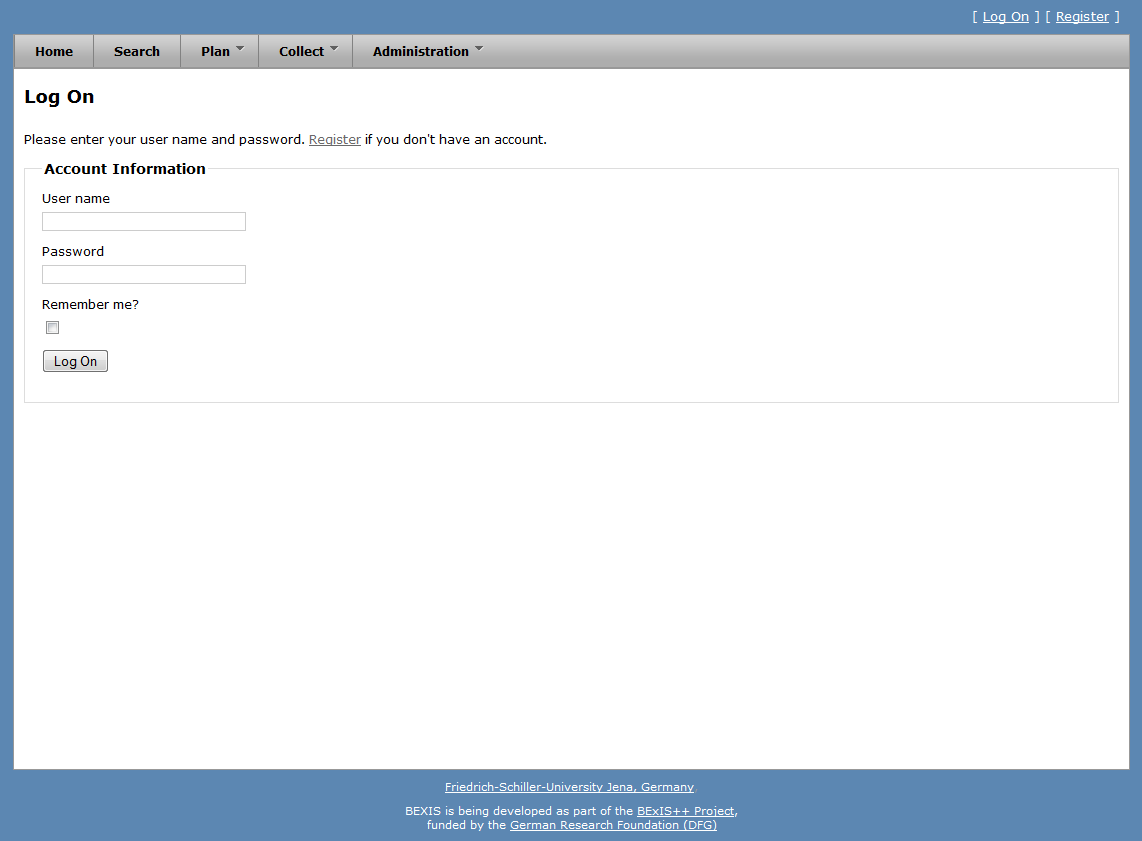
First press **[ Register ]** in the upper right corner. Figure 1 shows the registration form. You have to complete the form before you press Register, otherwise the registration will fail. Furthermore, the system supports you with different validation messages if any information is not valid.



Figure

## LogOn

First press **[ LogOn ]** in the upper right corner. The system redirects you to the logon form (see fig 1) and you have to enter your account credentials (user name and password). If the logon is successful, you will see a short welcome message instead of **[ LogOn ]**. Otherwise, the system will notify you about the status and reason why the logon was not successful.



Figure

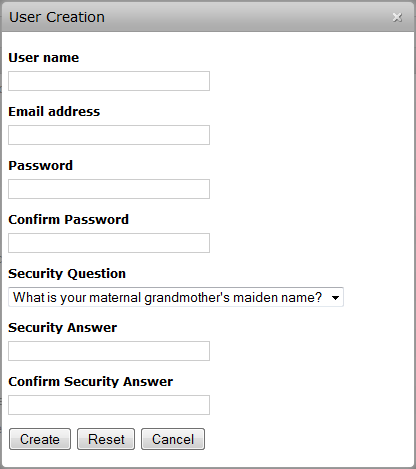
# Users

**Caution! This part of the system is secured. You may not have access to it.**

BExIS provides different features for managing users. These are typically available to system administrators only. Each of them is described in more detail in one of the following subsections.

## Creation

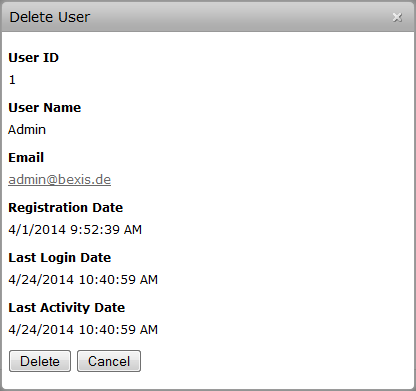
In addition to the self-registration procedure, user accounts may also be created by an administrator. This feature is available from **Administration > Authorization > Users Manager**. Please press the **+New User** button. A modal window will pop up that contains the user creation form (see fig 3). Similar to the self-registration, the system supports you with validation on all information entered.



Figure

## Deletion

Please go to **Administration > Authorization > Users Manager** and press the delete button of the corresponding user. Figure 4 shows the modal window that will pop up after pushing the button. It contains the users contact details and some information on the users activities. The deletion will be confirmed by clicking the **Delete** button.



Figure

## Details

Within BEXIS you are able to display and modify user information (see fig 5 and fig 6). For security and usability reasons, the system allows modification only for certain parts of the user information. Please go to **Administration > Authorization > Users Manager** and press the details button of the respective user. By default, the modal window only shows the user information. To enable the edit mode, please press the **Edit** button. You are now able to alter the user information. Changes are committed to the system once you pressed the **Save** button.

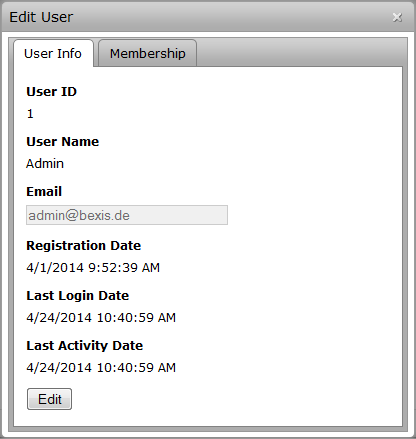
For any given user memberships to certain roles need to be specified. Please go to **Administration > Authorization > Users Manager** and press the details button of the respective user. Within the modal window please select the tab **Membership** (see fig 7). There are two different states:

user is in role 🡪 checkbox selected

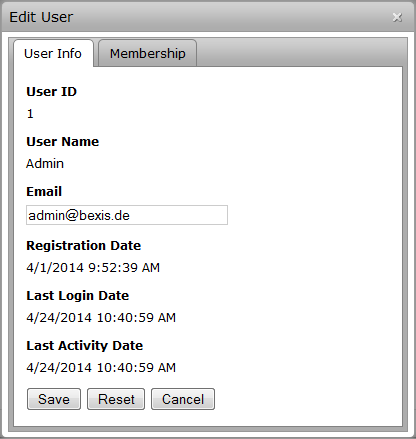
user is NOT in role 🡪 checkbox NOT selected

You can change the status easily by (un)select the corresponding checkbox.

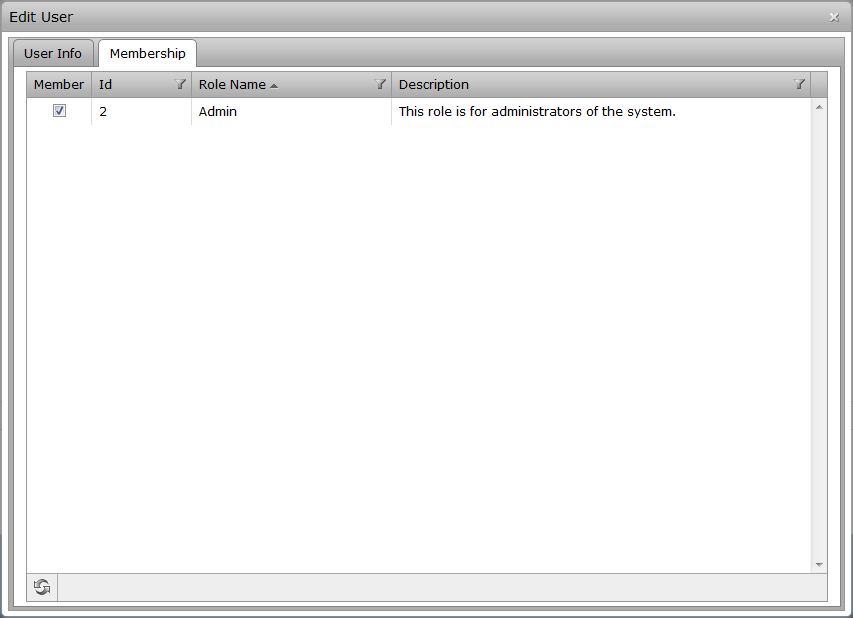
Furthermore, the system supports you for all mentioned features with different validation messages if any status or information is not valid.



Figure



Figure



Figure

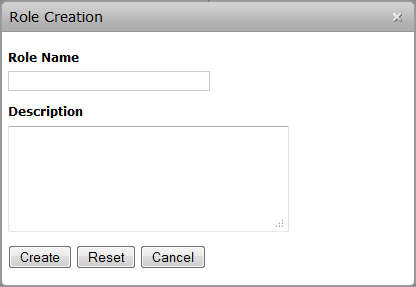
# Roles

**Caution! This part of the system is secured. You may not have access to it.**

BExIS provides different features for the managing roles. They are typically available to system administrators only. Each of them is described in more detail in the following subsections.

## Creation

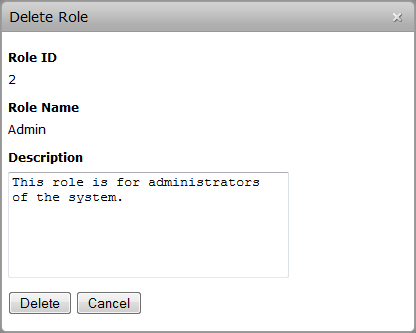
To use this feature, This feature is available from **Administration > Authorization > Roles Manager**. Please press the **+New Role** button. A modal window will pop up that contains the role creation form (see fig 8). The system supports you with validation on all information entered.



Figure

## Deletion

Please go to **Administration > Authorization > Roles Manager** and press the delete button of the corresponding role. Figure 9 shows the modal window that will pop up after pushing the button. It contains a few details about the role. The deletion will be confirmed by clicking the **Delete** button.



Figure

## Details

Within BExIS you are able to display and modify role information (see fig 10 and fig 11). Please go to **Administration > Authorization > Roles Manager** and press the details button of the respective role. By default, the modal window only shows the role information. To enable the edit mode, please press the **Edit** button. You are now able to alter the role information. Changes are committed to the system once you pressed the **Save** button.

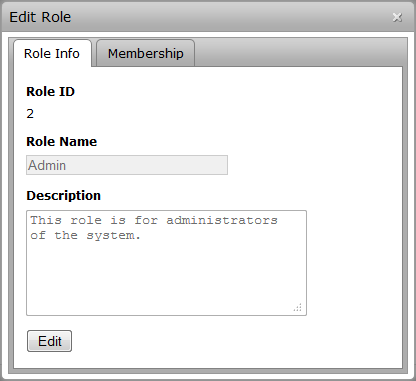
For any given role memberships of users need to be specified. Please go to **Administration > Authorization > Roles Manager** and press the details button of the respective role. Within the modal window please select the tab **Membership** (see fig 12). There are two different states:

user is in role 🡪 checkbox selected

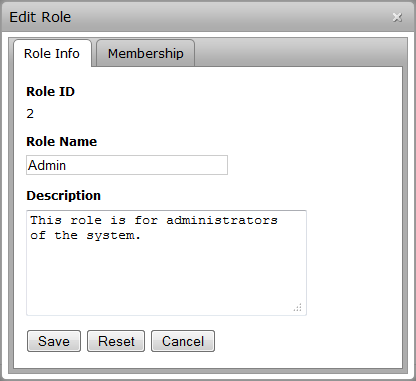
user is NOT in role 🡪 checkbox NOT selected

You can change the status easily by (un)select the corresponding checkbox.

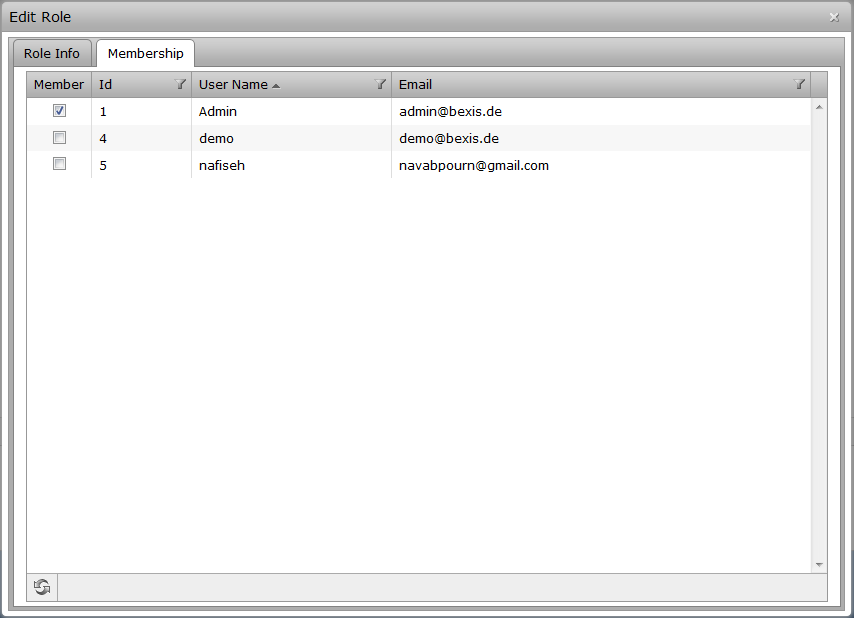
Furthermore, the system supports you for all mentioned features with different validation messages if any status or information is not valid.



Figure



Figure



Figure

# Permissions

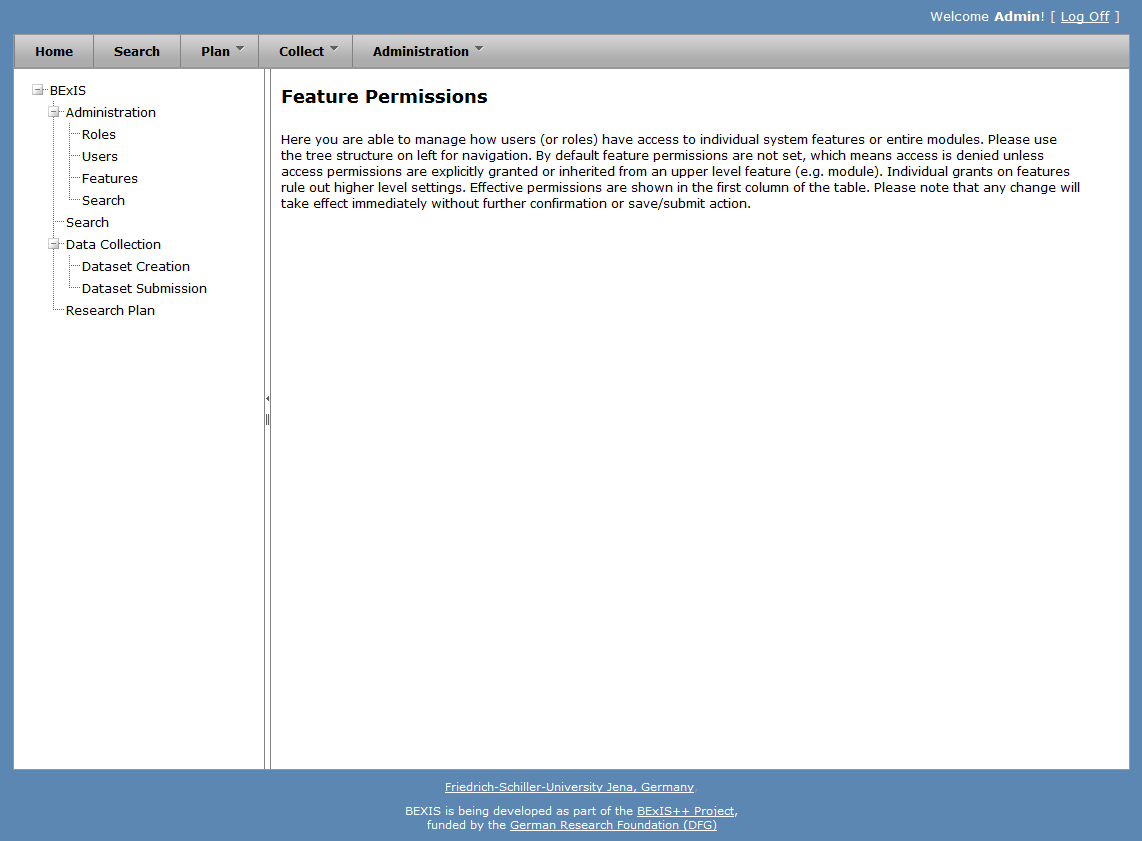
**Caution! This part of the system is secured. You may not have access to it.**

A permission is a rule that contains certain security regulations. In general, it is possible to set a rule on both, users and roles.

The security system of BExIS distinguishes between two types of permissions. On the one hand, there are feature permissions, which allow or prohibit the access to well-defined and delimited areas of the application. This type of permissions is working on functional objects (e.g. actions that should be performed) – so called *Features*. On the other hand, data permissions provide the ability to protect real data (e.g. datasets, research plans and so on). However, data permissions are not implemented yet.

## Features

Currently, features are fixed. It is not possible to add, edit or remove features. Furthermore you are not able to change well-defined and delimited area of a feature, e.g. cannot add “Search” into the feature “Administration”. The feature tree is predefined, but later on, the administrator will be able to modify features also. Nevertheless, BExIS provides a tree that shows all features and their hierarchical relationship (parent /children). Therefore, please go to **Administration > Authorization > Features Manager**. This will bring up a page with the mentioned tree on the left side (see fig 13).



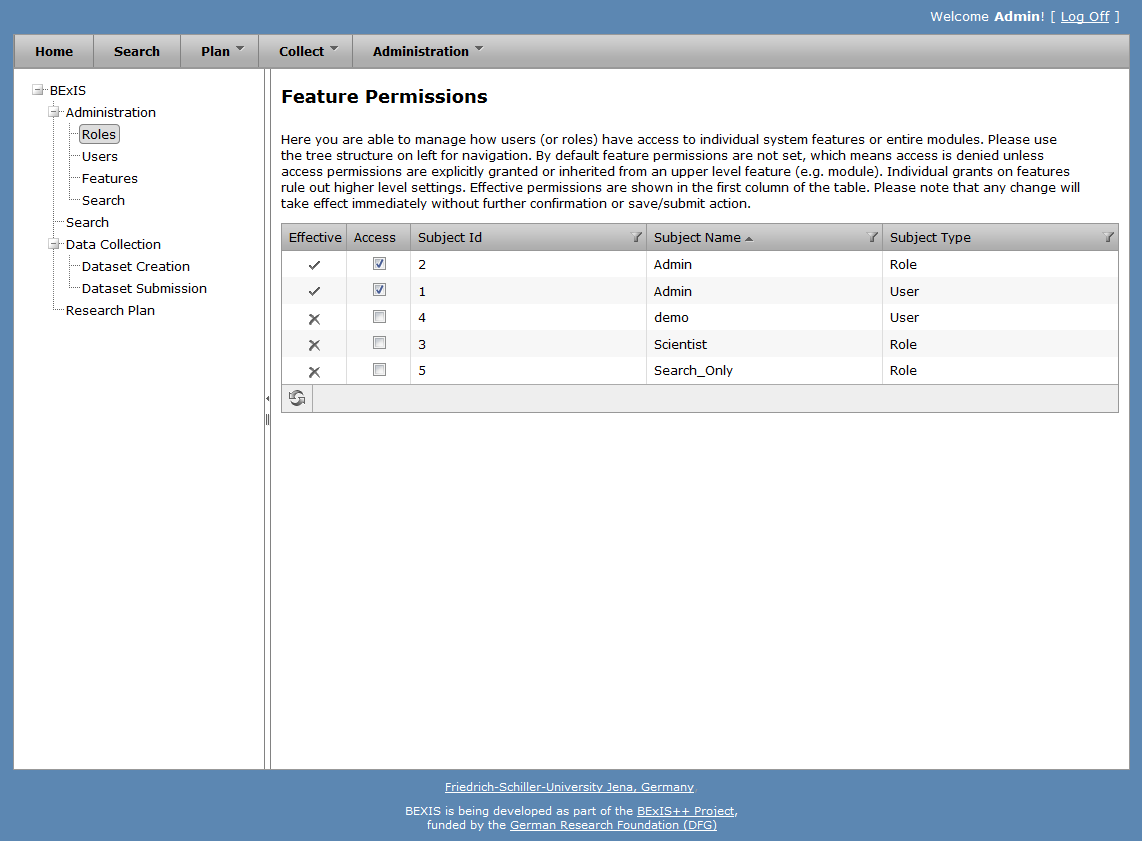
Figure

## Feature Permissions

BExIS provides an easy-to-use and user-friendly interface to manage feature permissions. First of all, please go to **Administration > Authorization > Features Manager.** This will bring up a page with a tree of all features on the left side (see fig 13). By selecting a feature (one node of the tree), the system will show a table on the right side, which contains all subjects (users and roles) and their feature permission status regarding the selected feature (see fig 14). On this page, you are also able to alter all feature permissions for the selected feature.

For each user access request to the application, the security system must allow or deny the access. This decision is called *effective right* and it is calculated with the help of the security policy. In general, BExIS provides a set of different security policies from which one can be selected. Each of them contains specific rules, e.g. how to decide if there is no feature permission for a given user. Unfortunately, the current system works with a default and fixed security policy. This one uses the following rules for a user ***u*** and a requested feature ***f***, whereby ***r*** = {r1, …, rn} is the set of roles which ***u*** belongs to.

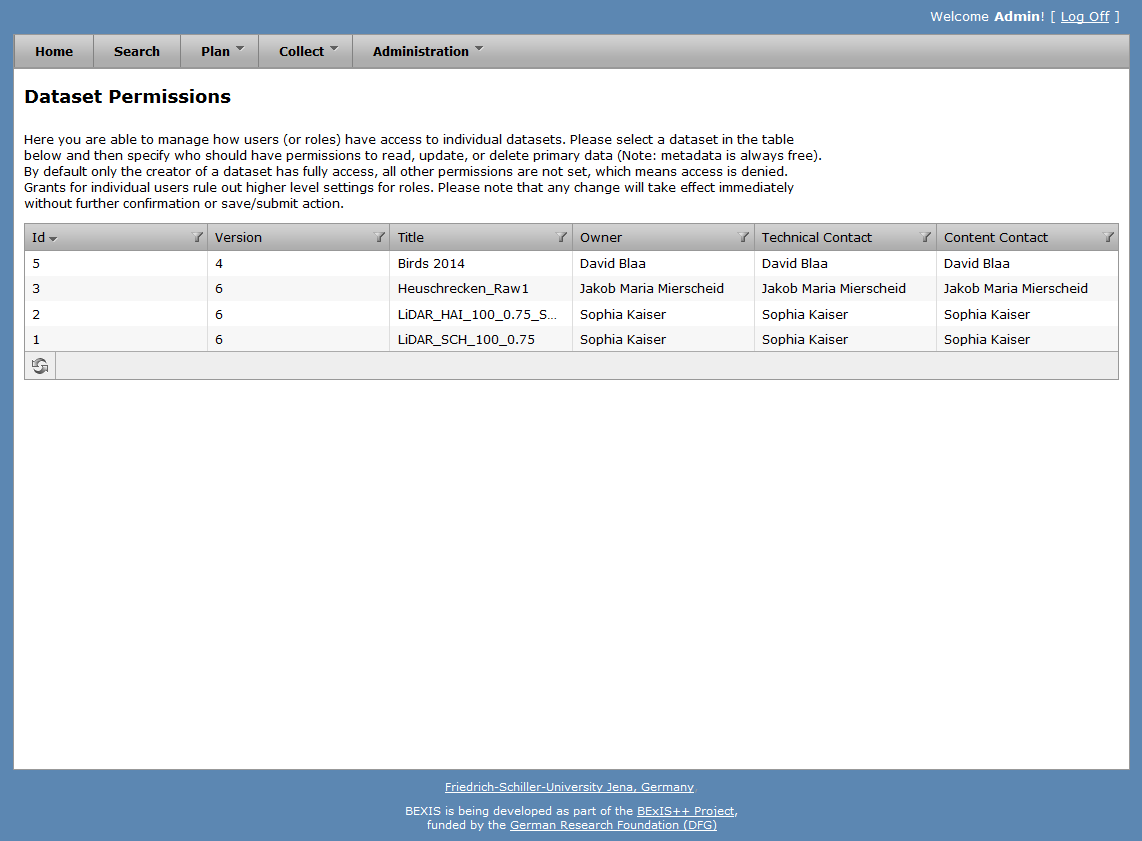
1. If there is a feature permission for user ***u*** or role ***v*** (whereby ***v*** ***r***) and feature ***f***, then return allow. Else go to step 2.
2. If ***f*** has a parent feature, then go to step 1 with ***f*** = *parent(****f****)*. Else return the default permission type (deny).



Figure

## Data

The security system of BExIS is working on both, functional (features) and non-functional (data) items. Basically each entity in BExIS can be considered as "data". But currently, it is only possible to manage the access of datasets. Please go to **Administration > Authorization > Data Manager** if you like to manage the access. This will bring up a page with the mentioned tree on the left side (see fig 15).



Figure

## Data Permissions

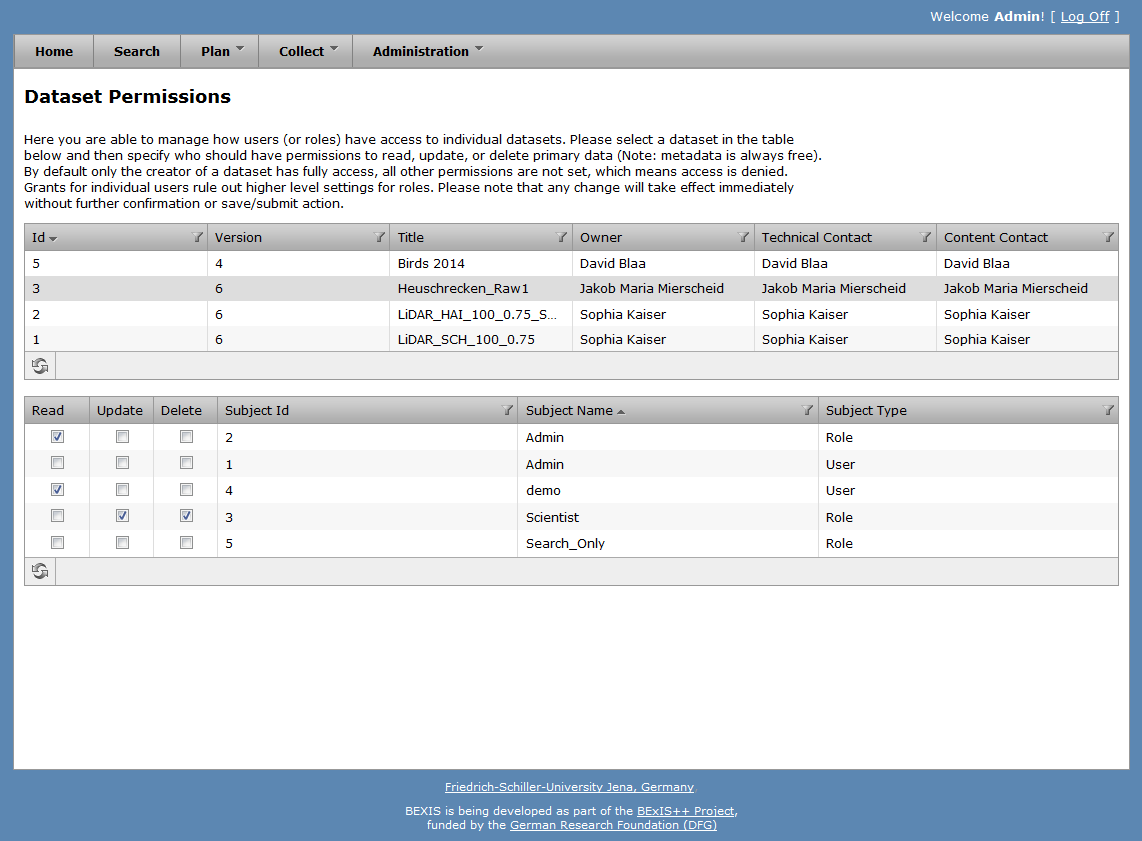
First of all, please go to **Administration > Authorization > Data Manager.** This will bring up a page with a table that contains all datasets of the system (see fig 15). By selecting a dataset, the system will show a second table underneath the first one, which contains all subjects (users and roles) and their different data permission statuses regarding the selected dataset (see fig 16). On this page, you are also able to alter all different kinds of data permissions for a selected dataset.

In general, the system is working on three different data permission types. Each of them is responsible to regulate the access for specific data. In the case of datasets, this means:

* Read: allow/deny access to primary data
* Update: allow/deny manipulation (upload and update) of primary data
* Delete: allow/deny deletion of the whole dataset

Currently, only ***read*** and ***update*** are implemented. Similar to feature permissions, the security system must allow or deny the access. This decision is called *effective right* and it is calculated with the help of the security policy. In general, BExIS provides a set of different security policies from which one can be selected. Each of them contains specific rules, e.g. how to decide if there is no data permission for a given user. Unfortunately, the current system works with a default and fixed security policy. This one uses the following rules for a user ***u***, a requested dataset ***d*** and a permission type ***t***, whereby ***r*** = {r1, …, rn} is the set of roles which ***u*** belongs to.

1. If there is a data permission for user ***u*** or role ***v*** (whereby ***v*** ***r***), dataset ***d*** and permission type ***t***, then return allow. Else return deny.



Figure