DOCUMENTS

DOCUMENTS MANAGER

User Guide

Table of Contents

1. Introduction

The DOCUMENTS Manager application is an administrative tool used to set up and configure a DOCUMENTS principal.

Its essential functions are setting up user accounts, creating file types including fields and tabs, as well as folders in which processes in DOCUMENTS 4 are represented. You also set up archives and integrate workflows here.

This guide describes the most important areas, elements and functions for the DOCUMENTS Manager. All examples are provided by the demo principal *peachIT* that comes with the product. It is therefore useful to install this demo system and, if necessary, use it side by side with the real-time system to initially utilize changes to specific settings in a test.

Help

The DOCUMENTS Manager contains both a utility and context-related help. You can use this by initially clicking on the *?* icon and then, for example, on a field. A tooltip then provides a more detailed definition on use. In this way you will get more information which due to a variety of configuration options, might not be provided by this documentation.

2. Login

Start the DOCUMENTS Manager application via the program group DOCUMENTS 4 from the Windows start menu. Following this, the Login dialog shown in Fig. 1 opens.

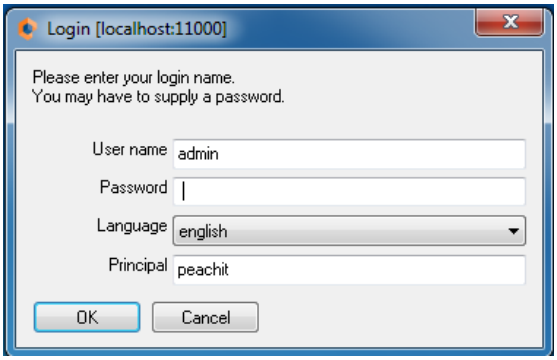

Fig. 1: DOCUMENTS Manager Login dialog

The following input is required:

- *User name*:
  Name of an editor, or specify the implicit account `admin` (default).
- *Password*:
  For the `admin` account you need to leave the password in the shipment state empty.
- *Language*:
  Choose the desired language in which to run the DOCUMENTS Manager. This can also be changed within the application.
- *Principal*:
  Name or number of your principal. To create a new principal, you need to leave this field empty, i.e. log in without a principal. If you have agreed to set up the Sample during *DOCUMENTS 4* installation, a demo principal named `peachit` will already be available.

Click OK to launch the DOCUMENTS Manager.

3. Design and Structure

The DOCUMENTS Manager (Fig. 2) is essentially composed of three panels and a menu structure.

In the left panel all available objects are organized in a tree structure. The individual items are described in detail in the chapters below.

When you select an entry from the tree structure, all objects of this selection will be listed in the top right panel. The bottom right panel in turn shows the properties, attributes and set values for the individual entry that you select in the top right panel.

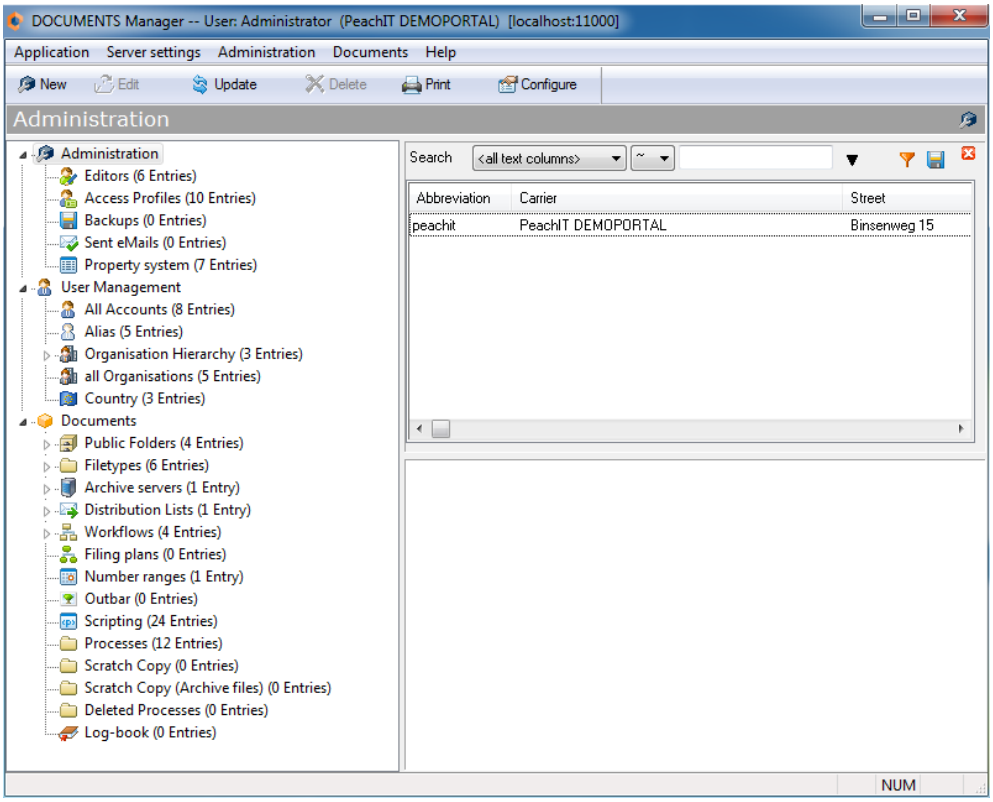

Fig. 2: DOCUMENTS Manager

Fig. 2 shows the DOCUMENTS Manager with installed demo principal `peachit`. Here, entries which will be taken as samples in the chapters below have already been created for the most important objects.

If you do not have installed the demo, you can log in only without principal (see Fig. 1). The DOCUMENTS Manager will then be empty and the tree structure will not be fully available. Moreover, you cannot create any new objects. These are always created for a principal and you cannot work with the Web application without a principal.

For further reading of this document it is therefore recommended that you set up a principal (see chapter 5.1).

4. Key Functions

The dialogs and tabs in DOCUMENTS 4 have an identical structure regarding functions and buttons. These emerge from the context and yield, for example, a new object of the open type, or sort lists of objects. If you keep the mouse button depressed on one of the buttons, the tooltip will display the corresponding hot key for this command.

Fig. 3 shows the generally available buttons in dialogs or tabs.

Fig. 3: General functions in dialogs

The following functions are available here (from left to right):

- *New record* icon: The left button always creates a new object. The graphic varies here, depending on the type of object viewed. Fig. 3 shows the example of the button for a new field of a file type.
- *Open folder* icon: Opens the dialog for editing a selected object.
- *Table* icon: Opens a searchable dialog containing a list of all available records to enable defining the linking of various objects.
- *Delete* icon: Deletes a selected record.
- *Disconnect* icon: Removes a connected record.
- *Arrow up / down* icon: Changes the position of a selected record in a list structure. DOCUMENTS 4 processes various lists (e.g. displayed fields of a file type) in order, from top to bottom.
- *Configure table* icon: Opens a dialog for configuring the displayed columns.
- *Information* icon: Opens a list of records in an overlapping dialog window and provides filter options for their representation.

Moreover, in the top section of the application window you will find a number of buttons that enable creating the items within the tree structure. Fig. 4 shows the available actions.

Fig. 4: Buttons for key functions

These functions are continually available in the DOCUMENTS Manager and always behave in the same manner. For this reason, we forego describing them in the chapters below.

5. Principal Management

The DOCUMENTS 4 Server and the archive component are *principal enabled* and therefore designed to simultaneously run on various customer systems.

A *principal* in this sense is a *controlling entry* and therefore a single unit within the DOCUMENTS system in terms of data and organization. Except for essential system configuration data, all data and objects are principal dependent, particularly those of the *DMS business concept*.

Access by users to the DOCUMENTS 4 Web application always takes place by logging in to a principal. Use without a principal is therefore categorically precluded.

Each principal requires a separate license file (`.pem`). This must be deployed in the DOCUMENTS 4 installation directory prior to creation or use.

5.1 Creating a principal

Launch the DOCUMENTS Manager and log in *without principal* (!) using the `admin` account (Fig. 5). In shipment state, a password is not necessary.

Fig. 5: Login without principal to the DOCUMENTS Manager

Confirm the warning that you want to run the application without selecting a principal (Fig. 6).

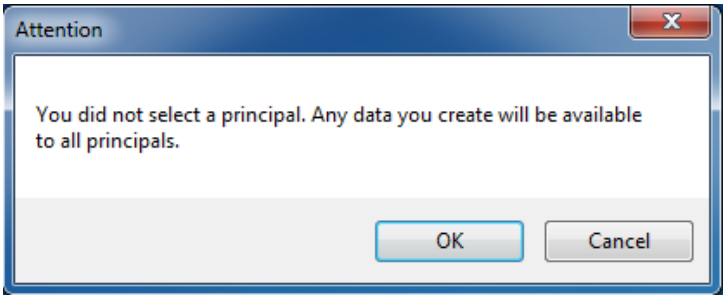

Fig. 6: Confirmation message for login without principal

The tree structure in the DOCUMENTS Manager (Fig. 7) has now significantly changed because all principal-related objects cannot be created anyway.

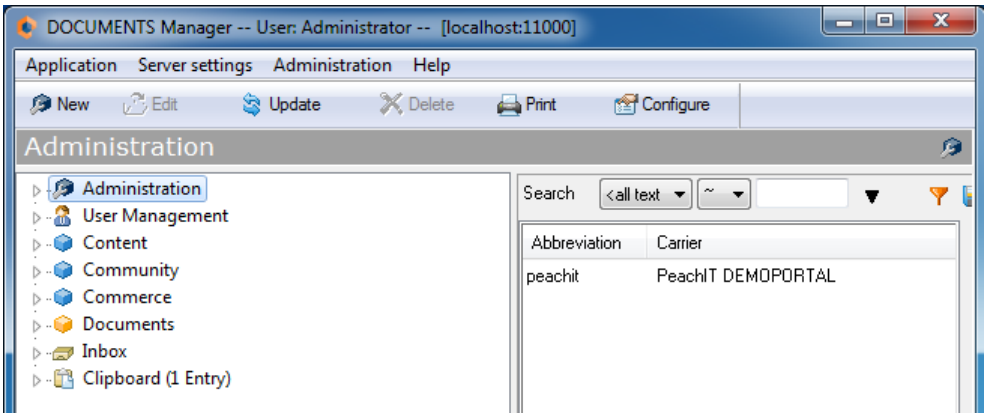

Fig. 7: DOCUMENTS Manager after login without principal

In the tree structure, select the *Administration* entry, and click the *New* button on top left.

This will open the dialog for configuring a principal (Fig. 8). Type the name of your principal here (equivalent to the name of your license file) in the *Abbreviation* field, and click *Apply*. Information on the *carrier* is automatically read from the license, and entered in the field of the same name.

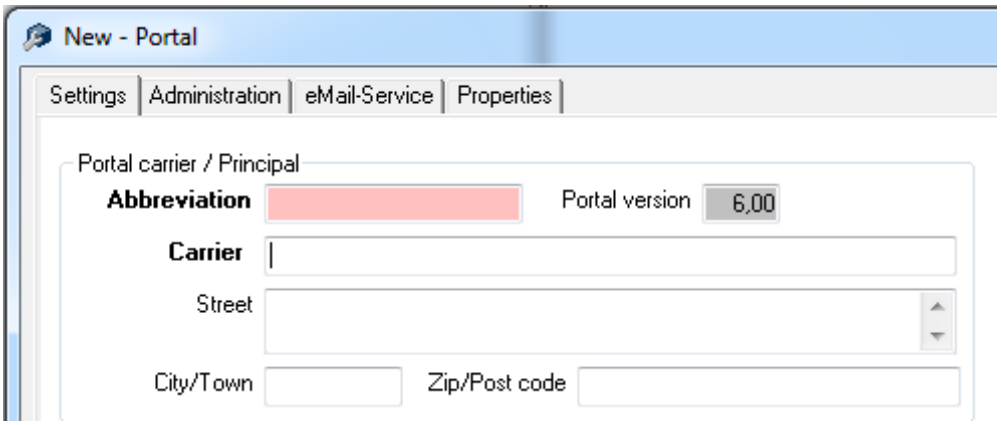

Fig. 8: Dialog for configuring a principal

Click *OK* to confirm the dialog, and exit the DOCUMENTS Manager via the "*Application -> Quit*" menu item.

Restart the DOCUMENTS Manager and log in *specifying the setup* principal.

5.2 Configuring a principal

Long-term important settings for the principal are made in the dialog for configuring a principal. These must be created for each principal, even if specific entries are redundant and are made multiple times here.

5.2.1 Settings

The left panel of the *Settings* tab (Fig. 9) allows optionally augmenting communication data on the *carrier* as well as on a *responsible employee*. The required fields *Abbreviation* and *Carrier* have already been populated while creating the principal.

The abbreviation here is a system-wide, unique value; in some places, it is used to allocate key values.

The right panel is used to define the languages used. A principal can run concurrently in up to six different languages. The international abbreviation for the language is entered in the *Locale* field. This usually corresponds to the extension for domain addresses in the respective country.

When the *online* option is enabled, the current language can be selected by users of the Web application. When this option is not enabled, you can edit the language in the DOCUMENTS Manager, but the end user cannot yet see it.

When adding a new language to DOCUMENTS, an administrator can already create that language, and enter all translations into the DOCUMENTS Manager without having to activate these during the development phase. When the translation is complete and the new language should be made available to the users, all you have to do is enable the *online* checkbox.

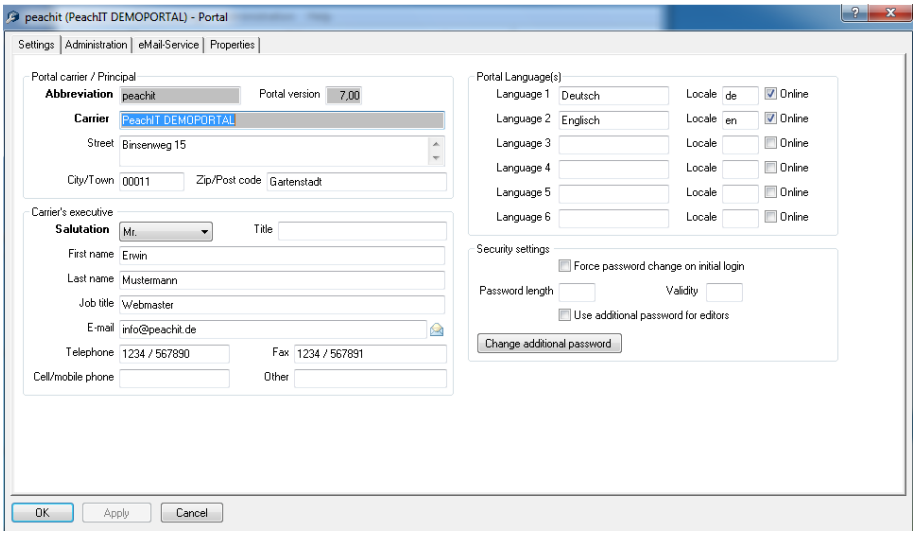

Fig. 9: Configuring a principal

The Security settings section allows defining password policies in the first place.

For instance, a user must change their password at initial login before they can work with DOCUMENTS for the first time. Moreover, minimum *password length* and its *lifetime* (in days) can be defined.

When connecting a central user administration (e.g. *Active Directory*) via the *LDAP* protocol, this area should not be used because policies will then be managed in the user administration.

5.2.2 Administration

Fig. 10 shows the *Administration* tab. Make sure that the *Documents* checkbox is enabled with the activated modules. Otherwise, there is an error in the license. Moreover, the status must be set to *released*.

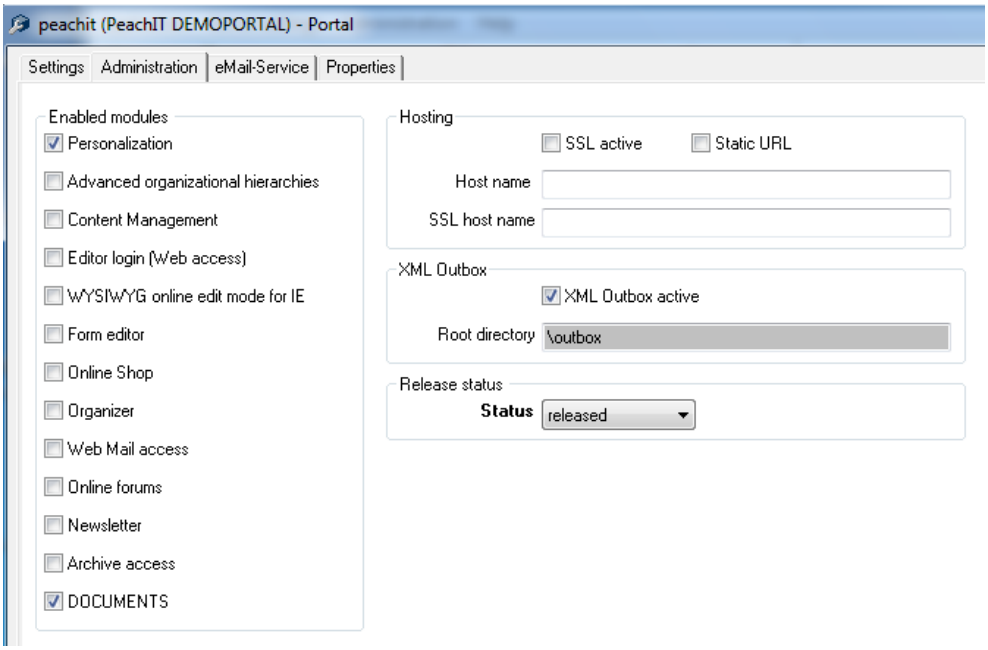

Fig. 10: The "Administration" dialog

5.2.3 E-mail settings

On the *eMail-Service* tab (Fig. 11) you define which credentials are used to authenticate DOCUMENTS 4 against a SMTP server.

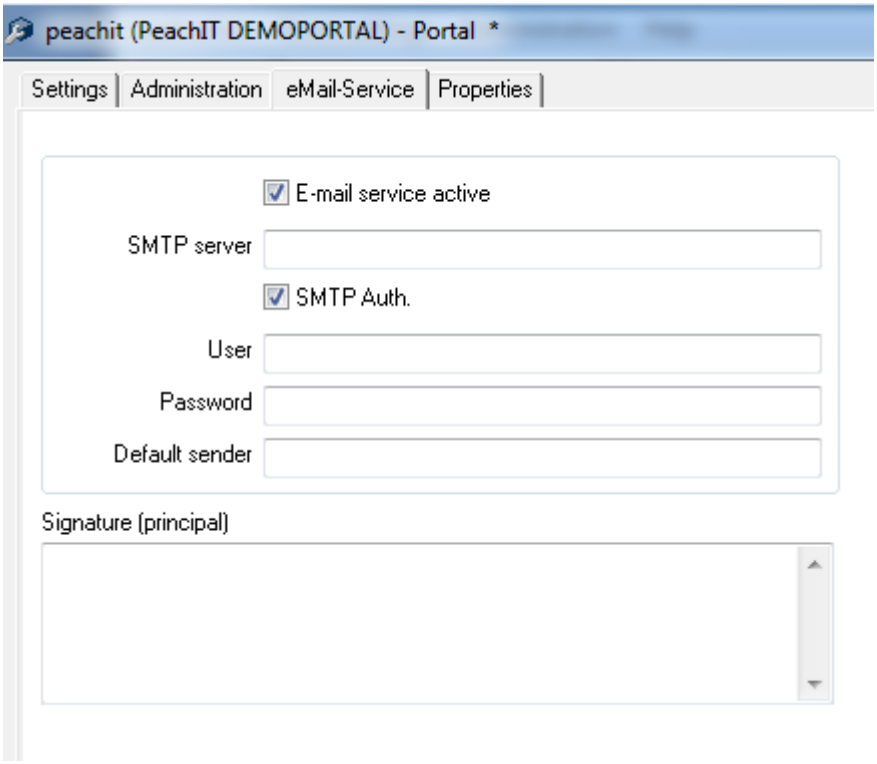

Fig. 11: E-mail settings

The *E-mail service active* checkbox is used as a global switch. If disabled, e-mail messages will not be sent principal-wide. Furthermore, it will not be allowed to make any entries in the following fields.

The *SMTP server* field requires either the IP address or the complete name (*FQDN*) of the SMTP server (e.g. Microsoft Exchange). Since DOCUMENTS 4.0d it is possible to specify a port using the notation ":port number" followed by the *SMTP server* specification, via which the e-mails will be routed. This specification overwrites the global property `SMTPPort` that can be found in the documents.ini configuration file.

Authentication can be turned on by ticking the same-named checkbox. If enabled, it is required to provide login information (i.e. *user name* and *password)* of an existing account with SMTP access. The principal property `SMTPAuthMethod` allows you to specify one of the following authentication methods: `plain`, `login`, `cram-md5` and `ntlm`, whereas the `plain` method is used by default.

The *Default sender* field enables to define an e-mail address under whose ID e-mail messages are to send by DOCUMENTS.

Please be aware that this address must, when in doubt, actually exist for authentication as well as for incoming replies. This address will be either used for automated system messages or for personal e-mail messages if no mail address

has been entered for the DOCUMENTS account of a user and therefore no personal ID is available.

By default, logged-in users send e-mail messages with the address specified in their account.

The optional *Signature (principal)* text-field allows specifying a signature.

Since DOCUMENTS 4.0d SSL/TLS encryption is supported when sending mail using STARTTLS. However, encryption is disabled by default. To activate encryption globally the property value `SMTPSSL` needs to be set to `true` in the documents.ini file. With the same-named principal property you can overwrite this global setting. In addition, a PEM/CRT file containing a list of trusted root CA certificates (in X.509 format) is still needed to verify the authenticity of the mail server. The path of the file including the root certificates, which is absolute or relative to the DOCUMENTS Server installation directory, is set as value of the `$CAInfoFile` property in documents.ini. In case multiple principals are using different mail servers, all corresponding root certificates must be listed in this file. If a certificate on the mail server side was not issued by a root CA, the certificate chain must be listed in the correct order in the file. More precisely, each certificate must stand ahead of its issuer certificate, at last the self-signed certificate of the root CA (root certificate). The individual certificates are in PEM format enclosed respectively by

`-----BEGIN CERTIFICATE-----`
and
`-----END CERTIFICATE-----.`

Whenever one of these root certificates is revoked (especially after an intrusion into CA servers), the file must be updated and it is required to restart the DOCUMENTS Server.

DOCUMENTS does not provide such a file. Neither can it keep the file up-to-date automatically, i.e. it lies in the responsibility of the administrators to maintain the file.

5.2.4 Properties

Properties are used to extend the function or configuration scope; they are entered with a name and a value. Many object types include a tab for adding properties. Separate documentation includes an overview of allowed properties.

6. Tree Structure Elements

The tree structure shown in Fig. 12 represents central administration of all objects comprising the business concept of a principal. At the first level, this breaks down into the three sections *Administration*, *User Management*, and *Documents*.

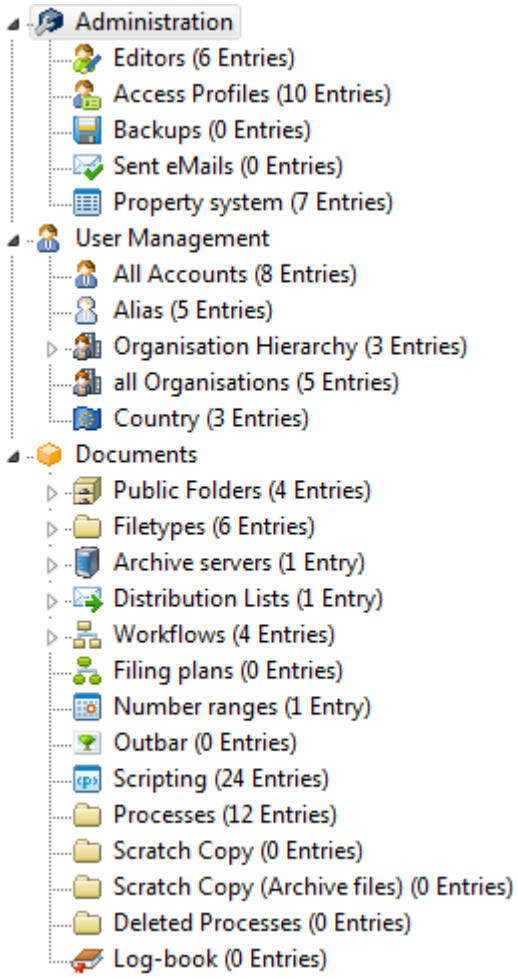

Fig. 12: The DOCUMENTS Manager tree structure

The most important tasks in the *Administration* and *User Management* sections are setting up accesses to DOCUMENTS 4. After creating *user accounts* these can be assigned to *access profiles* to group users with the same permissions and to therefore implement a permission concept.

In the *Documents* section, the business concept is actually built. Here different *file types* are defined which are the template for all files created in *DOCUMENTS 4*.

In DOCUMENTS, DOCUMENTS files are displayed via a structure of *public folders* which, like file types, may be subject to a permission concept. This is implemented by granting explicit permissions on files and folders only to specific users or access profiles.

Another aspect from the *Documents* section is setting up process-driven operations using *workflows* or *distribution lists*. These items facilitate specific routing of individual DOCUMENTS files within DOCUMENTS 4, and therefore rule-based editing of previously defined operations or task chains.

DOCUMENTS 4 also includes a *Java script engine* that can be used to implement smaller features. The source codes are stored in a library in the *Scripting* section; they can be linked to defined events.

7. Administration Area

The *Administration* area in the tree structure of the left pane breaks down into four different subsections which we will describe in greater detail in the course of this chapter.

7.1 Editors

Editors are user accounts working, like "normal" users, in DOCUMENTS 4, i.e. they may have DOCUMENTS access. The actual meaning of these nodes, however, is in two other functions.

First, editors can log in to the DOCUMENTS Manager and perform activities as part of administration and configuration.

Unlike the `admin` account, access to the DOCUMENTS Manager for an editor is always limited to the principal under which the editor account has been created.

Secondly, an editor account can be used to import DOCUMENTS files, e.g. via *DocImport* or a *Capture scenario*. In that case, new DOCUMENTS files are created by stacks, without requiring users to perform this manually via the Web application.

Editors should not, due to these two main tasks, be considered named, operationally active user accounts, but as service accounts automatically performing their operations.

In the tree structure, select the *Editors* entry, and click the *New* button to create a new editor.

This will open the dialog from Fig. 14 for configuring an editor. The option of subsequent processing is available anytime by clicking again on the entry in the tree, and then opening the desired entry by double-clicking in the top right pane (see Fig. 13).

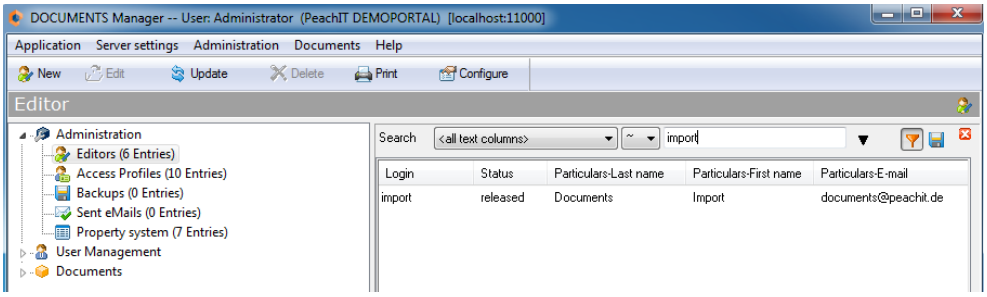

Fig. 13: List of created editors

The *General* tab (Fig. 14) starts with information on the editor's user account.

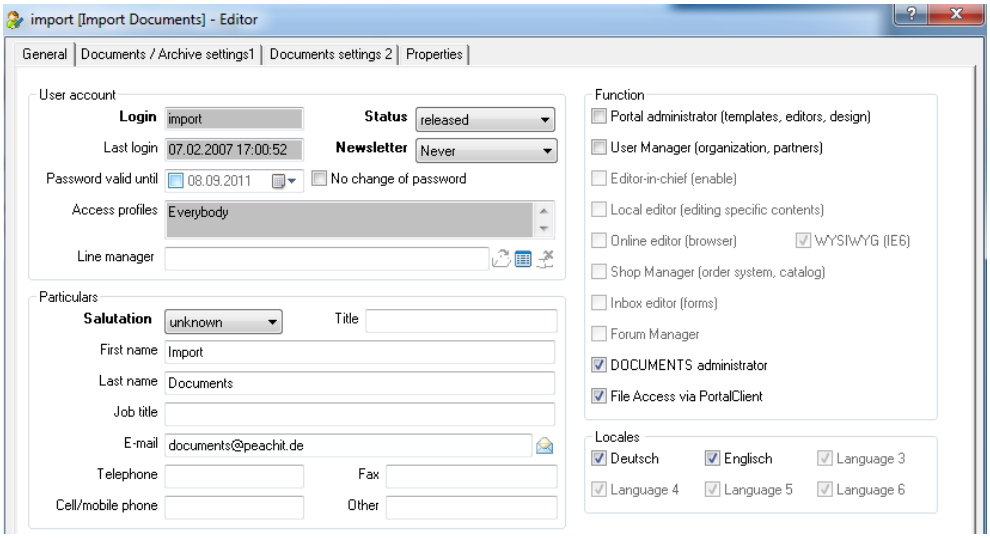

Fig. 14: The "General" tab

Specify a *Login* name as a unique ID first. This is bound to a host of restrictions: Initially, you may not use any special characters or umlauts. The length of the name is limited to 45 characters and the name must be unique principal-wide. This additionally implies that there may be no *access profiles* that use the same name. This uniqueness is checked when saving. After initially saving the editor, the *login* will, moreover, become read-only; it will then no longer be modifiable.

Ensure that the *status* is set to *released* and enter other *particulars*.

Enabling some checkboxes in the *Functions* section is useful:

- As *user manager,* the editor may create *user accounts* in DOCUMENTS 4 and manage them as well as assign them *access profiles*.
- As *DOCUMENTS administrator* the editor can administer the *Documents* tree section via the DOCUMENTS Manager and, for example, maintain file types.
- *File access via Portal client* means that the editor can view (or change, on a case-by-case basis) all DOCUMENTS files created by users in DOCUMENTS 4 in the DOCUMENTS Manager under *Processes*. This is equivalent to full access to all data created using the DOCUMENTS Manager.

You use this button, named *Change password*, to create a *password* for the editor or subsequently change it in the bottom right section of the dialog.

Fig. 15 shows the *Documents / Archive settings 1* tab. The policies for DOCUMENTS 4 access are controlled and global permissions are granted here.

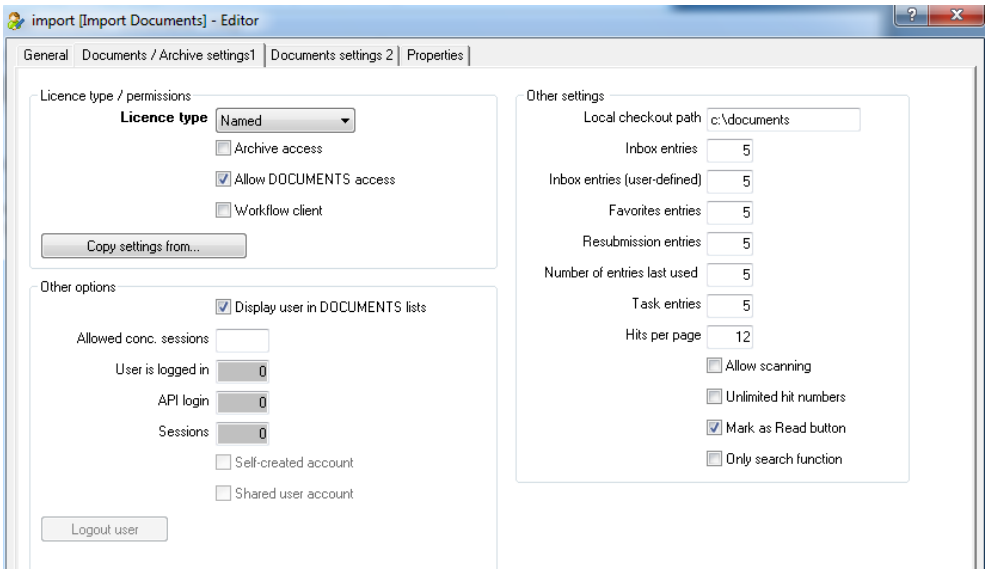

Fig. 15: The Documents / Archive settings 1 tab

When defining the *license type* you should be aware that a *Named* license will be recommended when using the account continually for imports, for example. Only this will ensure that access is granted anytime and data files to be imported are promptly imported into the system.

Moreover, determine whether the editor should have access to the archive component, the Web-side DOCUMENTS 4 application and the workflow client.

The *Display user in DOCUMENTS lists* checkbox controls that the editor should be displayed in the ad hoc routing or in user selection lists in DOCUMENTS 4. This will not be useful if an editor is only used for import but without any operational tasks by other users. If this is **not** included in the corresponding lists, you will not be able to send tasks to the editor.

The *Documents settings 2* tab incorporates some aspects that in DOCUMENTS 4 the *editors* can set themselves. These include *e-mail settings* and the option to enter scheduled *absence*.

In case an editor's absence is *unscheduled* and editing or sending DOCUMENTS files is therefore blocked, the *Documents settings 2* tab lets you set *absence* for other people (Fig. 16). To do this, enable the *User is absent* checkbox, and optionally configure an *absence message*. Moreover, the *Delegated files for information* checkbox lets you add a function that will place all DOCUMENTS files in the absent user's Inbox. In this way the absent user is informed about which DOCUMENTS files a delegate has edited.

However, the *Delegate current processes to* button is most of all important here. This allows you to enter the *login name* of an absent co-worker in a dialog, so all currently blocked DOCUMENTS files of the absentee will be delegated to the selected user for editing.

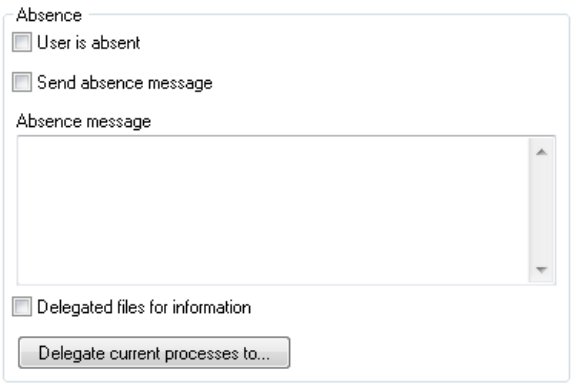

Fig. 16: Absence management

7.2 Access profiles

Access profiles are used to allocate permissions on DOCUMENTS objects. All users or editors whose permissions on specific objects should be identical are assigned to an access profile here.

The profile's permissions are automatically transferred to the users, editors and subprofiles entered there. This makes allocation of permissions easier and precludes error sources because it is no longer all people but only the related profiles that need to be given permissions.

On the *General* tab (Fig. 19), specify a unique *profile name* first. This is bound to a host of restrictions: You may not use any special characters or umlauts, and it must be unique principal-wide. This also entails that there may be no *editors* or *users* using the same name. This uniqueness is checked when saving. This name can no longer be changed after initial saving.

7.3 Assign access profiles

This tab (Fig. 17) lets you assign the *editor* different *access profiles* that already exist. For more information on creating and using access profiles, see 7.2.

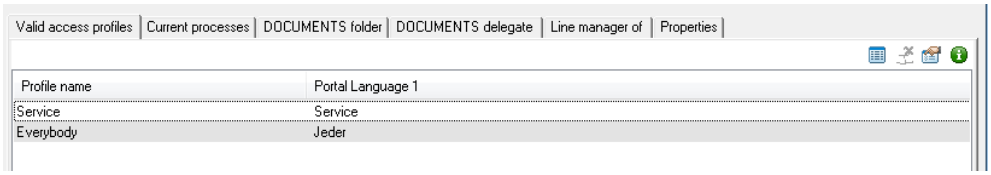

Fig. 17: Access profiles of an editor

On this tab you will see all access profiles to which the editor is currently assigned in a list view. The top right section of the tab provides buttons for more assignments or deleting assignments. The left button enables you to assign the editor more access profiles. This opens a searchable dialog for selecting from all profiles that already exist (Fig. 18).

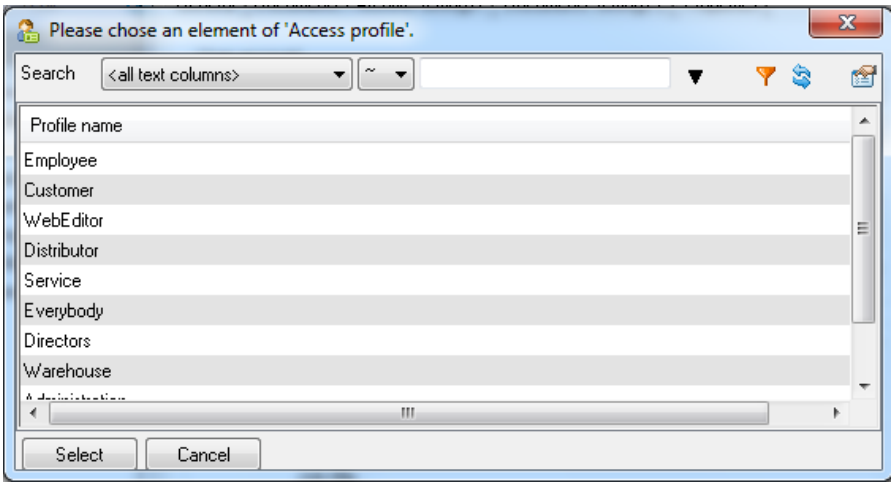

Fig. 18: Dialog for assigning access profiles

Mark the desired profiles here, and click *Select* to assign them.

Assignments are deleted by initially marking the access profile to be removed in the list (see *Each* profile in Fig. 17). Then click the button to delete the connection. Here only the assignment between editor and access profile is removed. The access profile itself is not deleted through this action (see chapter 4).

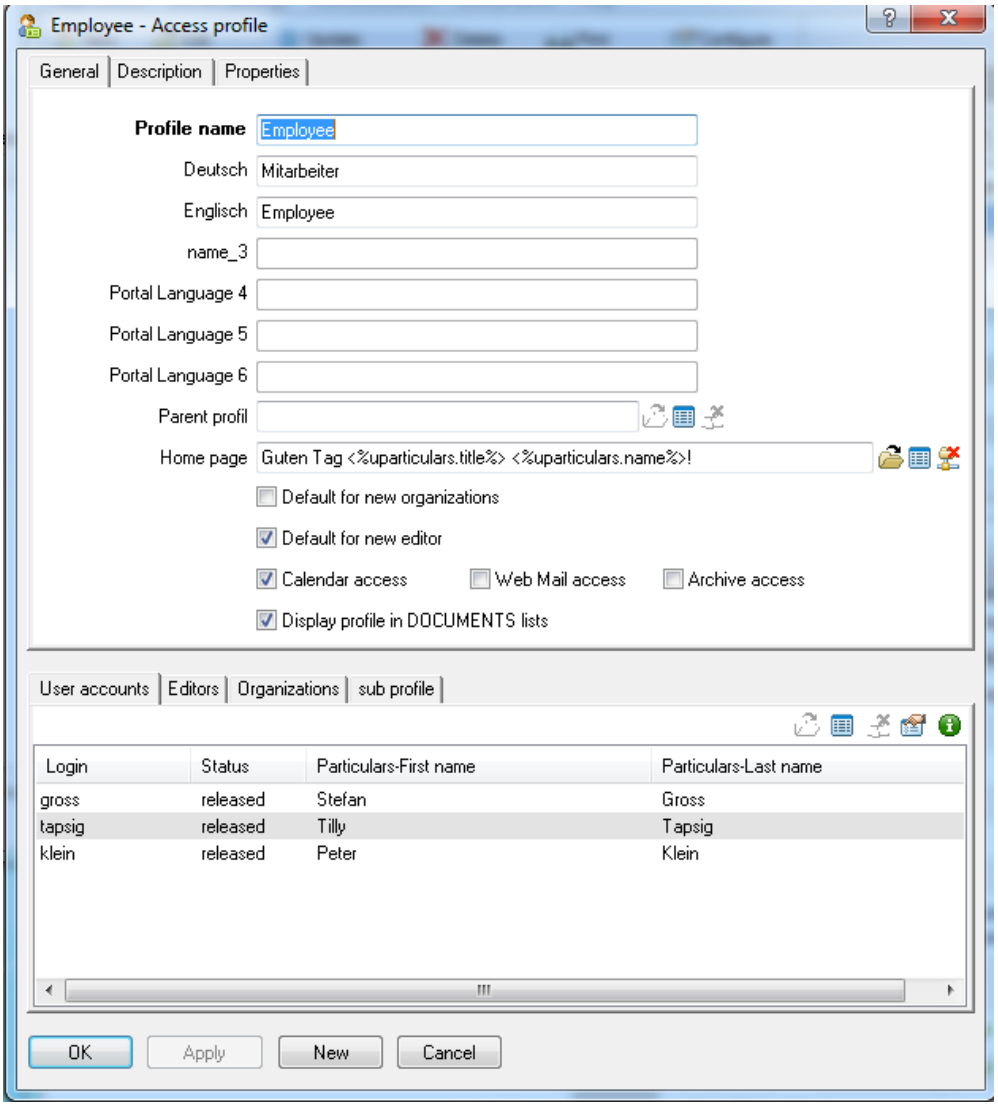

Fig. 19: The General tab

The following fields allow optionally allocating ergonomic names for up to six languages. Depending on the language in which a user logs in to the Web application, the displayed name of an access profile is resolved into the ergonomic name accordingly.

You can structure profiles hierarchically by defining a *parent profile* here, where the parent profile incorporates all permissions of the child profiles.

Fig. 19 contains more tabs where this access profile can be assigned users, editors and child profiles. Use the buttons on the right above the respective lists to add or remove assigned entries.

Alternatively, such an assignment can also be made on the user or editor dialogs. Both directions move in sync.

7.4 Backups

The DOCUMENTS Manager provides various options to perform backups of the existing DOCUMENTS 4 system. This is optionally performed as automated daily backup, or manually via offered actions.

Backups are useful for migrating a configured DOCUMENTS 4 system to another server or to make changing the underlying database management system easier. Such a backup is generally platform independent, and therefore makes migration to other systems easier.

> *The backup function by no means replaces a complete backup strategy! Although the option of daily backup is available, this is not working incrementally, nor does it ensure data integrity.*
>
> *Therefore, back up the database on a regular basis using the mechanisms available there, and set up incremental backups of the DOCUMENTS repository on a different drive!*

Complete backup consists of two files. One of these two files contains the database contents in compressed JEX file format, while the other contains the DOCUMENTS repository, i.e. the data files and documents as well as document templates attached to the DOCUMENTS files.

Important notes to backup duration and size

Backup duration is dependent on the amount of data as well as the number and size of documents within the repository; it may take up to a few hours. The required space on the drive also depends on these two factors; by necessity, it is at least the size of the repository because this is packed completely into a ZIP file.

Storage path for backups

The storage path for backups is set on the *Server settings -> System parameters* menu under the *Data backup/Directory for data backup* entry. You can set any UNC path there, where complete access to the path must be guaranteed. If the Documents4Server is running as a service, the login account of the service must be able to access this path. Whereas if the server is running as an application, the logged-in or executing user must have the corresponding rights there.

Verify, prior to backup, that the selected directory has enough storage space!

Principal and principalless backups

When you log in to the DOCUMENTS Manager, you ultimately decide what contents to back up: If you log in *by specifying a principal*, only the contents and parameters of that principal will be included in the backup. Whereas if you log in *principalless*, the contents of all principals existing within the system will be backed up in case multiple principals are present.

Another difference is in importing system parameters such as the paths to the repository or the backup directory:

For backup through specifying the principal, the parameters that have already been set there will **not** be overwritten during recovery to another DOCUMENTS installation. So, the principal uses the parameters of its new environment.

Whereas for principalless backup the parameters are also backed up and set during recovery. This means that, for example, a UNC path to the repository is set in the same manner after backup as was the case on the original server. If these paths cannot be reached on the new server, this may cause difficulties. Backup by specifying the principal is therefore the more portable option. For instance, you can use this to load a preconfigured principal on a customer's server because it will integrate with the specified environment.

7.4.1 Portal backup

Instant backup can be initiated anytime via the *Administration -> Perform portal data backup* menu item.

> *Please be aware that backup starts directly by clicking the menu entry. Other dialogs will not appear!*

The system automatically creates a new record named *Backup* for each backup process (Fig. 20).

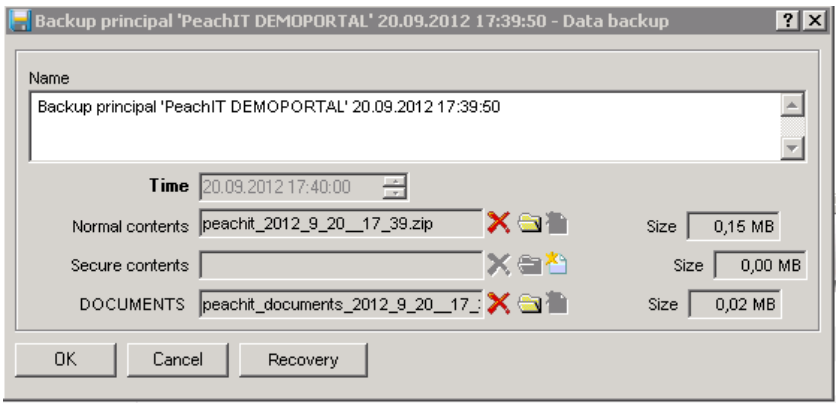

Fig. 20: Backing up a principal

You can copy the backup contents by downloading the created ZIP files from the DOCUMENTS Manager. The corresponding buttons are stored behind the individual files.

7.4.2 JEX export

During JEX export, only the database contents are written to a JEX file. This does not affect the repository contents. The references to the documents, however, are part of backup, so after recovery the DOCUMENTS files links to the data files will be recovered if the repository can be found in the specified path.

JEX export can be started via a corresponding button in the server application window. In this case, backup is performed automatically principalless.

Another option can be found on the *Server settings -> Server-side data export (jex)* menu. After starting the menu command, a dialog will open where you need to enter the complete path including file name and extension to the export file (Fig. 21).

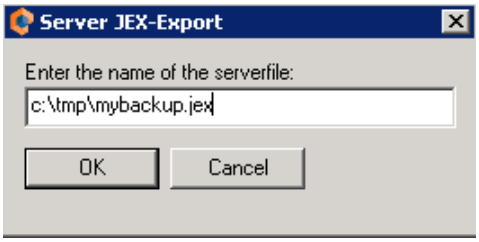

Fig. 21: File path to jex export

The differences between principalless and principal login on creating backup are also valid for JEX export.

7.4.3 Configuring daily backup

Regular backups are configured on the *Server settings -> System parameters* menu. Two parameters must be considered here:

- *Data backup/Time of automatic daily backup*: Enter a value in the 0-23 range to specify the hour of daily backup. Specifying a negative value will *disable* daily backup!
- *Data backup/Directory for data backup*: Specifying the storage directory (see above)

Daily backup will create automatic backup at the defined time. Here, the following files are written within the set directory:

- `daily.jex`: Daily database export (overwritten every day!)
- `dailydoc.zip`: Complete DOCUMENTS repository content (overwritten every day!)
- `daily_0.zip`: Analogous to `daily.jex`, but including the ID of the day of the week (0 = Sunday, 1 = Monday, etc.). These files will therefore be overwritten every week.

7.4.4 Recovering backup

Depending on the backup type (portal backup or JEX export), recovery is performed by different methods, which we will describe in the course of this chapter. The common features of both methods are the following important issues:

- Recovering a backup requires *principalless* login to the DOCUMENTS Manager.

- By recovering a backup, DOCUMENTS 4 is set to the state of that backup. Data that is added later will be lost. Therefore, use this operation with absolute caution!
- The principal to be recovered may *never* be present within the system! Please make sure this is the case and delete the principal prior to starting the recovery operation.

Indexing documents

If you have configured automatic indexing of documents on uploading to run the principal (defined for the file type), you will need to execute this again after successful recovery. Owing to its amount, this data will not be included in backup. Therefore, log in to the DOCUMENTS Manager after successful recovery and execute the `reIndex` maintenance operation. This is performed on the *Administration -> Perform maintenance operation* menu. All repository documents are re-indexed in batches and the corresponding tables for the search are rebuilt.

JEX export recovery

Log in *principalless* to the DOCUMENTS Manager and make sure that the principal to be recovered is not present. Server-side JEX export is recovered via the *Server settings -> server-side data import (JEX)* menu command. In the dialog that follows, specify the complete path name to the data file, including extension, and click *OK* to start importing the JEX backup.

Alternatively, a JEX backup can also be recovered via the *Import* button in the *Documents4 server* application's window. Also, make sure here that the principal does not exist prior to recovery. Log in to the DOCUMENTS Manager principalless prior to this, and check this!

Portal backup recovery

Log in *principalless* to the DOCUMENTS Manager and make sure that the principal to be recovered is not present. Portal backup recovery is performed via the DOCUMENTS Manager's tree structure. Initially, open the *Administration -> Backups* entry, and then click the *New* button. In the dialog that follows, enter a name for the backup first, and then select the data files of an existing backup. The `principal_TIMESTAMP.zip` file is selected as *normal content*, while the `principal_documents_TIMESTAMP.zip` file is selected as *DOCUMENTS content*. Fig. 22 illustrates this procedure.

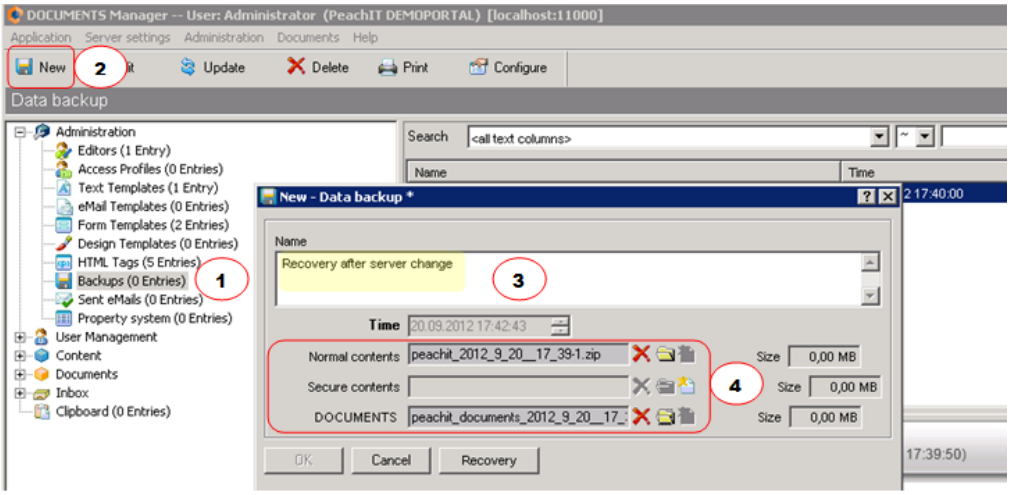

Fig. 22: Portal backup recovery

Following this, click the *Recovery* button and, in the dialog that follows, enter the name of the principal to be recovered. By clicking *OK* you start the recovery action.

7.5 Sent eMails

This area allows checking editing and resending e-mail messages that have already been sent. An error code informs about the status of the e-mail message. By creating a new message and by using the Send function, e-mail messages can also be sent via the DOCUMENTS Manager.

8. User Management

This area is used to set up accounts for the users to be granted access to DOCUMENTS 4 or this archive component.

To do this, select the *All Accounts* entry in the tree structure, and click the *New* button.

Fig. 23 shows the *General* tab of the dialog for creating a new user account.

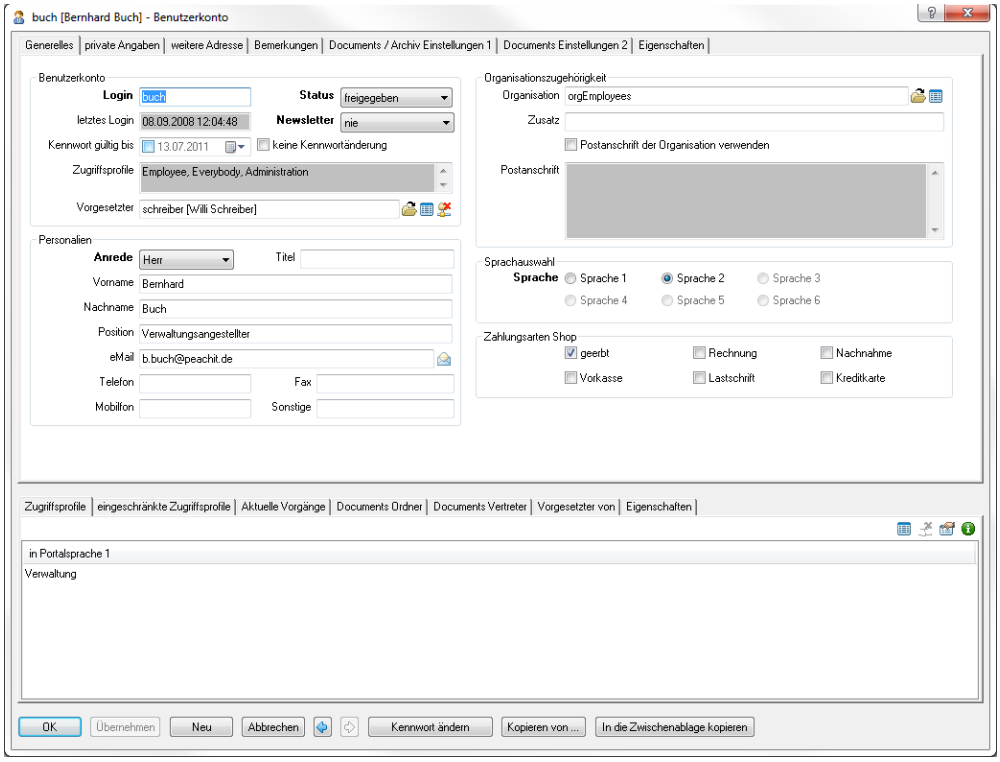

Fig. 23: The "General" tab

Specify a *Login* name as a unique ID first. This is bound to a host of restrictions: Initially, you may not use any special characters or umlauts. The length of the name is limited to 45 characters and the name must be unique principal-wide. This additionally implies that there may be no *access profiles* that use the same name. This uniqueness is checked when saving. After initially saving the user, the *login* also becomes read-only, and can then no longer be changed.

Ensure that the *status* is set to *released* and enter other *particulars*.

The *Documents / Archive settings 1* tab is used to control the policies for accessing DOCUMENTS 4 and to allocate global permissions.

Name the *license type* to be used here and, moreover, determine that the user should have access to the archive component, the Web-side DOCUMENTS 4 application or the workflow client.

The *Display user in Documents lists* checkbox controls that the user should be displayed in the lists of "ad hoc" routing or in the user selection lists in DOCUMENTS 4.

Moreover, the *Access profiles* tab lets you define the user's group memberships.

Eventually, create such a profile for the user via the *Password* button. After saving the dialog, the user can log in to *DOCUMENTS 4* using their *credentials* and password.

The *Documents settings 2* tab incorporates some aspects that DOCUMENTS 4 enables the *editors* to set for themselves. These include *e-mail settings* and the option to enter scheduled *absence*.

In case an editor's absence is *unscheduled* and therefore editing or sending DOCUMENTS files is blocked, the *Documents settings 2* tab lets you set *absence* for other people (Fig. 16). To do this, enable the *User is absent* checkbox, and optionally configure an *absence message*. Moreover, the *Delegated files for information* lets you add a function that will place all DOCUMENTS files in the absent user's Inbox. In this way the absent user is informed about which files a delegate has edited.

However, the *Delegate current processes to* button is most of all important here. This allows you to enter the *login name* of an absent co-worker in a dialog, so all currently blocked DOCUMENTS files of the absentee will be delegated to the selected user for editing.

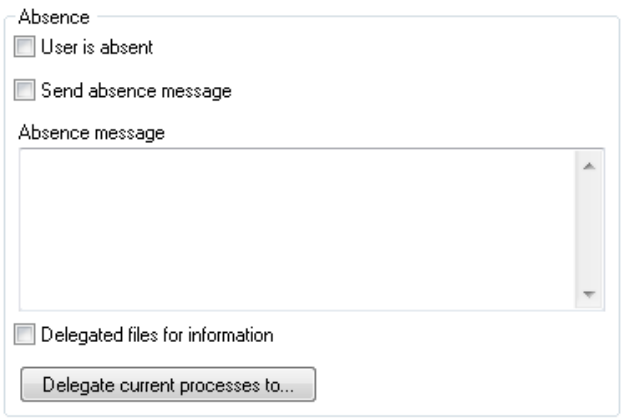

Fig. 24: Absence management

Assigning access profiles

The *Valid access profiles* tab (Fig. 17) lets you assign the *user* different *access profiles* that already exist. For more information on creating and using access profiles, see 7.2.

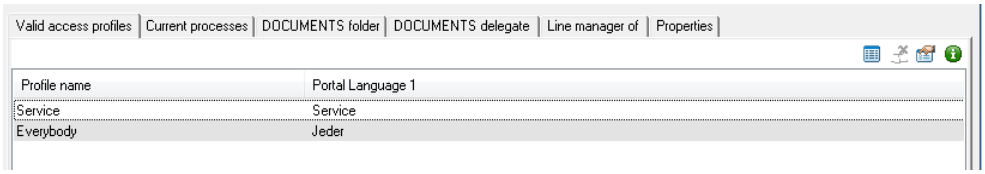

Fig. 25: Access profiles of an editor

On this tab you will see all access profiles to which the user is currently assigned in a list view. The top right section of the tab provides buttons for more assignments or deleting assignments. The left button enables you to assign the editor more access profiles. This opens a searchable dialog for selecting from all profiles that already exist (Fig. 18).

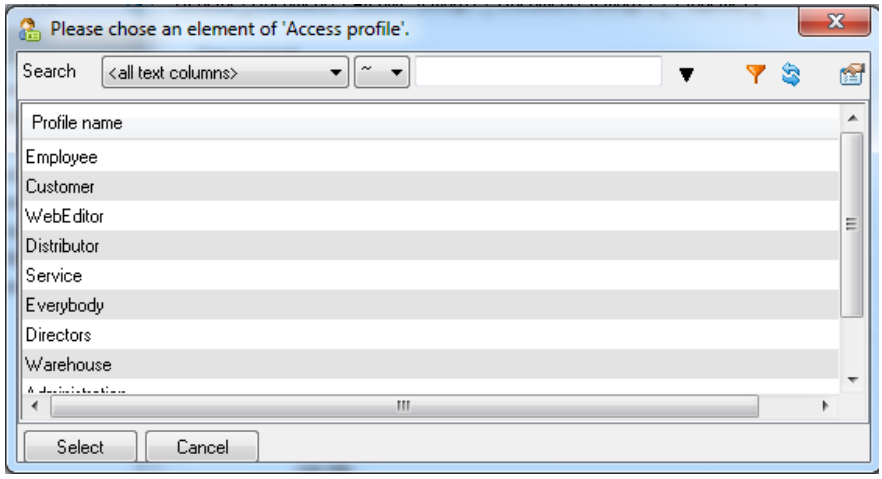

Fig. 26: Dialog for assigning access profiles

Mark the desired profiles here, and click *Select* to assign them.

Assignments are deleted by initially marking the access profile to be removed in the list (see *Everybody* profile in Fig. 17). Then click the button to delete the connection. Here only the assignment between editor and access profile is removed. The access profile itself is not deleted through this action (see chapter 4).

DOCUMENTS AREA

In the *Documents* area you determine the business concept of your principal by defining *file types*. You define which data the users can enter in DOCUMENTS 4 and you facilitate document storage. For each file type *fields* are created which are displayed on the cover of a DOCUMENTS file in DOCUMENTS 4. Depending on the *field type*, users can enter specific information.

DOCUMENTS files are displayed in *public folders* which are also created in this area. Filter criteria are used to control which files are displayed in a folder. Moreover, you can limit the group of users for each file type by allocating permissions. Step by step you can build a document management system for a wide range of business units and purposes of use.

9. File Types

File types are the key element in DOCUMENTS 4 because all connected data is saved in the form of digital *files*. These can be compared to the popular files or folders used in everyday office life; they contain a wide range of documents and information.

You should therefore define a separate file type for each category of managed documents (e.g. invoice, vacation request, contract) to provide the different pieces of information.

The appearance of a file type with regard to contents and structure of the entered data is defined through its *fields* and *tabs*.

Fig. 27 shows an example of a DOCUMENTS file of the *Invoice* type from the demo scenario *peachit*. This is organized in three tabs named *Fields*, *Documents*, and *Status*. The *Fields* tab (open in Fig. 27) manages – as the name suggests - *fields* and provides the most important information specifically on an invoice.

The fields defined for the file type are displayed on the Web in DOCUMENTS 4, and are maintained by the users. Ultimately, fields are used to clearly visualize the essential information from the invoice document and make it specifically searchable without requiring the user to open the actual document.

The actual invoice document is stored on another tab of the DOCUMENTS file. In addition to a *field tab* (i.e. a tab that can include fields), you therefore need to define a *document tab*. This provides an entirely different functionality from the field tab because it is designed to upload and display documents; however, it is equally an integral part of each DOCUMENTS file of the corresponding type.

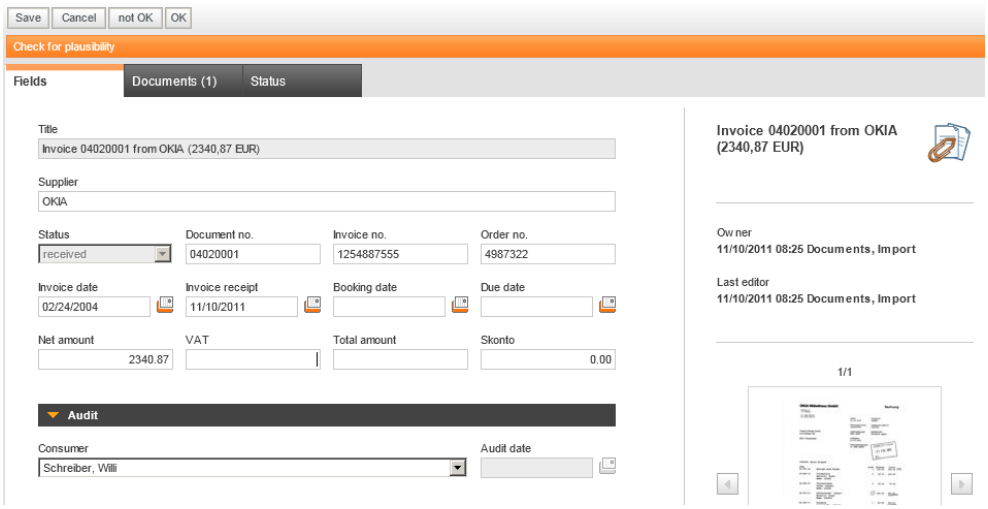

Fig. 27: Displaying a DOCUMENTS file in the DOCUMENTS 4 Web application

In the tree structure, select the *Filetype* entry and click the *New* button to create a new object. Fig. 28 illustrates the basic structure taking the example of an existing file type from the peachit scenario.

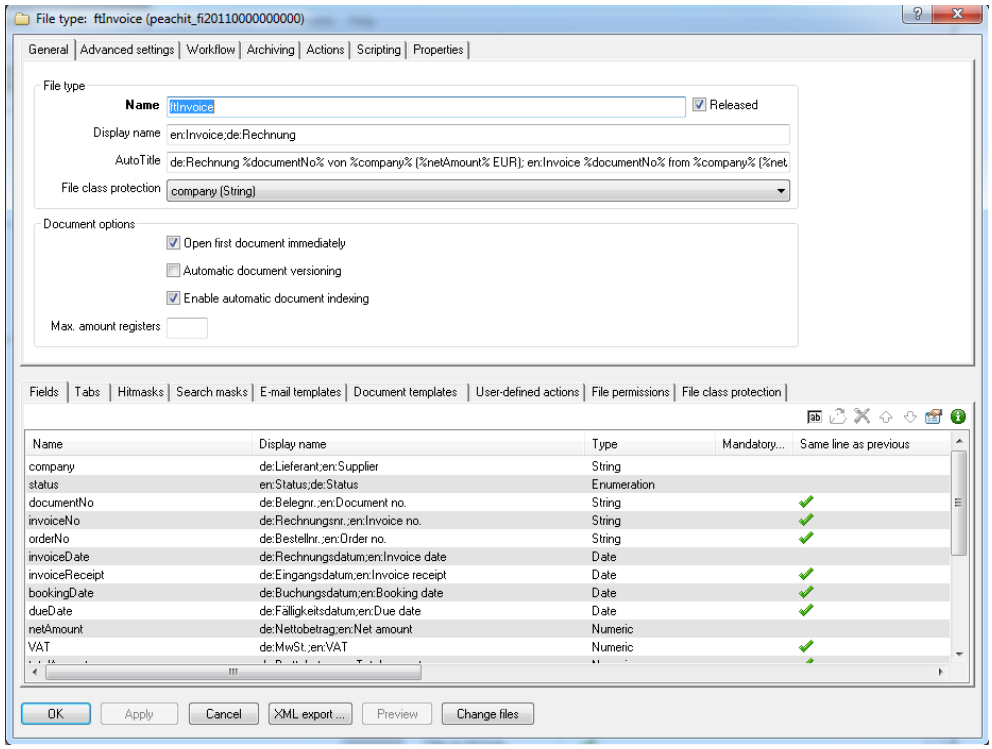

Fig. 28: Structure of a file type

9.1 General

Allocate a unique *name* and an ergonomic *display name* first. The *name* may not contain any umlauts or special characters; it should be chosen in such a manner that the data to be managed can be recognized by it (e.g. `ftInvoice` for the *Invoice* file type, where `ft` is the abbreviation for *file type*).

If the created file type is to be used in DOCUMENTS 4, you will have to enable the *Released* checkbox.

The *display name* may appear in multiple languages through the use of the *Locales* defined for the principal, as the following example illustrates. Here, `de` and `en` represent the corresponding *locales*. A colon is followed by the ergonomic name of the desired language. The separator used between two language entries is a colon:

```
de:Mitarbeiter;en:Employee;
```

AutoTitle

DOCUMENTS files are named via a *title*. The users optionally enter this title as free text, or the system automatically allocates it. In the case of system allocation, the structure is, by default, a combination of the name of the file type and the timestamp of the creation of the relevant DOCUMENTS file.

This structure can be customized by allocating an *AutoTitle* which combines static text and field values of the DOCUMENTS file. To differentiate, referenced field values are enclosed by percent signs. In the example from Fig. 28 the AutoTitle has the following structure, where field values - unlike static texts - are highlighted in yellow:

```
de:Rechnung %documentNo% from %company% (%netAmount% EUR)
```

In Fig. 28 below, moreover, the fields used here are shown.

When using an AutoTitle, the users naturally cannot specify a title. Entered field values are automatically imported into the file title when saving DOCUMENTS files. Fig. 29 illustrates the context, where the AutoTitle is displayed in the dark section on top.

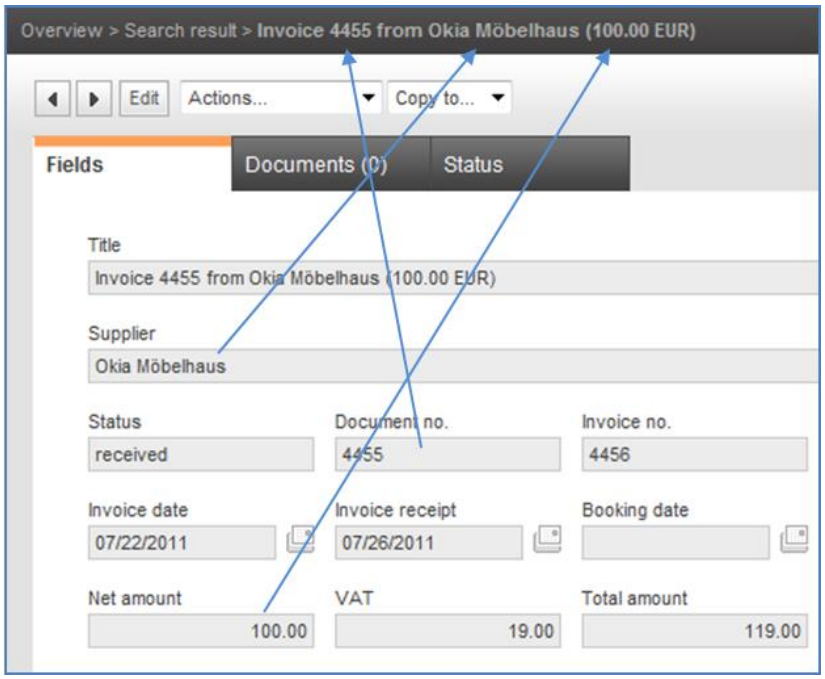

Fig. 29: Field values in the AutoTitle

The use of AutoTitles is particularly useful with inhomogeneous hit lists such as Searching across multiple file types, in the *Inbox folder* or in the *Recently used* folder. Here file titles are useful which provide information about the respective type and the most important contents. Fig. 30 in this context shows a folder containing two DOCUMENTS files of different types and with different titles.

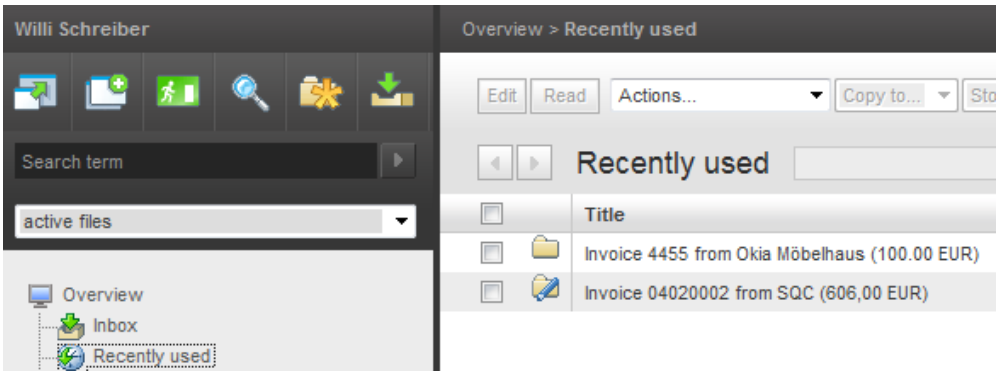

Fig. 30: Folders containing DOCUMENTS files of different types

9.2 Fields

By defining fields you determine which information and data is saved for a specific DOCUMENTS file (in addition to the documents). Fields are always defined in the context of a file type, so all files of a type have an identical data structure.

Fields are either displayed on the (implicitly present) file cover or on an explicitly created field tab.

The values are normally entered by the users; however, automated (pre-)entries are also possible, e.g. the timestamp of creation of a new DOCUMENTS file.

Fields are created on the tab of the same name of the desired file type via the left button.

9.2.1 General

On the *General* tab (Fig. 31), enter a *name* first. The use of umlauts or special characters is not permitted. In the next line you enter a(n optionally multilingual) label. The *None* checkbox can be used to explicitly forego the use of a label.

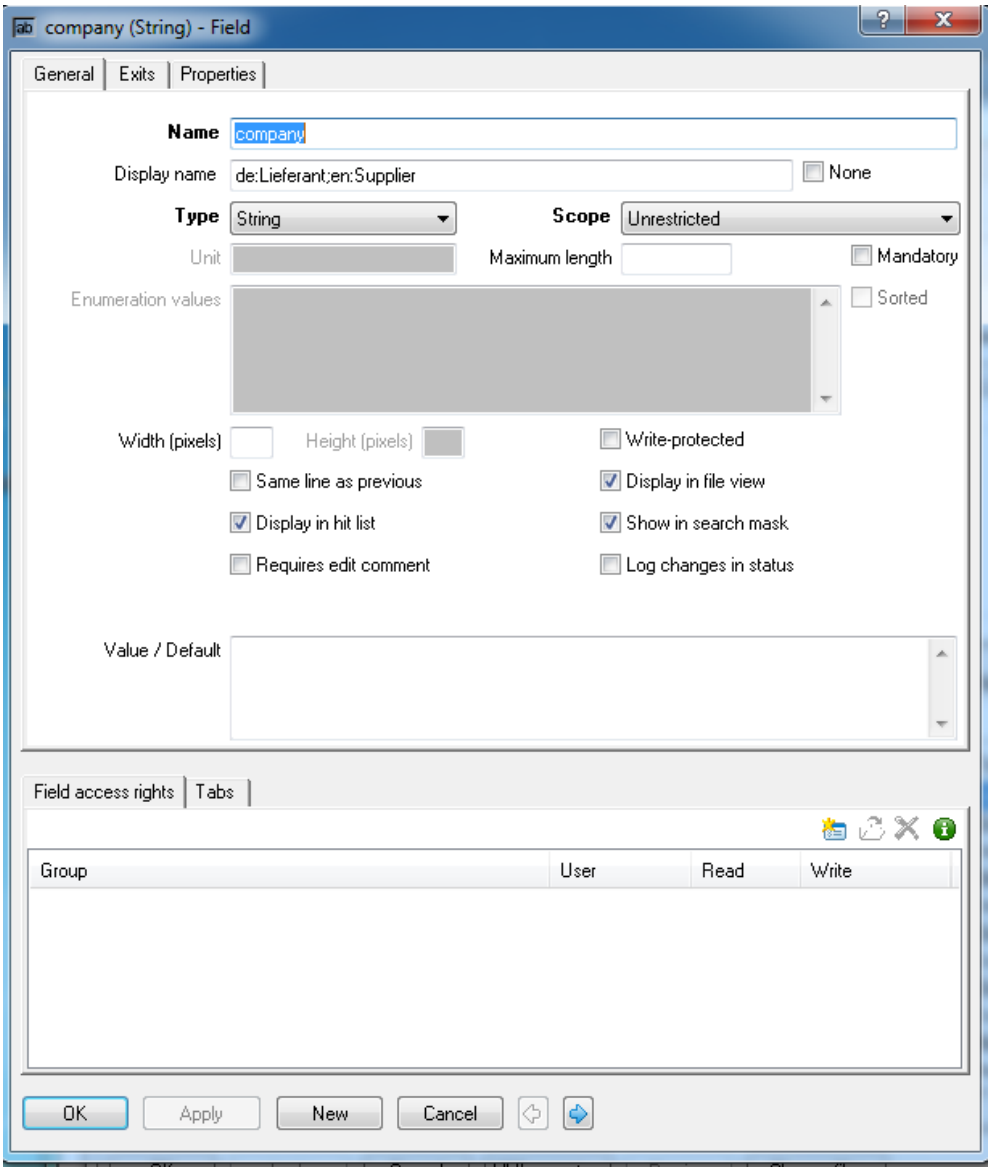

Fig. 31: Entry dialog of a field

9.2.2 Field type

Type defines which data the field can hold. Besides these contents, the appearance of the field is influenced by selecting the type. The most important field types have the following properties:

- *String*: A string may take alphanumerical values; its height is limited to one line.
- *Checkbox*: Checkbox which is enabled by putting the checkmark in the box.
- *Date*: Date field including checking the entered date and automatically entered selection calendar. This can be opened via a button next to the field on the right.
- *Enumeration*: List with a fixed selection of *enumeration values*, automatically *sorted* on request.

- *Horizontal ruler*: This field type is used for better structuring and division into the form sections. In Fig. 27 for example, the "*Audit*" heading is nothing other than such a horizontal separator. An area grouped by such a separator can be collapsed and cascaded on the user side.
- *Numeric*: Field type that only accepts number values. You can additionally define a unit (e.g. a currency or a dimension) and an *enumeration value* `n.2` for numeric fields, so two decimal places are automatically created on saving (other numbers are also allowed).
- *Text*: As with the string, alphanumerical, except that text fields are multi-line. A vertical scroll bar is used for scrolling.
- *Reference*: Reference fields replicate a link to another transaction file. Here you must determine which file type should be referenced and which fields of the file type should be entered in the reference field. For more information on the exact syntax, see in chapter 9.2.6 below.
- *HTML*: Basically, this is a text field; however the content is saved in HTML format, so some text highlighting options (bold, italics, etc.) are provided.
- *Double selection list*: Basically, this field type is also an enumeration; unlike the enumerated list, however, you can enter multiple values.
- *Bool*: Yes/No field.
- *History*: Also a text field. The contents, however, continue to be recorded chronologically and an automatic note is made saying when and by which user the entry was continued.

The *maximum length* allows limiting the entry to a number of characters.

When you enable the *Mandatory* checkbox, the DOCUMENTS file can only be saved if the field is populated.

9.2.3 Enumeration values

If you have selected a new field of the *Enumeration* or *Double selection list* type, you need to insert the individual enumeration members here. For all other field types, an entry in *enumeration values* is not expected. Entries are simply separated by pressing the ENTER key. Enumeration fields appear in *drop-down list* format. This drop-down list cannot accept values other than the defined enumeration values.

The *Numeric* field type in turn is an exception: As described above, the Enumeration values attribute can be used here to define a fixed number of decimal places (e.g. `n.2`).

When you enable the *Sorted* checkbox, the individual entries will be automatically sorted in DOCUMENTS 4 (`A..Z` or `1..9`), even if you have entered the values here in a arbitrary order. This option is only useful for the above fields of the *Enumeration* and *Double selection list* types.

9.2.4 Size and order

By default, a field always occupies the entire available form width. Optionally, you can specify a *width* (in pixels) to control this behavior. Analogous, the *height* can be specified in pixels for multiline field types.

If multiple fields are accommodated in a single line, these will proportionally share the available space. If a fixed width is entered for a field, the other fields will proportionally use the remaining width.

The *Same line as previous* checkbox is used to define that a field is not positioned in a new line but displayed in the same line as the field that is residing directly in the field list above the previous one. This, then, is the *previous line*. You control the order of the fields (and, with it, the question of the respective predecessor) via the buttons with the blue arrow icons or via Drag& Drop in the field list.

This position allows consulting Fig. 28 and Fig. 29 again: The order of the fields in DOCUMENTS 4 corresponds to the order of the field list in the DOCUMENTS Manager. Moreover, in Fig. 28 it is determined for the "*Document no.*" (`documentNo`) and "*Invoice no.*" (`invoiceNo`) fields that these should be positioned in the same line as the predecessor. In Fig. 29 the *document number* can be found after its predecessor *Status* and, (still) in the same line, the *Invoice number* in turn after its predecessor *Document number*.

9.2.5 Display options

To make a field visible, you need to *display it in file view*. This option is enabled by default; it can be removed when required, e.g. for auxiliary fields. Thus, you can save specific values for a DOCUMENTS file which should not be viewed or edited by the users.

The *Display in hit list* checkbox causes the field in the tabular overview of a public folder to be displayed.

Another option named *Show in search mask* ensures that this field is taken as an independent filter criterion when using the global search. When this checkbox remains disabled, this field can only be searched via the integrated full text search engine.

When creating a new DOCUMENTS file, a field is automatically used with the optional entry from *Value / Default setting*. To prevent subsequent changes to the default setting by users, you can also use the ***write-protected*** option.

9.2.6 Using reference fields

Reference fields replicate a link to another transaction file. Here you must determine which file type should be referenced and which fields of the file type

should be entered in the reference field. Fig. 32 shows an example of a reference field.

Fig. 32: Reference field

In edit mode, you can select a DOCUMENTS file to be linked with the relevant DOCUMENTS file via the icon to the right of the field. The selection is made via a dialog window with an automatically integrated search function.

When saving the DOCUMENTS file afterwards, you can directly jump to the referenced DOCUMENTS file via the same icon by clicking, and view all information in single view there.

When creating a reference field, at least two pieces of information that must be entered in a specific syntax are useful:

- *Line 1*: Technical name of file type, the DOCUMENTS files of which should be made available for selection. Separated by a dot, this is followed by a key field that can uniquely identify any DOCUMENTS file of this type.
- *Line 2*: Fields of the referenced file, the contents of which should be entered in the reference field.

Fig. 33 shows an example of the definition of a reference field which references a user's delegate with the last name and first name in the *peachIT* environment.

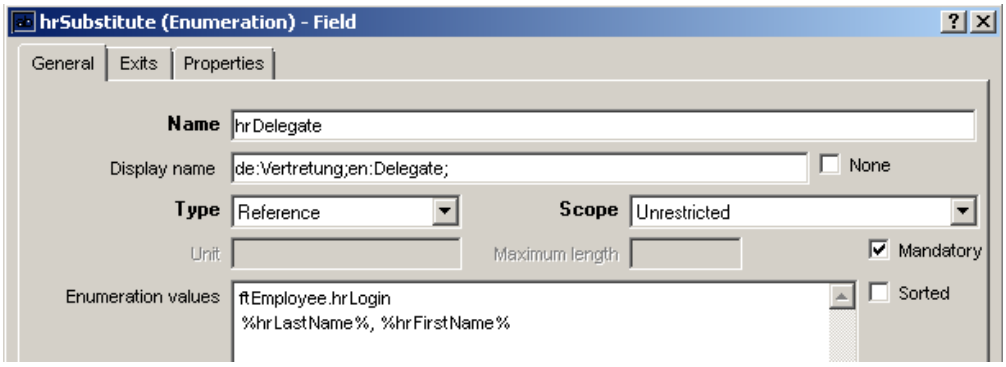

Fig. 33: Syntax of a reference field

In *Line 1* it is determined that the reference generates a selection dialog in which only files of the *Employee* type (`ftEmployee`) are available for selection. These are identified via the `hrLogin` field. It is presumed that the login name of an employee is unique system-wide.

In *Line 2* you define which *fields* from the linked DOCUMENTS file should be entered in the reference field, i.e. which fields are visible as field content.

In Fig. 32 the *last name* (`hrLastName`), followed by a *comma* and then by the *first name* (`hrFirstname`) is entered in the reference field as an example. You will obtain all additional information on the referenced DOCUMENTS file anyway by clicking the reference because in this way you jump to the individual view of the referenced DOCUMENTS file.

Fig. 33 shows the corresponding structure in *Line 2*:

```
%hrLastName%, %hrFirstName%
```

The fields from the referenced DOCUMENTS file are enclosed in the reference field by the usual replacement flags (`%`), as is customary with e-mail templates, auto titles, etc.

9.2.7 Field access rights

Specific fields can be equipped with special permissions within a DOCUMENTS file. This is useful, for instance, when a file type is used by different addressees but specific information should not be accessible to the entire group of people. Thus, for instance, the salary could be entered in the *Employee* file type from the *peachIT* principal. However, it should only be visible to the directors. Other users who, for example, enter the employee's sick notes, should not be allowed to view this information.

Such scenarios can be implemented via *field access rights*.

To allocate such a right, go to the corresponding tab and click the button to *Create new record*.

The dialog shown in Fig. 34 lets you allocate permissions for any combinations of *user accounts* (including editors) and *access profiles* (in this example, *groups*). These are always assigned sequentially. So, access profiles with identically scheduled permissions cannot be authorized through multiple selection in a single step; instead, they need to be processed one after the other.

> *Permissions should preferably always be allocated at access profile level. In practice, controlling and subsequently entering the personal permissions of individual (changing) users transpires to be complicated in the long run.*

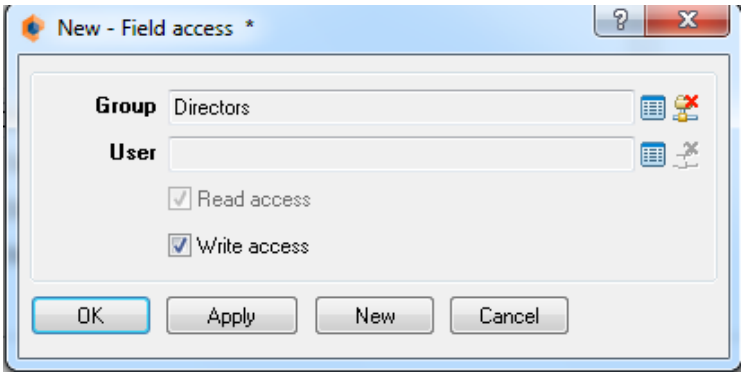

Fig. 34: Field access rights

This always grants *read access* to the field. Another checkbox allows granting additional *write access* to the selected access profile or user.

Overall, the principle of a *positive permission* applies here: Until permissions have been allocated, all existing access profiles and users have full permissions on the relevant field. However, once a permission has been allocated, all permissions are withdrawn from all other access profiles and users. So, when starting to allocate permissions for a field, this must be persistently completed for all existing access profiles and users. For this reason, the read permission cannot be de-selected. If a user should not be granted permissions to the contents of this field, you simply allocate permissions for all other items, but not for this user.

9.3 Tabs

In addition to fields, tabs represent another structuring feature of DOCUMENTS files.

Each file type at least contains a tab to display the contents. This *file cover* is implicitly always available; you do not have to create it.

Each field created is initially displayed on the *file cover* unless it is expressly assigned to another tab.

Other tabs can be added to any file type where, unlike the file cover, an individual tab type is defined for new tabs. By allocating a type it is determined which type of information is to be displayed on the tab. Each type is optimized with regards to its use and automatically provides essential features for the managed functions. The most important tab types are explained below.

9.3.1 Field tabs

Field tabs display the fields of a file type. The implicitly existing file cover ensures that creating new field tabs is, in principle, not necessary because the file cover itself is implemented as a field tab. Yet in specific scenarios creating other field tabs makes sense: Transparency with complex file types including a variety of fields will improve when transferring related fields to another tab. Moreover, a permission concept can be implemented where specific users or access profiles may view or edit only parts of a DOCUMENTS file. Here the corresponding fields can be placed on separate tabs where permissions are implemented at tab level. This is significantly easier to administer than allocating permissions at the level of individual fields, although allocating permissions at field level is also possible.

9.3.2 Document tabs

The data files and documents of a DOCUMENTS file are managed on document tabs. Basic functions for uploading, displaying, renaming or deleting documents have already been implemented here.

Fig. 35 shows a document tab in DOCUMENTS 4. The documents are displayed in a list; the document-specific functions that this tab type automatically provides can be found above it.

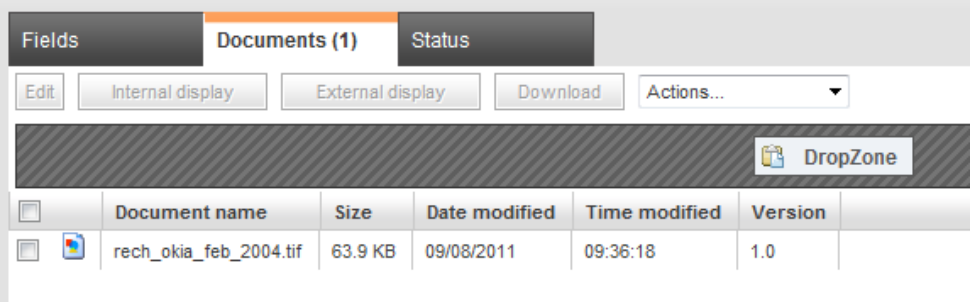

Fig. 35: Overview of a document tab

Clicking a document from the list will change appearance. The document is now displayed in an internal viewer on the tab (Fig. 36). A section for selected fields can be found to the left of the document, i.e. the document tab additionally applies the functions of a field tab here to enable direct comparison of the contents and field values.

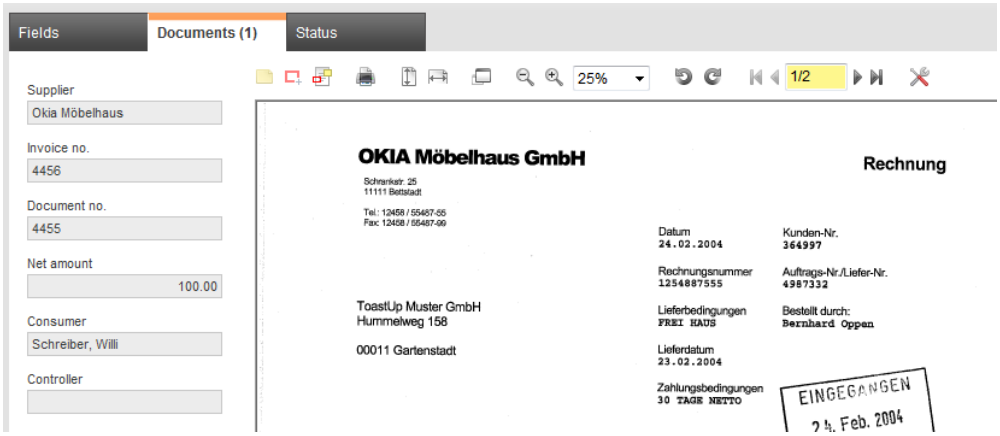

Fig. 36: Document in internal viewer

Documents or data files are saved in their original format on the server, so there are no limits with regard to their file format. So, all types of data files – from Microsoft Office documents through to digital images – can be saved in a DOCUMENTS file.

9.3.3 Link tabs

Link tabs replicate references between different DOCUMENTS files. Such a tab therefore provides a list of DOCUMENTS files linked or connected to the current file. Which other DOCUMENTS files are displayed on this tab is determined by filter criteria. These allow initially defining filters for file types or EAS sources. At a deeper level, other filters can be augmented with field values existing there, so not all DOCUMENTS files of the selected types are linked but only those where other content filters match. Fig. 37 shows an example from the *peachIT* principal: On files of the *Employee* type you can create *sick notes* (Sick certificate) and *vacation requests* (Vacation Application) for the relevant employee using a button. Both are independent file types which, however, are inevitably connected with the *Employee* files. On the *Employee* files, *sick notes* and *vacation requests* are therefore referenced on a link tab named *Absence*.

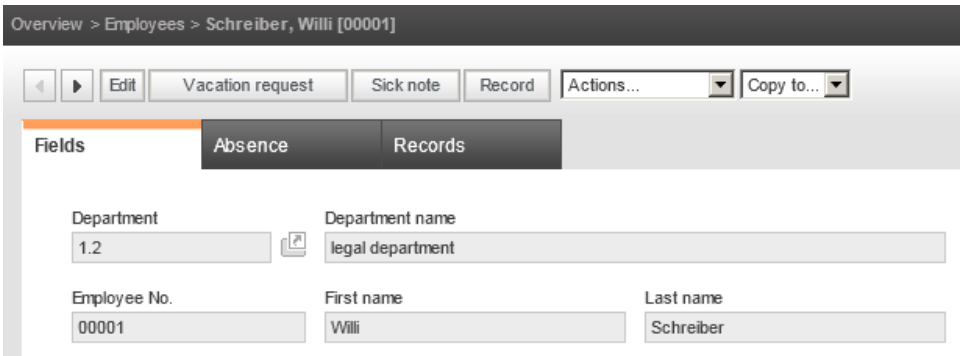

Fig. 37: Link tab on employee absence

Fig. 38 shows the referenced DOCUMENTS files of absence on the link tab. When you click on one of the entries, the referenced DOCUMENTS file will open in single view.

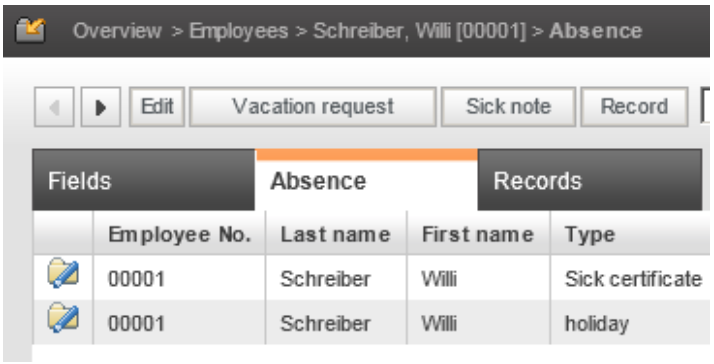

Fig. 38: Contents of a link tab

9.3.4 Early archived documents

This tab type can be used when an archive system is connected and the uploaded documents are stored directly in the archive, but not the related DOCUMENTS files. The latter will initially be retained as processes in DOCUMENTS 4 and in this tab override documents that have already been archived.

9.3.5 Tab access rights

Specific tabs can be equipped with separate permissions within a DOCUMENTS file. This is useful, for instance, when a file type is used by different addressees but specific information should, however, not be accessible to the entire group of people. Thus, for instance, sensitive documents might be entered in a separate document tab that can only be viewed by the directors in the *Employee* file type from the *peachIT* principal.

Such scenarios can be implemented using *tab access rights*.

To allocate such a permission, go to the corresponding tab and click the button to *Create new record*.

In the dialog shown in Fig. 39 lets you allocate permissions for any combinations of *user accounts* (including editors) and *access profiles* (in this example, *groups*). These are always assigned sequentially. So, access profiles with identically scheduled permissions cannot be authorized through multiple selection in a single step; instead, they need to be consecutively processed.

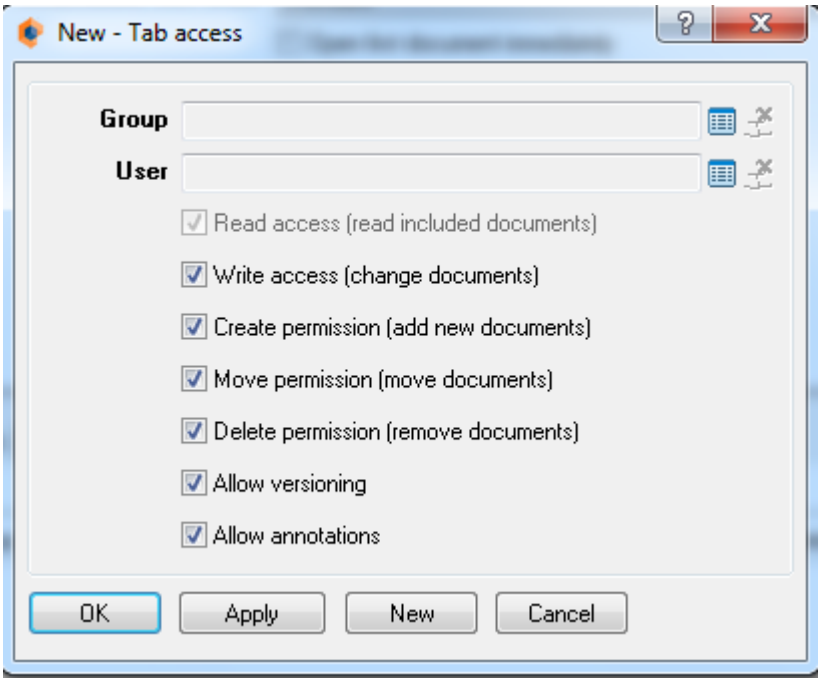

Fig. 39: The Access rights tab

Read access to the tab is always granted here. Other checkboxes allow granting the selected access profile or user additional permissions (e.g. modify, add, remove documents, etc.).

Overall, the principle of a *positive permission* applies here: Until permissions have been allocated, all existing access profiles and users have full permissions on the relevant tab. However, once a permission has been allocated, all permissions are withdrawn from all other access profiles and users. So, when starting to allocate permissions for a tab, this must be persistently completed for all existing access profiles and users. For this reason, the read permission cannot be de-selected. If a user should not be granted permissions to the documents of this tab, you simply allocate permissions for all other items, but not for this user.

9.4 Hit lists

When starting a public folder in DOCUMENTS 4 all DOCUMENTS files contained therein will be displayed in a tabular overview – of the *hit list*.

A hit list is implicitly already available for each file type; however, it cannot be accessed as a visible object. This is defined via the "*Show in hit list*" checkbox on individual fields. This implicit hit list therefore displays all fields in whose configuration dialog this checkbox is enabled. In this case, the order of fields within the hit list depends on the order of fields in file view.

Other hit lists can be created for each file type on the corresponding tab of the file type configuration. Such a hit list will then be configurable regarding selection and order of displayed fields.

It is created in the file type dialog on the Hit list tab as usual via the button on the left, initially opening the intermediate dialog shown in Fig. 40.

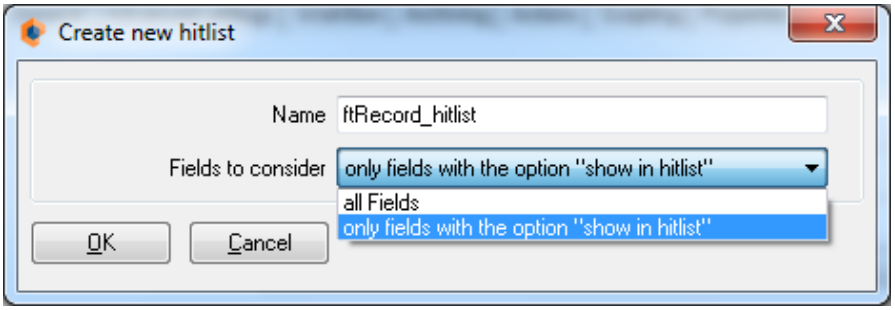

Fig. 40: Create new hit list

Give the hit list a name here and also choose whether *all fields* should be included or only those fields where the "*Show in hit list*" checkbox is enabled.

The latter case comes in conveniently when you have already made the right choice through the "*Show in hit list*" checkboxes on the individual fields but the order of fields in the hit list should differ from the order in file view.

After you exit this dialog, a new hit list is created which can be opened on the tab by double-clicking. Fig. 41 shows the configuration dialog. Besides the already allocated *name*, an optional multilingual *label* and a *description* can be added.

The hit list columns are managed in the bottom section. These correspond to the fields determined in advance and can be extended, deleted or put into a new order via the buttons. The order from top to bottom is, of necessity, transformed in DOCUMENTS 4, and therefore shown from left to right.

Deleting a field from a hit list only causes this field to be no longer displayed here. Whereas the actual field object is not deleted.

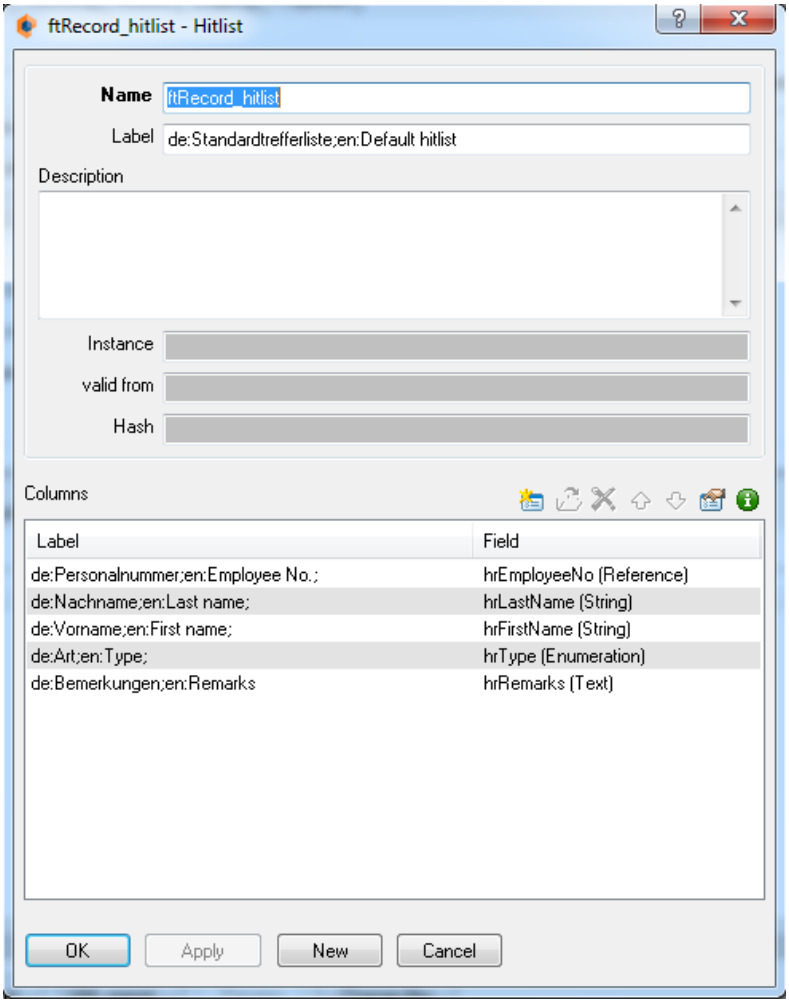

Fig. 41: Configuration dialog of a hit list

The same method can be used to create more hit lists which will then be provided for selection to DOCUMENTS 4 users. By default, the order of their arrangement on the file type's tab is also decisive for hit lists. It is always the (top) list that is used unless a user explicitly selects a different hit list.

Selecting a hit list is via global DOCUMENTS 4 search. When searching within a file type a user can influence how the search results are represented (see Fig. 42).

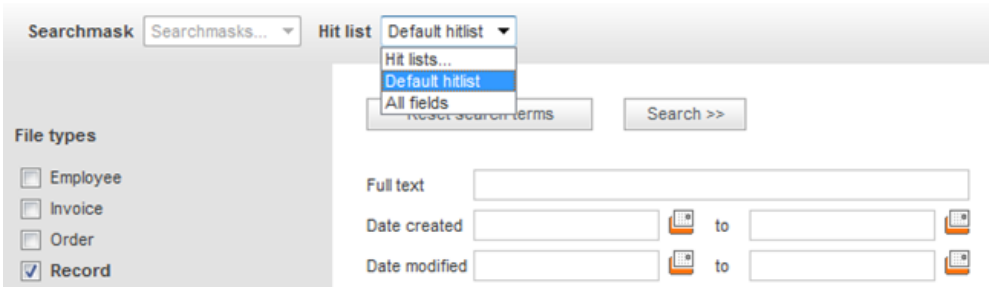

Fig. 42: Selecting a hit list

Creating different hit lists is of advantage particularly for complex DOCUMENTS files because it allows specific user or interest groups to optimize the search results to the fields relevant to them.

9.5 Search masks

Search masks provide a field selection that can be used as filter criteria for global search in DOCUMENTS 4. Appearance has been optimized for the respective field types. Numeric fields or date fields are, for example, automatically displayed with two fields for start and end to facilitate search within an interval. Fig. 42 in this context shows such a search mask for a selected file type.

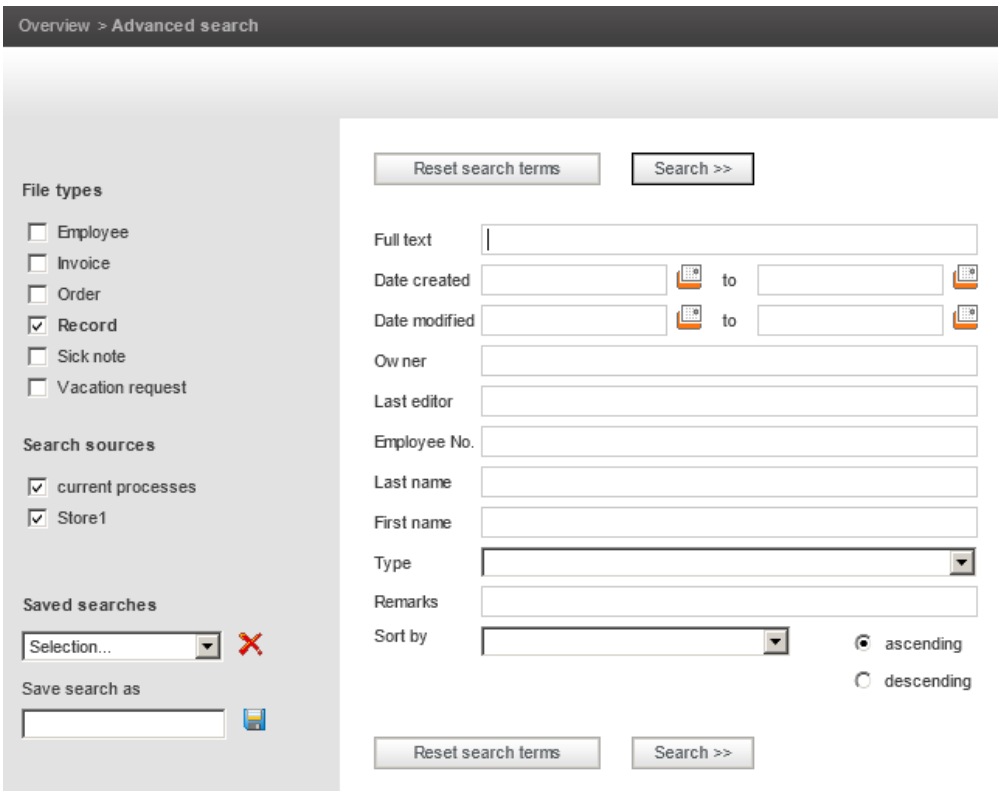

Fig. 43: Search mask including full text and fields as filter criteria

The properties of a search mask are similar to those of a hit list because these two elements can also be used together. The result of a search is always displayed in a hit list.

Analogous to hit lists, search masks are also objects that are defined for the respective file type. Also, each file type includes an implicit search mask that is determined by the "Show in search mask" property through its fields. So, if no other search mask is defined, all fields with the named checkbox enabled will be available for selection as filter criteria for the relevant file type in global search.

Creation of a search mask is performed on a file type's tab of the same name; it is identical to creating a hit list. This also implies that optimized search masks can be created for different interests and selected by the users, where selection and order of fields can be arbitrarily customized.

Additionally, a checkbox can be used to include another field for a *full text search*. Search terms initiated here are then searched throughout the DOCUMENTS file. Depending on the search method, the *full text search* can also be expanded to the full text indexed documents of the DOCUMENTS file. Fig. 44 shows a search mask with only two fields without the option of a full text search, while in Fig. 43 the full text search above the file fields is active.

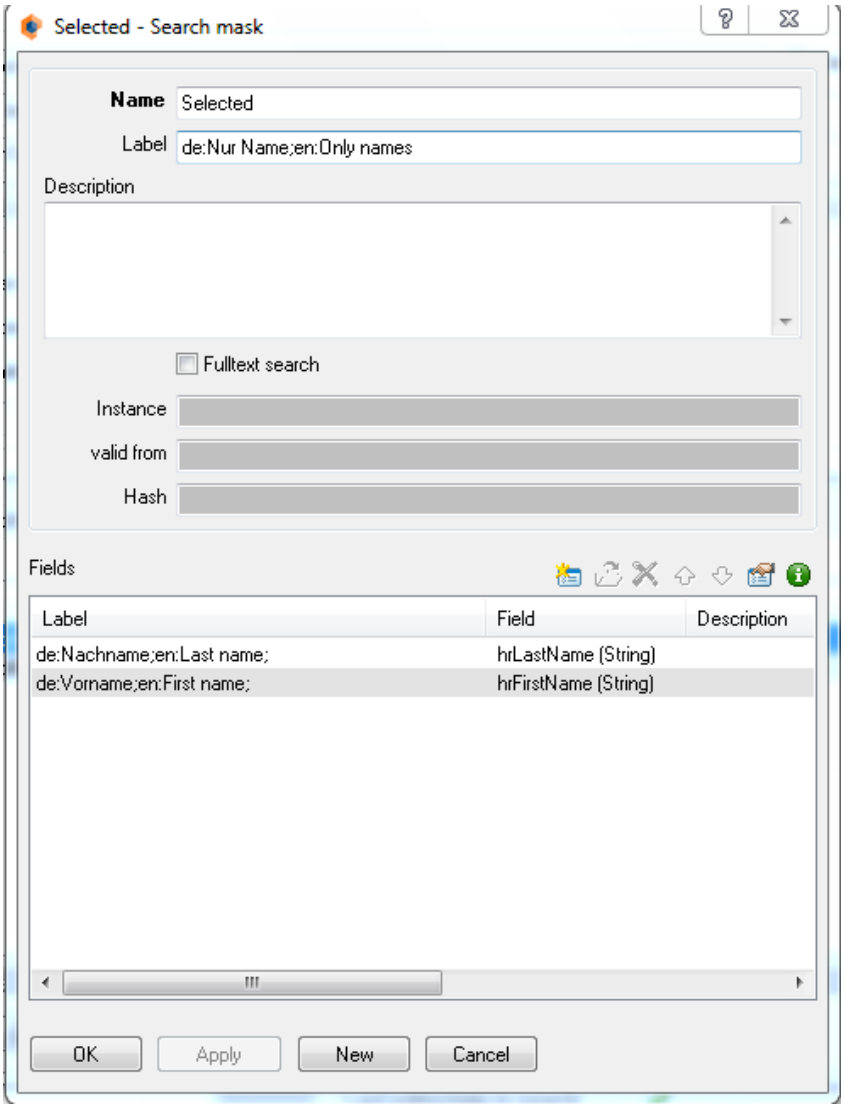

Fig. 44: Customized search mask

9.6 E-mail templates

Any *e-mail templates* whose contents are generated based on a template and which are sent either through user actions or specific events that occur can be defined for each file type.

The contents can be created as a combination of static text and values of existing fields. Moreover, *auto texts* can be used which provide various environment variables.

In the bottom section of the file type dialog, open the *E-mail templates* tab, and click the button to create a new template, whereupon the dialog shown in Fig. 45 opens.

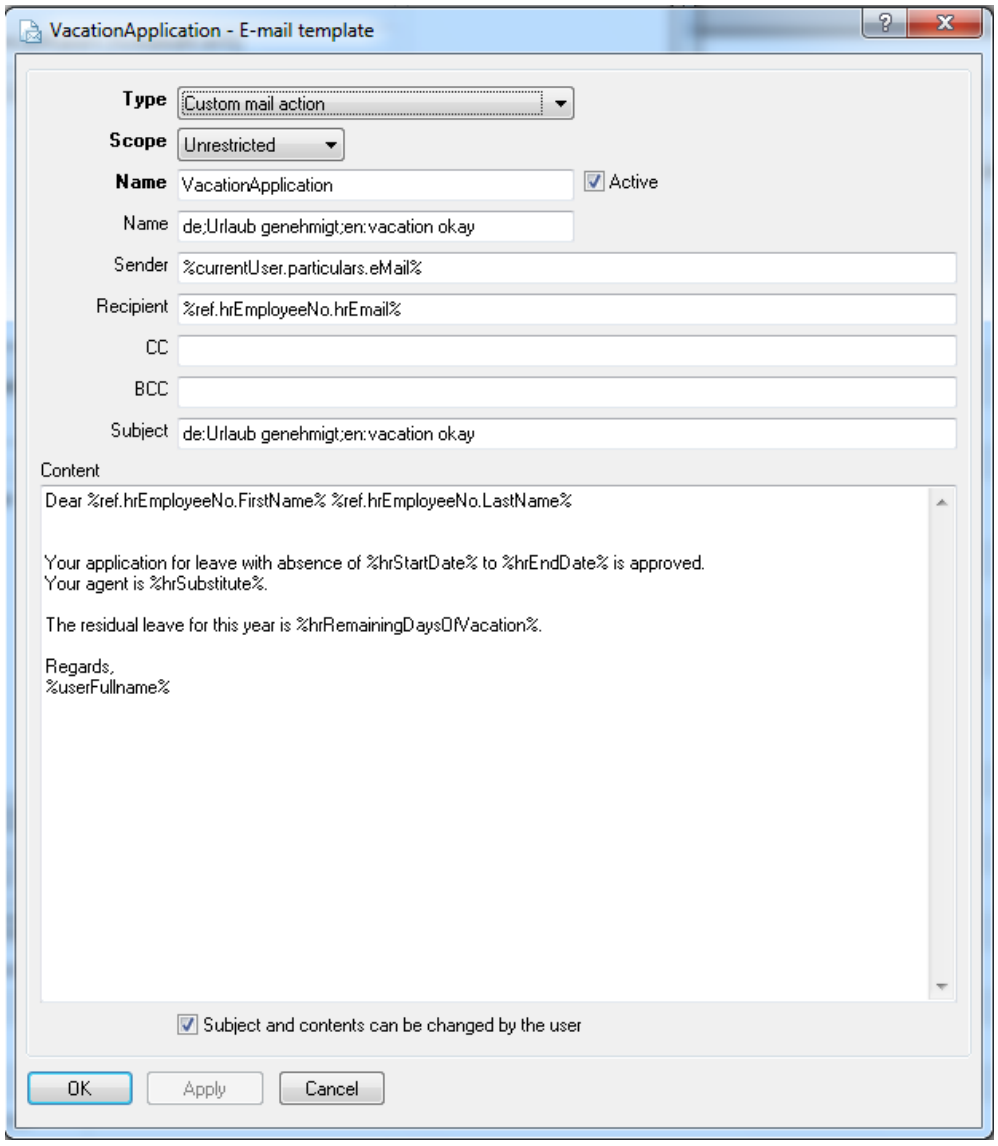

Fig. 45: Dialog for creating an e-mail template

The *Type* list contains five entries that define the respective action causing the e-mail message to be sent. In a *custom mail action*, the e-mail message is sent manually by the user. This is performed via the *Actions* list, which is extended with an entry for all DOCUMENTS files of the type that triggers sending the e-mail message. The name of the entry is in turn defined via the *Label* field.

The *Default mail action* is also started via the *Actions* list; however, it does not have a freely definable name, but it is submitted through the default command *Send*.

The other actions are automatically sent by DOCUMENTS 4 once specific events occur; they notify the user about their occurrence. These automatic events are:

- *New file in Inbox*: If a DOCUMENTS file is placed in a user's Inbox by sending it or via a workflow, the user will be notified via the mail template entered here. Only one template of this type can be defined for each file type. Using the auto text `%fileLink%`, which generates a hyperlink to the DOCUMENTS file, is useful here. If the user follows the link from the e-mail message, this DOCUMENTS file will open immediately after login.
- *File returned from routing*: The initiator of a routing action receives an e-mail message based on this template once sending the DOCUMENTS file has been completed. It is therefore not required that you observe the routing.
- *File during absence*: During active absence the original user is notified via this mail template about which DOCUMENTS files were edited by the selected delegates.

Sender and *recipient* (also applies to *CC* and *BCC*) can either be fixed addresses, or be inferred via field values or auto texts. An example of dynamic addresses is shown in Fig. 45 for the peachit demo. A custom e-mail template was generated for the `ftVacationApplication` file type which reads the recipient's mail address from the `ftEmployee` file type via a reference.

The message's *subject* line and its *content* also allow you to continue using field references or auto texts. Finally, you can also define whether subject and content of the e-mail message can be changed by users. Fig. 45 shows an example including static text, dynamic field values, references and auto texts for the `ftVacationApplication` file type. Finally, enable the new template by clicking the "*Active*" checkbox to make it available to the users.

9.7 Document templates

Comparable to *e-mail templates*, *document templates* provide the option to generate documents predefined in terms of contents.

When you create a new document template via the corresponding button, the dialog shown in Fig. 46 opens.

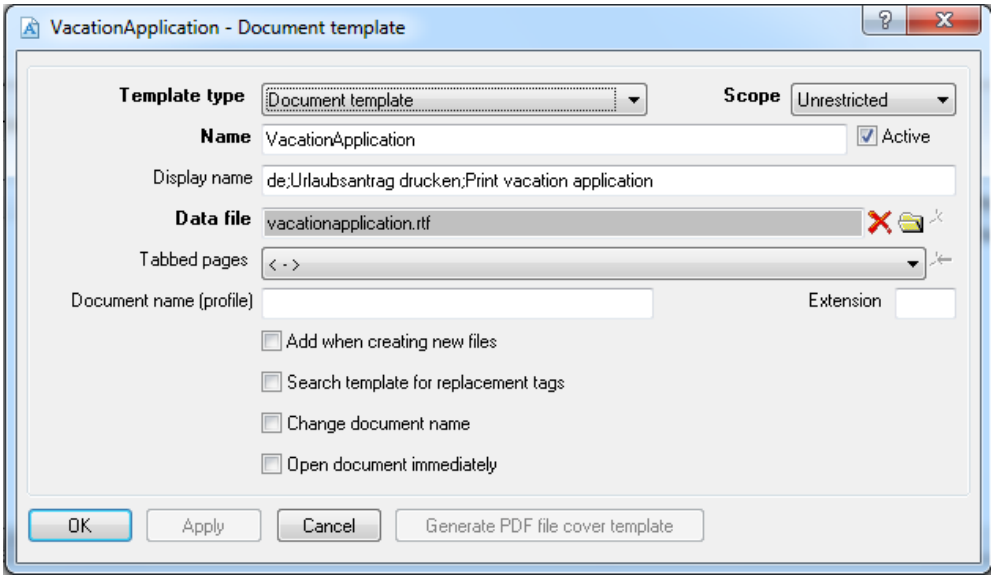

Fig. 46: Integrating a print template

When creating a new print template, selection of the *template type* is crucial. DOCUMENTS 4 provides three variants for selection here which fulfill significantly different purposes:

- *PDF cover template*: This type enables generating a PDF document from a DOCUMENTS file. This PDF document is at least composed of an overview of fields as printable file covers. Optionally, attached documents can also be integrated.
- *Print template*: Templates of the *Print template* type generate documents based on a predefined `rtf` file. When a user starts this action in DOCUMENTS 4, a document is created and opened immediately so it can be printed.
- *Document template*: Like the *print template*, a document is also generated based on an `rtf` file here. Unlike the print template, however, this will not be opened but saved in a document tab of the DOCUMENTS file.

9.7.1 PDF file cover templates

Specify name and label first, and then click the *Generate PDF file cover template* button. The DOCUMENTS Manager automatically creates a file named `cover.xml`. This includes a tabular structure of all file fields and can be individually customized. *Enable* the template using the checkbox.

When a user starts the related action, a *PDF document* with the structure from the `cover.xml` file will be generated.

Optionally, attached documents of the relevant DOCUMENTS file can also be integrated with the generated PDF file. When starting the action the users can select in a dialog whether to generate the PDF only from the field values or also from the documents (Fig. 47).

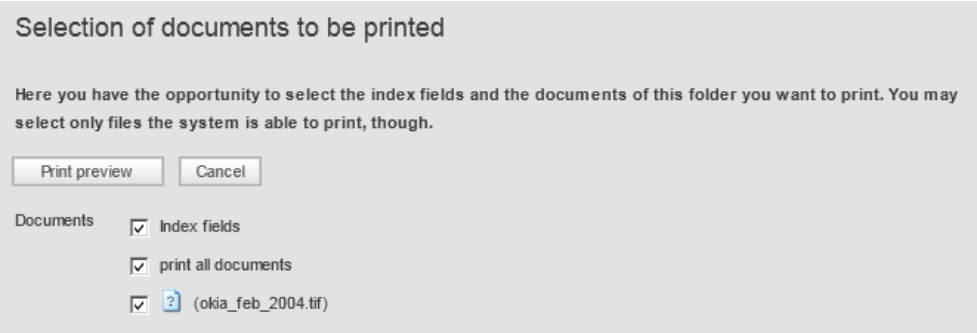

Fig. 47: Selection of documents to be printed

To do this, however, it is assumed that, depending on the file type of the attached documents, an appropriate PDF converter is installed.

In principle, this template type can be used to combine the entire content of the DOCUMENTS file, including all documents, into a single PDF file.

9.7.2 Print and document templates

Print and document templates facilitate, based on a predefined text template, generation of complete documents in `rtf` format, for example. These represent a combination of static text passages and the values from existing fields or auto texts.

Both *template types* are by definition also similar in their use. Their behavior differs only in providing the generated document: After generation, a *print template* immediately opens in a window of the assigned text editor, and can be printed, edited or saved. Whereas a *document template* does not open but is added to a fixed *document tab* of the relevant DOCUMENTS file.

It is true for both template types that the template is initially created in an editor or text processor and that a text file (e.g. `txt`, `rtf` or `docx`) must be created there. Binary formats such as Microsoft `doc` format are not supported.

These templates let you reference the values of existing fields or revert to auto texts. The documents are then automatically populated with these values, which largely eliminates the need for the manual process of transmitting the field values to documents. Documents based on a template are created by a user at the touch of a button and are automatically generated, where users can subsequently edit and change them.

Basically, notation on using field values or auto texts works as with e-mail templates, where the type of template specifies an essential difference here: While subject and content of an e-mail template can be used directly in the DOCUMENTS Manager due to the simple structure, the document must initially be created and then uploaded to the DOCUMENTS Manager for the use of a document template (see entry in the *File* field in Fig. 46).

Document templates can be created in any application that supports the file format.

Integrating *field value* is performed as usual by enclosing the respective field name in single or double percent signs (%%). Fig. 48 shows an example of a document template for the *Vacation request* file type from the peachit scenario. The inserted graphic is representative of the option to implement a corporate design in these templates.

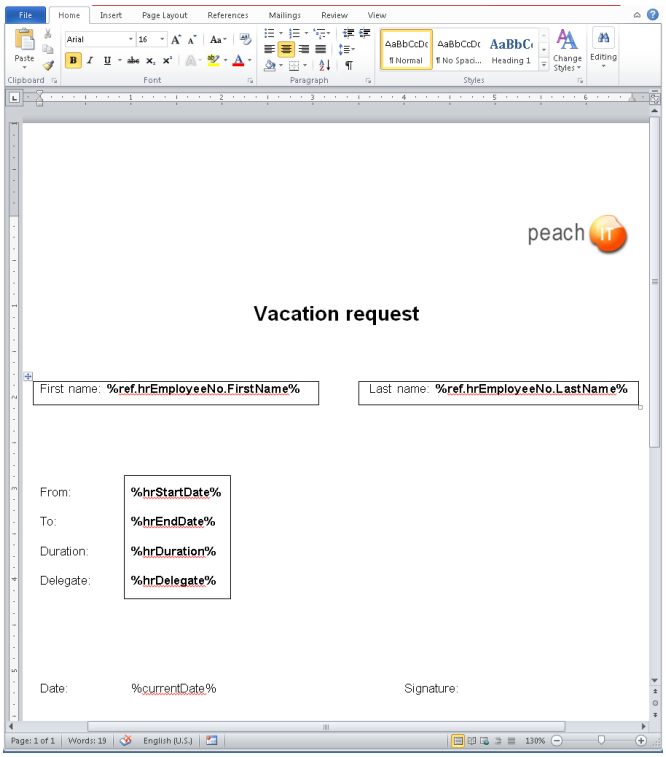

Fig. 48: Example of a document template

The generated document then contains the respective field values, while the original template remains unchanged, and can therefore continue to be used.

Fig. 49 shows the Create dialog in a sample *document template*. Here the template must be integrated as a *data file* first. Moreover, for *document templates* you need to select a *tab* of the file type in which to store the generated document. This entry is redundant for the template type *Print template*.

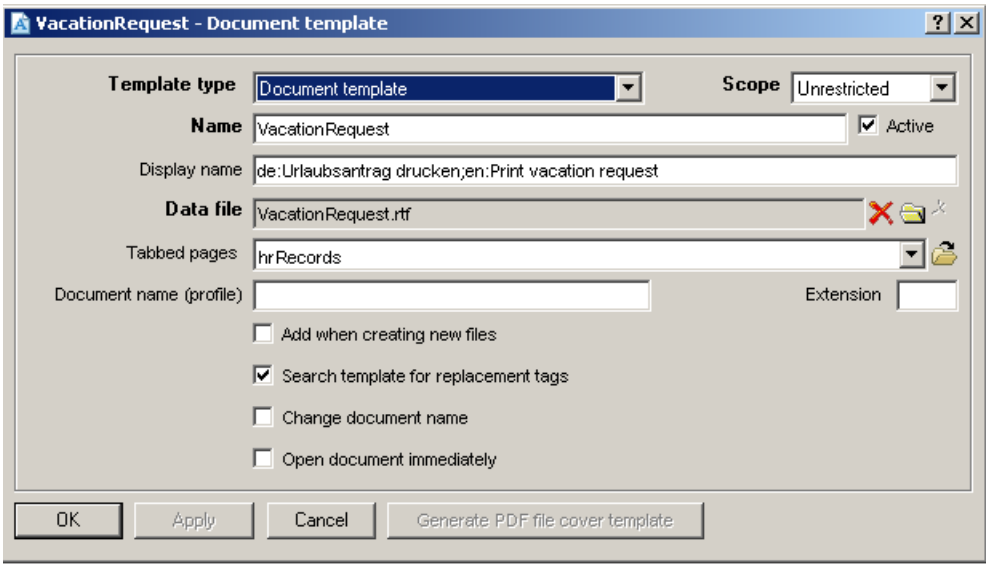

Fig. 49: Creating a document template

Optionally, you can enter a *default value* for the *document name*.

Moreover, other options can be exclusively selected for the *Document template* type via checkboxes:

- *Add when creating new files*: A document based on the template is automatically generated once a new DOCUMENTS file is created.
- *Change document name*: After generation the user is prompted in a dialog to specify a suitable name for the document.
- *Open document immediately*: This option combines the document template and print template functions: After generation the document is saved in the DOCUMENTS file and opened immediately.

> *The* Search template for replacement flags *checkbox is crucial for the functionality of both template types: Only when this is enabled will field values and auto texts be resolved into the corresponding values. If this checkbox remains disabled, any interpretation will be redundant and the values in "percent notation" will be treated as static texts.*

9.8 Custom actions

Commands can be made available on the relevant file type via *Custom actions*. These will then be available in DOCUMENTS 4 in each DOCUMENTS file of this type. Optionally, this can be performed in the form of a function button or as an entry in the *Actions* list. Fig. 50 shows a DOCUMENTS file of the *Employee* type from the *peachit* principal. Different user-defined actions are in use here. Thus, for instance, directly from the employee file you can create new DOCUMENTS files of the *Vacation request*, *Sick note* or *Record* types which are connected to the employee file. Context viewing, of course, suggests that a vacation request, for example, is created only for the relevant employee.

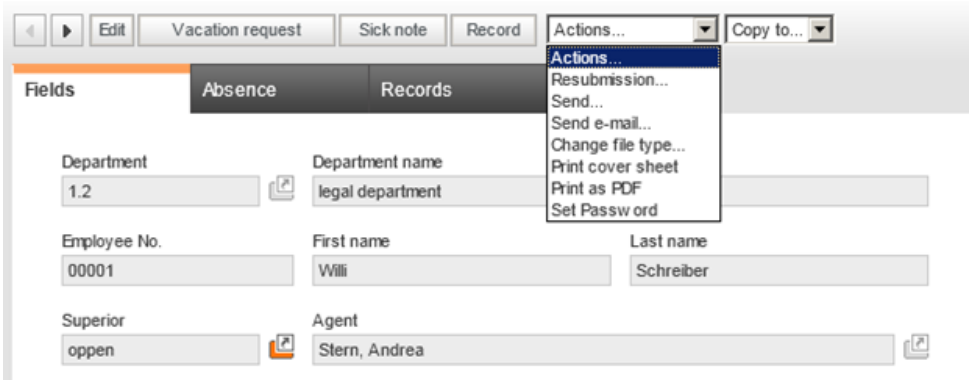

Fig. 50: Custom actions of a DOCUMENTS file

The cascaded Actions list shown in Fig. 50 includes another custom action named *Set Password* at the very end. Unlike the other actions, this does not create a new DOCUMENTS file but executes a script that resets the user account's password.

We will pick up this example of a *Portal script* again in chapter 9.13 (*Scripting*).

Fig. 51 shows the *Employee* file type in the DOCUMENTS Manager including the custom actions reflected in Fig. 50.

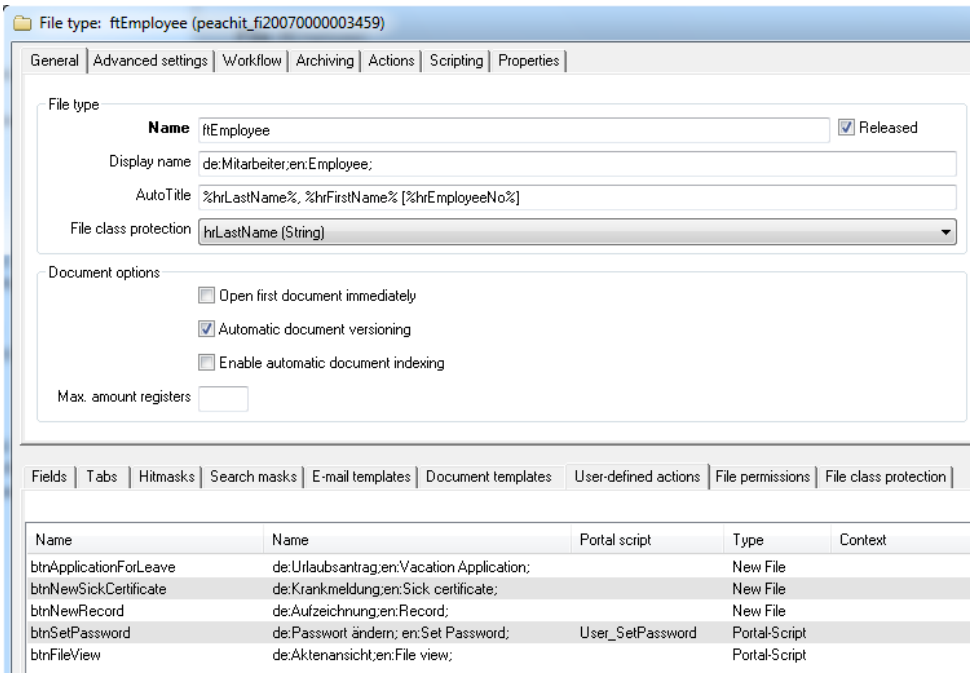

Fig. 51: Custom actions of a file type

Fig. 52 shows the dialog for creating a custom action (again, an example from the *peachIT* principal).

Specify a *name* and a *label (name)* here first. The *interactive element* lets you choose whether the action should be reached via a *function button* or an entry in the *drop-down list* (actions).

To select the *type*, the *New file* and *Portal script* variants have already been introduced in this example. Therefore, you can optionally also use a *JavaServerPage (JSP)* or a *gadget*.

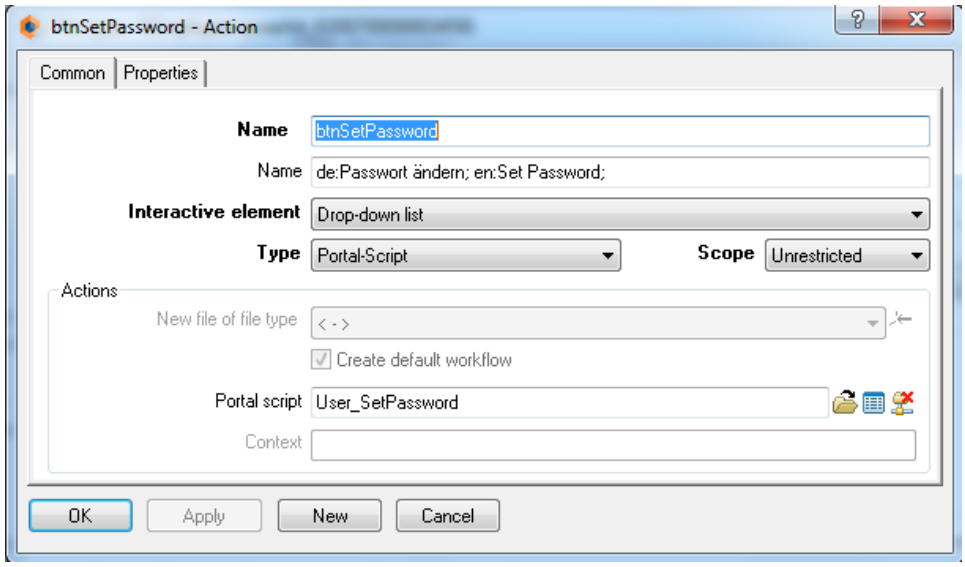

Fig. 52: Custom action for creating a new DOCUMENTS file

For the *New file* type you need to determine another field indicating the *file type*. Analogous, the script library is used for selection for the *Portal script*.

A special feature emerges with the use of the *New file* type when both file types are connected to each other via a reference field (see chapter 9.2.6): In this case, the reference is automatically set; the user does not have to add this. In the above example, for instance, a reference to the Employee file type has been created. When a user views an employee file creating a new vacation request from this context, DOCUMENTS will automatically populate the reference field on the employee file there, because it is clear from the context of new creation that the vacation request should obviously be created for this employee.

9.9 File permissions

Contents of DOCUMENTS files of a type can be restricted via *File permissions* to limit specific tasks and permissions. To do this, go to the corresponding tab and click the *Create new file* button.

The dialog shown in Fig. 53 lets you allocate permissions for any combinations of *user accounts* (including editors) and *access profiles* (in this example, *groups*). These are always assigned sequentially. So, access profiles with identically scheduled permissions cannot be authorized through multiple selection in a single step; instead, they need to be consecutively processed.

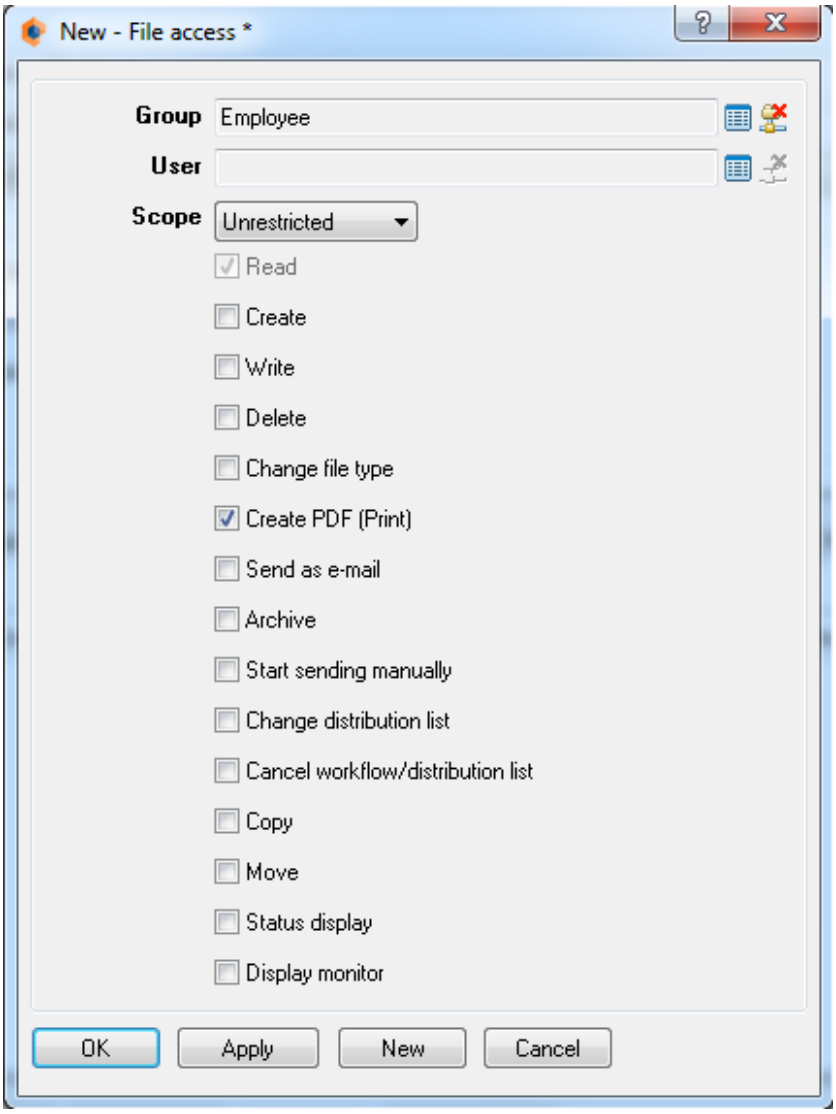

Fig. 53: Access rights of DOCUMENTS files of a type

Read access to a DOCUMENTS file is always granted here. Checkboxes allow you to decide whether other permissions should be additionally granted to the selected access profile or user. Moreover, with regards to the *scope* you can determine whether these permissions should apply to *processes*, *archived files*, or both (*unrestricted*).

Overall, the principle of a *positive permission* applies here: Until permissions have been allocated, all existing access profiles and users have full permissions on the files of this type. However, once a permission has been allocated, all permissions are withdrawn from all other access profiles and users. So, when starting to allocate permissions for a file type, this must be persistently completed for all existing access profiles and users. For this reason, the read permission cannot be de-selected. If a user or access profile should not be granted any permissions on the processes of this file type, you simply allocate permissions for all other items.

9.10 Advanced settings

On this tab, different display options for the file cover as well as search and hit list display are configurable. DOCUMENTS 4 saves different information for each file regarding creating and editing files (e.g. owner, last editor, and timestamp). Here you can define which of these data items should be visible. Removing specific options, however, does not have an impact on the fact that this data is saved.

Additionally, you can display the *Status* tab. This logs specific actions of the DOCUMENTS file as well as its time and the user. These actions, for instance, are uploading or deleting documents, as well as generally editing a DOCUMENTS file. Field value modifications are also logged on this tab. If this is desired, you can enable this on a case-by-case basis on the individual fields.

You can finally define an *icon* for the *file cover edge*, as the example in Fig. 54 shows. The related GIF graphic must be stored in the DOCUMENTS 4 installation directory. The exact storage location including a catalog of existing graphics reads:

```
..\Documents4\www\images\dlc\compact\margin
```

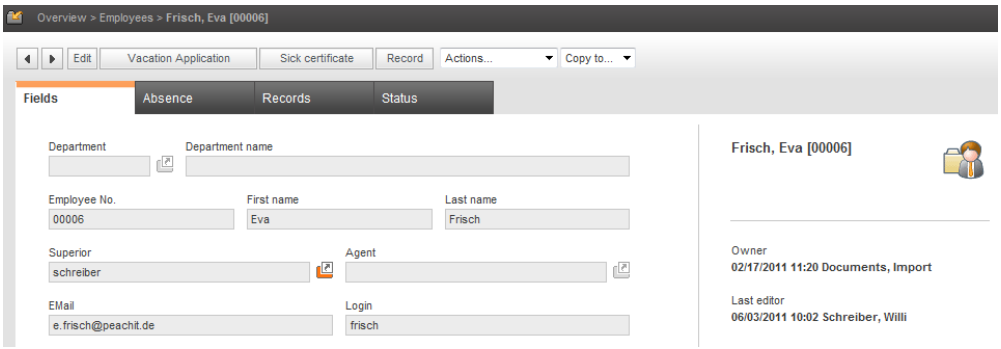

Fig. 54: DOCUMENTS file with icon at file cover edge

9.11 Workflow

If you want to enter a standard workflow for a file type, you can select this workflow or distributor here and also determine that this should be started immediately on creating a new DOCUMENTS file. In this way you ensure that a checked process starts for each DOCUMENTS file while it is being created.

Additionally, a *monitor tab* can be shown which represents information on the process flow. In this context, various options can be controlled via checkboxes.

9.12 Actions

This tab contains a host of checkboxes used to generally control which default *actions* that are available may be performed for the relevant file type. For each action, moreover, you can define whether its execution should be logged in the *Status* tab.

9.13 Scripting

Java scripts can be executed for specific events as part of editing to execute complementary functions for the DOCUMENTS file. The allowed events have already been defined; they represent actions executed by the users (e.g. create a new DOCUMENTS file, edit or save it). You can select a Java script for the respective event which must be created prior to this.

For more information on scripting options in DOCUMENTS 4, please refer to chapter 13.

9.14 File class protection

Access rights to DOCUMENTS files can also be linked to the content of an individual file field. There are three variants of this access protection which are commonly administered within the DOCUMENTS Manager as *file class protection*. They can be briefly described as follows:

- Ordinary file class protection restricts access to DOCUMENTS files with a specific field value to a subset of available users assigned to this value.
- An access list (ACL) saves a list of names of all users with read permissions in that field.
- A group access list (GACL) saves a list of all access profiles with read permissions in that field. Mixed lists of access profiles and user names are also allowed.

File class protection is determined by selecting the corresponding filter field on the *General* tab. Once you have selected a field here, another tab named *File class protection* appears for the file type on the bottom tab bar. You need to further specify the protection here.

Folder rights check is automatically turned off for file types including ACLs and GACLs.

9.14.1 Access Control Lists (ACLs)

Access Control Lists (ACLs for short) or *Group Access Control Lists* (GACLs) are an extension to the file class protection concept. These allow defining visibility (searchability) of a DOCUMENTS file depending on the logged-in user or their group membership (access profile). Unlike the global permissions definition of the file type, which then applies to all DOCUMENTS files of that file type, the visibility permission for (G)ACLs refers to individual DOCUMENTS files. For this, the authorized users and access profiles are entered in an index field.

The option to define file class protection of this type is available for:

- File types
- EE.i archive types using the XML server

> *Note: When using EE.x as the archive system, the file class protection available in EE.x must be used.*

With ACLs the *login* names of the authorized users, or with GACLs, the *access profile* names of the authorized groups, are written in an index field. Only authorized users can search the corresponding DOCUMENTS files or view them in a folder.

To use ACLs for file class protection, in the file type or in the archive file type, you create a field of the *Double selection list* type first.

Enter the `%login%` auto text as enumeration values. Thus, users are now available as a selection in the double selection list.

The creator of the DOCUMENTS file must be entered when creating a new file, so that they can actually edit it. Therefore, you need to define the default value (value/default setting) `%userLogin%`. Fig. 55 shows a sample configuration of such a field.

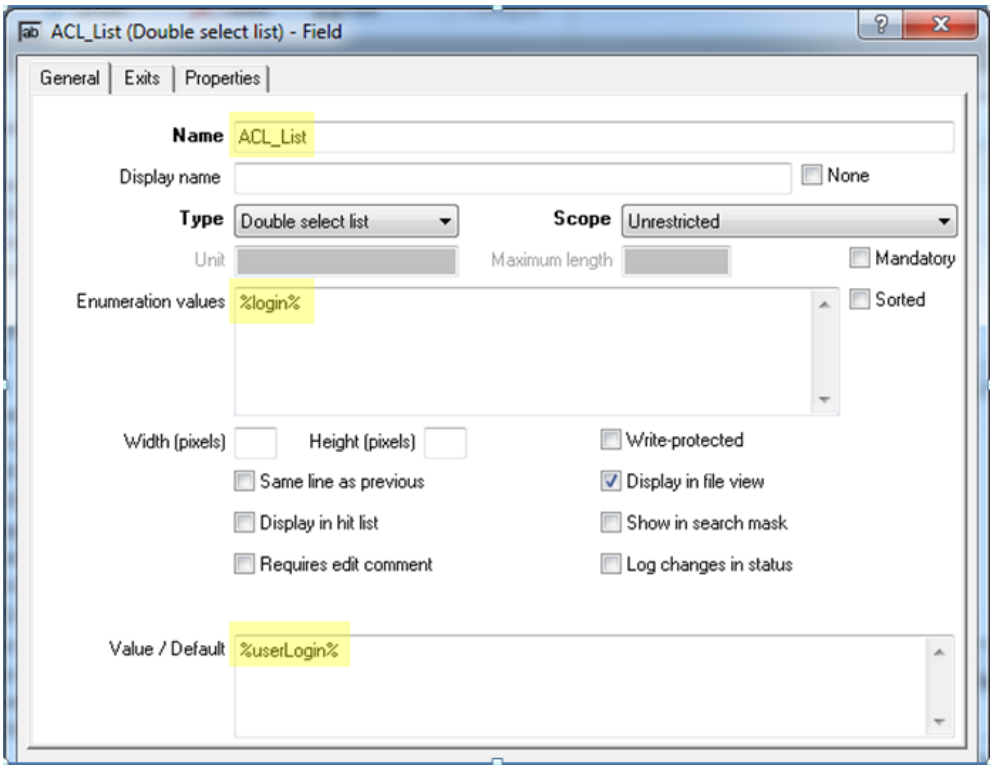

Fig. 55: Field for ACL use

Now define the field you have just created as file class protection for the file type, and apply your change ().

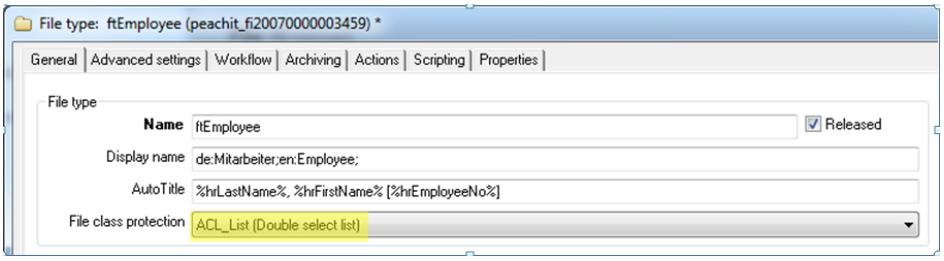

Fig. 56: Setup file class protection

By assigning file class protection, a new entry named *File class protection* is displayed in the bottom tab bar on the file type's dialog.

Create a new *file class protection profile* using the class name `ACL` (Fig. 57). This class name is required for ACLs. The field value of this ACL must then be `%userLogin%`. The definition of an access profile is not applicable.

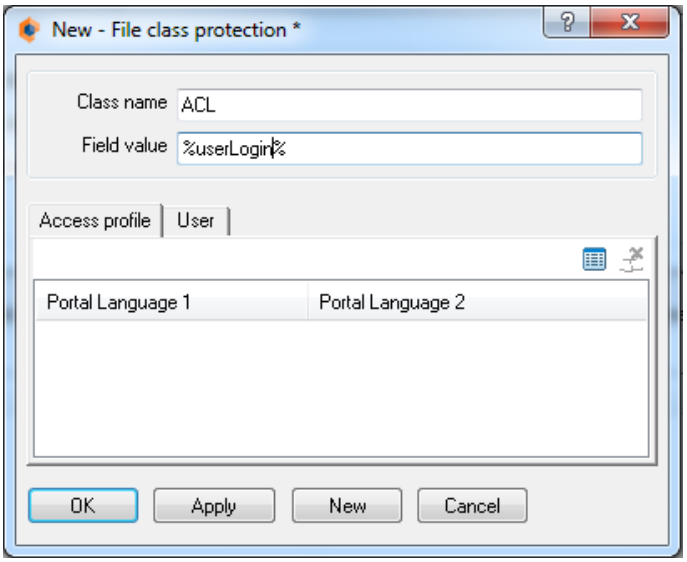

Fig. 57: New ACL with field value %userLogin%

The ACL configured in this way now impacts on public folders and while searching in that only users authorized in the ACL can retrieve and start the DOCUMENTS files. To all other users it appears as if these DOCUMENTS files did not exist.

9.14.2 Group Access Control Lists (GACLs)

Analogous to the above ACL, another extension to the file class protection is available: GACL. While ACL is limited to individual users, group ACL is suitable for access profiles. When you enter a GACL for a file type and store DOCUMENTS files of this type in public folders or in the archive, for example, only the DOCUMENTS users that have been selected on creating or processing the DOCUMENTS files for the GACL can access these DOCUMENTS files. You can implement this most conveniently via a double selection list. All other users do not have any access to these DOCUMENTS files.

To use GACLs for file class protection, proceed analogous to the example of ACL:

On the (EE.i archive) file type, create a field of the *Double selection list* type. Enter the `%accessProfile%` auto text as enumeration values. Thus, all access profiles created in the system (on the principal) are displayed in the left pane of the double selection list, and can be selected at a later time.

So that the user can create a DOCUMENTS file and is also granted access to a DOCUMENTS file they created, it is absolutely necessary that you define, as *value/default setting*, the `%accessProfilesGACLList%` auto text. This auto text is used to automatically select the access profiles of the creating user in the double selection list.

Now define the field you have just created as file class protection, and apply your change.

Following this, on the File class protection *tab*, create a new file class protection profile with the class name "`GACL`". This class name is required for GACL. The field value of this GACL must then be `%accessProfilesGACLFilter%`. This gives only the users with an assigned access profile that has been selected in the double selection list access to these DOCUMENTS files.

The GACL configured thus now affects public folders and while searching in that only users authorized in the GACLs can retrieve and start the DOCUMENTS files. To all other users it appears as if these DOCUMENTS file did not exist.

> *More information on ACLs is available in separate documentation.*

9.15 The "Change files" function

You always need to use the Change files function when changes have been made to the structure of the file type and files (processes) of this type already exist at the same time. For more information on the absolute necessity of this function, see chapter 17.

10. Public Folders

Public folders are displayed in the DOCUMENTS 4 tree structure; basically, they perform three different functions that are controlled through their *folder type*. Fig. 58 shows such a folder view in DOCUMENTS 4 at different levels. The *Personnel* folder itself, for example, does not contain any data; it is only used as a grouping element for the individual subfolders which in turn display all DOCUMENTS files of the *Employee*, *Absence* or *Recordings* types.

In the *Invoices* panel, DOCUMENTS files of the *Invoice* type are listed in all subfolders where, however, these are using a *dynamic filter* and only display DOCUMENTS files that have the corresponding criterion in a *Filter field* (*incoming*, *assigned*, etc.).

In Fig. 58 the *Employees* folder is highlighted. All contained DOCUMENTS files are listed in the right pane. Selection and arrangement of individual columns is defined here via a hit list (see chapter 9.4).

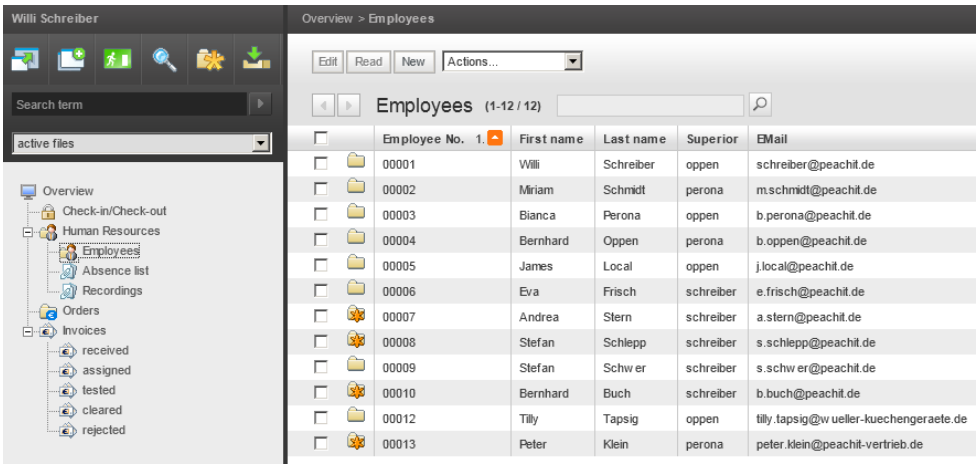

Fig. 58: Public folders in DOCUMENTS 4

To create a public folder, open the tree section *Documents* and select the *Public Folders* entry. Then, in the menu bar, click the *New* button. Fig. 59 shows the example of a folder from the demo principal *peachIT*.

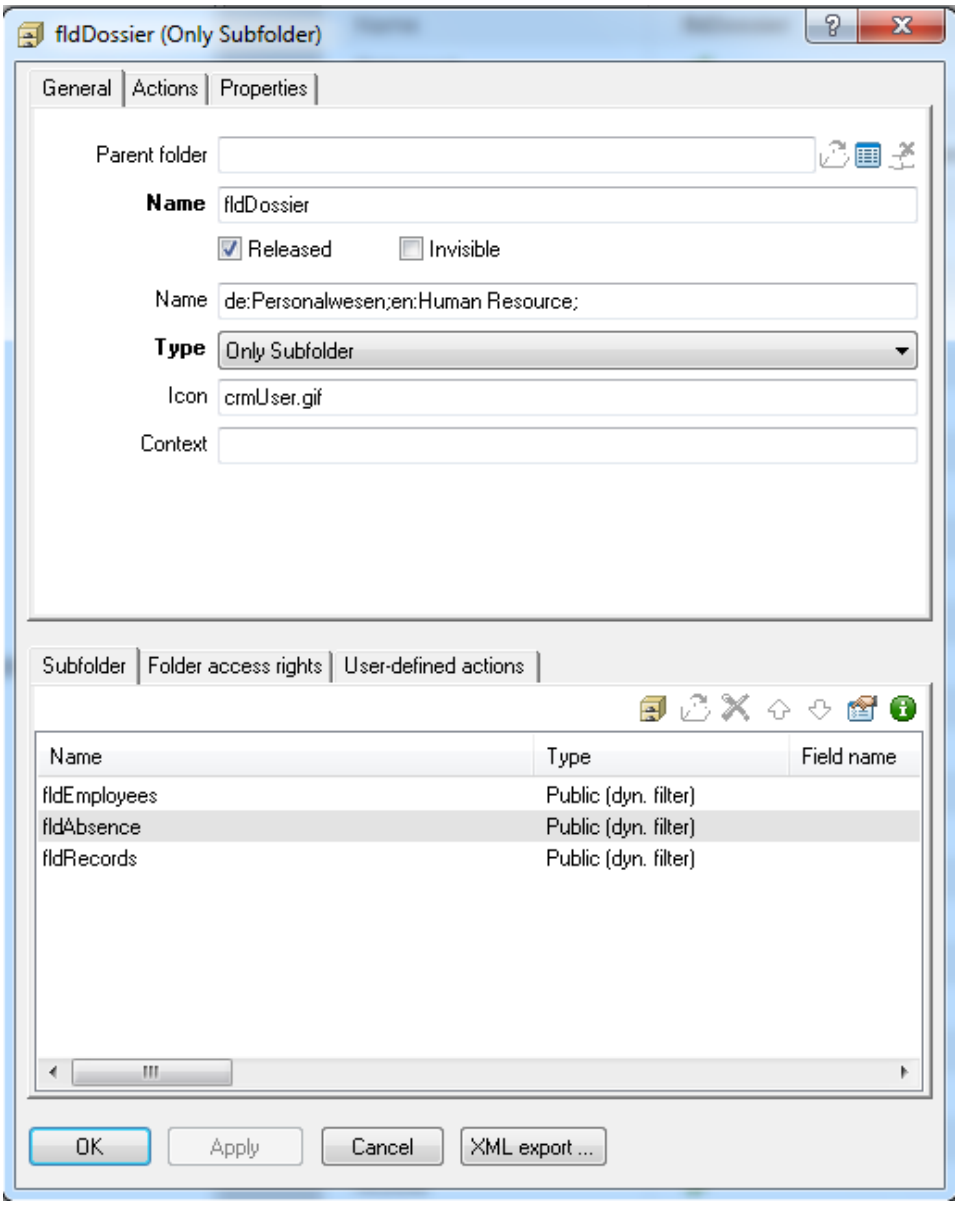

Fig. 59: Public folder

First, specify a technical name for internal use, as well as an ergonomic label. This can optionally be named in multiple languages; it is used as a visible label for the DOCUMENTS 4 users.

If the new folder is to be displayed in an existing structure at a deeper level, you can optionally select a *parent folder*.

Enable the *Released* checkbox to make the folder available to the users.

Finally, define the *type* of the folder to be created. Depending on this selection, new tabs are added to the dialog, if necessary.

10.1 The "Public" type

DOCUMENTS 4 lets users move selected DOCUMENTS files via the selection list *Copy to* to folders of the *Public* type by a user (Fig. 60).

Moreover, you can store DOCUMENTS file in such a folder within a distribution list or as part of workflows.

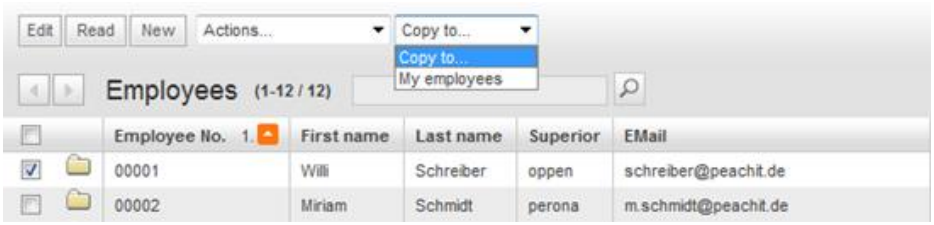

Fig. 60: Copying a DOCUMENTS file to a public folder

10.2 The "Only subfolder" subtype

Folders of this type are used to better structure the tree view in DOCUMENTS 4. A folder of the *Only subfolder* type does not contain any DOCUMENTS files itself. For this folder, however, you can create other subfolders which in turn can display each of the three folder types.

Creation of other subfolders within a folder is performed in the *DOCUMENTS Manager* via the *New* icon on the bottom tab named *Subfolders*.

Fig. 61 in this context shows a folder named *Human Resources*, which contains the *Employee*, *Absence list* and *Recordings* subfolders. You cannot store DOCUMENTS files in the *Human Resources* folder itself, but in the three subfolders. When you click this folder, only the subordinate section will be cascaded or collapsed.

Fig. 61: Folders with subfolders

10.3 "Public (dynamic filter)" type

This folder type represents a specified search. Thus, for instance, all DOCUMENTS files of a file type that satisfy a specific filter criterion can be displayed in this folder. Tracking status of DOCUMENTS files of a specific file type within a workflow is a frequent use case ("*new*", "*edited*", etc.). In the *peachIT* sample you can find this type of division for the *Purchase invoices* file type.

Public folders with dynamic filters contain a host of other tabs where you can define the desired filter criteria.

Here you initially determine a filter for *file types* or *EAS sources* to generally define which types of DOCUMENTS files should be displayed in this folder. Optionally, you can define *Queries* in the next step. This ensures that only those DOCUMENTS files that match the criteria defined there are displayed. Fig. 62 in this context shows the *audited invoices* folder from the *peachIT* principal. Here the *Filter (file type)* is initially limited to *Invoices* and then defined on the *Query* tab, so this folder may only display invoices whose `status` field has the value `tested`.

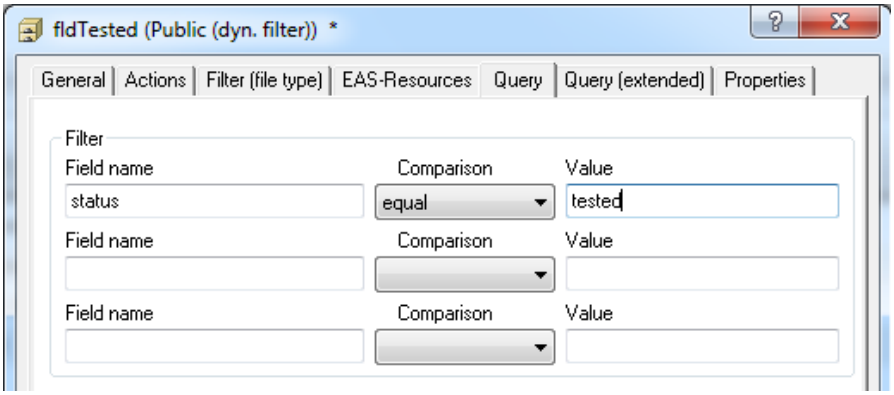

Fig. 62: Filter criterion of a folder

Fig. 63 in this context shows the definition of the related file type to clarify the structure of the filter.

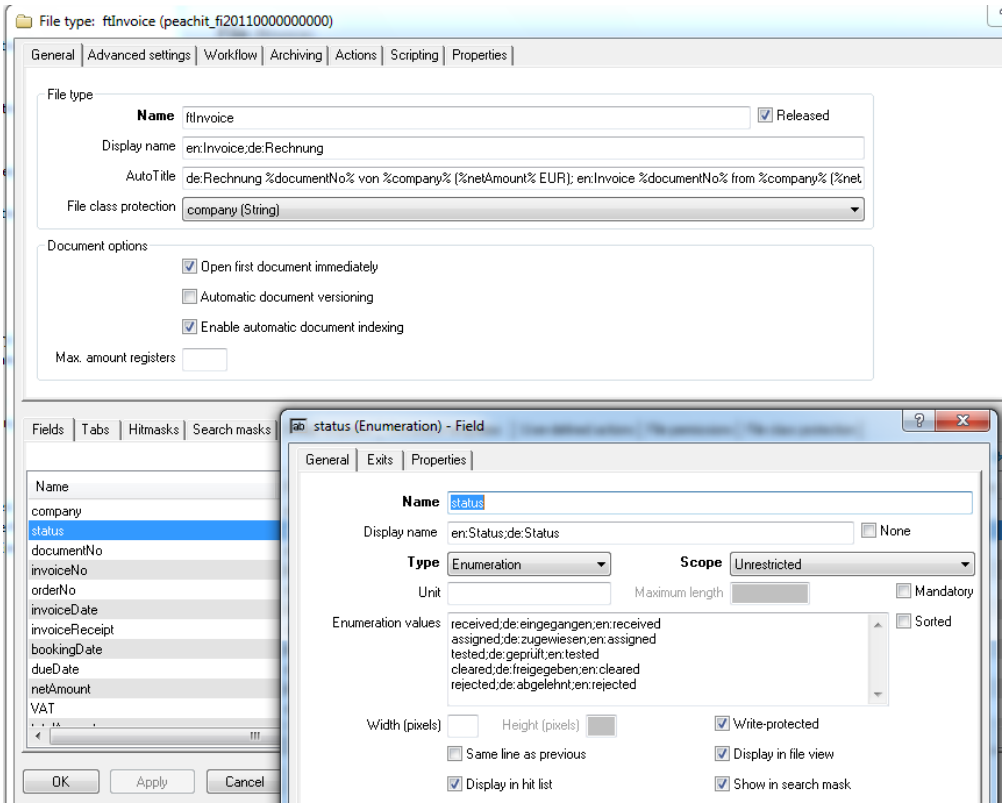

Fig. 63: File type definition of filter criteria

10.4 Custom actions

The *custom actions* allow making commands available for the relevant folder. In this way, for instance, scripts can be executed or new DOCUMENTS files can be created. Providing at least one public folder with dynamic filter for each file type which centrally lists the processes is generally useful. Moreover, this folder should have a custom action for creating a new DOCUMENTS file of this type. In Fig. 64 you can see an example from the *peachIT* principal. In this folder, a custom action is provided on the public folder *Employees* which enables creating a new employee file via a function button.

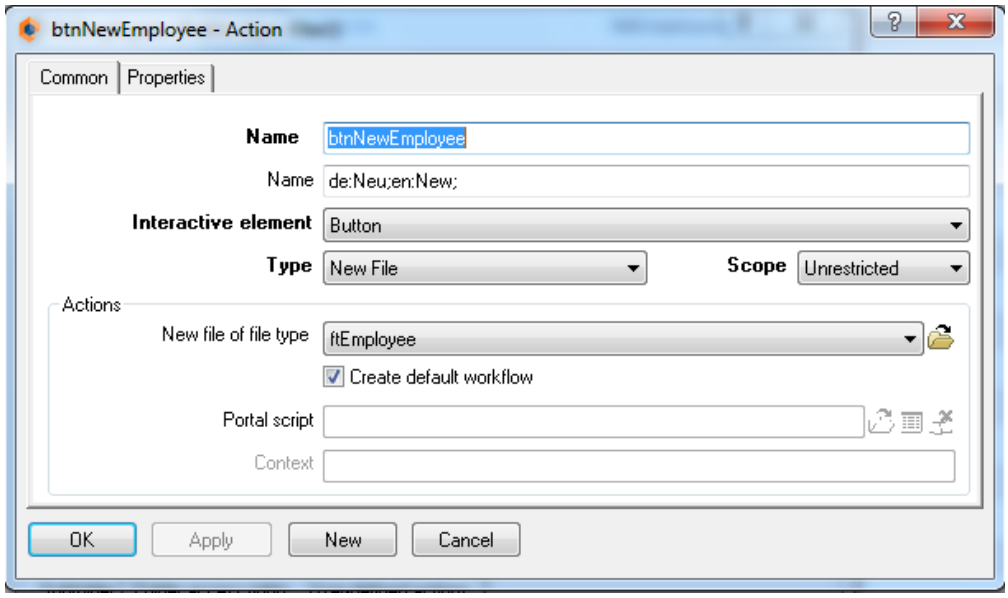

Fig. 64: Custom action for creating a new DOCUMENTS file

10.5 Folder access rights

Accesses to folders can be restricted via *folder access rights* to make their contents editable only by specific groups of people. To do this, go to the corresponding tab and click the *Create new record* button.

The dialog shown in Fig. 65 lets you set permissions for any combinations of *user accounts* (also editors) and *access profiles* (in this case, *Groups*). These are always assigned sequentially. Access profiles with identical permissions cannot be authorized through a single step by multiple selection; they need to be processed one after another.

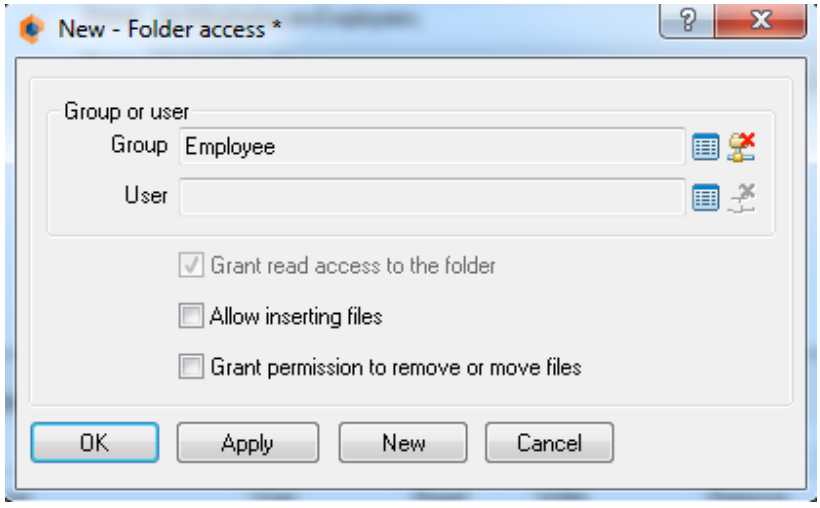

Fig. 65: Read folder access right

Read access to the folder is always granted here. Checkboxes allow you to decide whether the selected access profile or user should be additionally granted the

permissions to *Insert new DOCUMENTS files* or to *Remove or move DOCUMENTS files*.

Overall, the principle of a *positive permission* applies here: Until permissions have been allocated, all existing access profiles and users have full permissions on the relevant folder. However, once a permission has been allocated, all permissions are withdrawn from all other access profiles and users. So, when starting to allocate permissions for a folder, this must be consistently completed for all existing access profiles and users. For this reason, the read permission cannot be de-selected. If a user or access profile should not be granted any permissions on the contents of this folder, you persistently allocate permissions for all authorized access profiles.

11. Distribution Lists

Distribution lists are checked processes in which DOCUMENTS files are routed, specifically and with distributed tasks, through the company. The individual steps of a distribution list are iterated through in order, where prompts or forwarders can be used to allow temporary deviation from the specified order. Addressees of a *routing step* can be *users*, *access profiles*, *aliases or e-mail recipients*.

The definition of the process with its individual steps and tasks is performed either in the DOCUMENTS Manager or in DOCUMENTS 4. This variation supports the users through a visualized process wizard, enabling both flexible ad hoc routing and saving processes that have once been configured. These can then be reused in identical format for other DOCUMENTS files.

11.1 Distribution lists in the Web application

The *Send* command from the *Actions* list in file view is used to start internal routing of a file supported through a wizard. Fig. 66 shows a sample distribution list in DOCUMENTS 4.

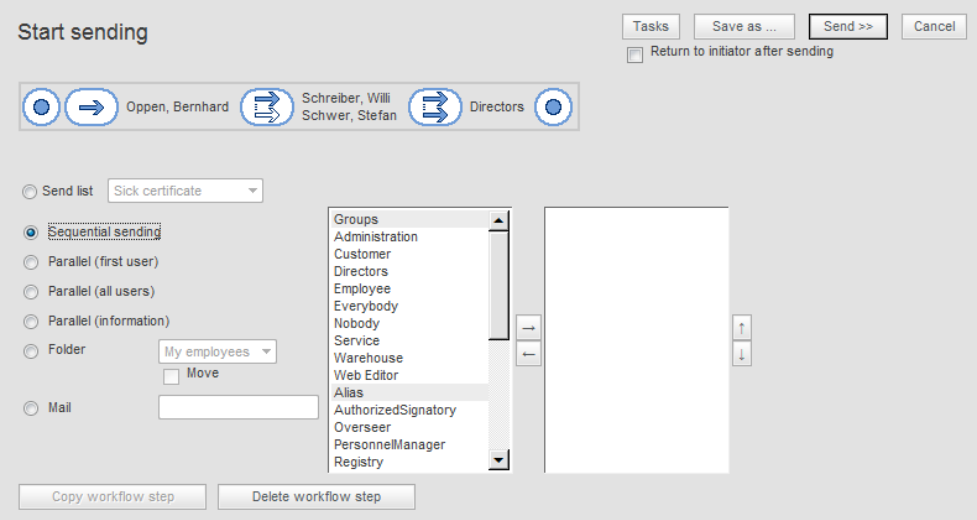

Fig. 66: Routing via DOCUMENTS 4

This method allows you to start a simple process (e.g. approving a vacation request) while the DOCUMENTS file is passing through different stages.

Routing consists of different steps that can be defined by the user on a case-by-case basis. Moreover, each *recipient* can be assigned a *task* that they have to complete before continuing with the next routing step.

You send a DOCUMENTS file to other users by opening it in view mode and, in the *Actions* list, select the *Send...* entry.

This opens the wizard for sending a DOCUMENTS file in the workspace (Fig. 66) which will support you in creating the individual routing steps. The wizard's key functions include a host of workflow options and a double selection list. Here you select the recipients of a routing step from the list of all registered users, aliases and user groups.

11.1.1 Define recipients

You define the recipients for a routing step by selecting the desired entries in the selection list in the left pane and by importing them into the list in the right pane by clicking the arrow pointing to the right. There, you can in turn remove wrong entries by reversing the procedure and clicking the arrow pointing to the left.

DOCUMENTS 4 provides flexible *workflow options* which you define in the left pane of the wizard by enabling the corresponding checkbox.

The following *workflow options* are available:

- *Sequential sending*: All selected users consecutively receive the DOCUMENTS file in their Inbox. The order of recipients is determined by the selection list in the right pane. The up and down arrows let you switch selected entries arbitrarily.

- *Parallel (first user)*: All recipients receive the DOCUMENTS file at the same time. However, once one of them edits it, it is withdrawn from the Inboxes of the other users. This type of routing is always recommended when a specific task can be edited by a group of people and it does not matter which of these persons takes on the task.

- *Parallel (all users)*: The DOCUMENTS file is concurrently sent to all selected users; however, it must be edited by all of these users. Unlike the *First user* variant, edits made by one of the recipients is not enough. The routing step that follows can only be started when the predecessor has been edited by all recipients.

- *Parallel (information)*: The DOCUMENTS file is concurrently sent to all users, where these do not have to execute any tasks, but are only informed about it.

- *Mail*: The DOCUMENTS file is sent to an e-mail recipient whose address is entered in the adjoining field. This provides the option to also include external recipients with no access to DOCUMENTS 4.

11.1.2 Import routing step

Once *recipients* and *workflow option* are defined, the routing step must be imported by clicking the corresponding button. In the top section of the wizard the defined step is then graphically displayed, and another one can be appended. Fig. 66 shows an example with various steps and workflow options which you can differentiate with the respective icons.

Optionally, you can determine via checkbox whether the DOCUMENTS file should be returned *to the initiator* after sending it (Fig. 66 top right). This allows you to subsequently check proper iteration of all steps and tasks.

11.1.3 Distribute tasks

Each recipient can be notified about a *task* that they must edit by a specific time before routing is continued (Fig. 67). The description of the task is displayed along with the DOCUMENTS file in the recipient's Inbox.

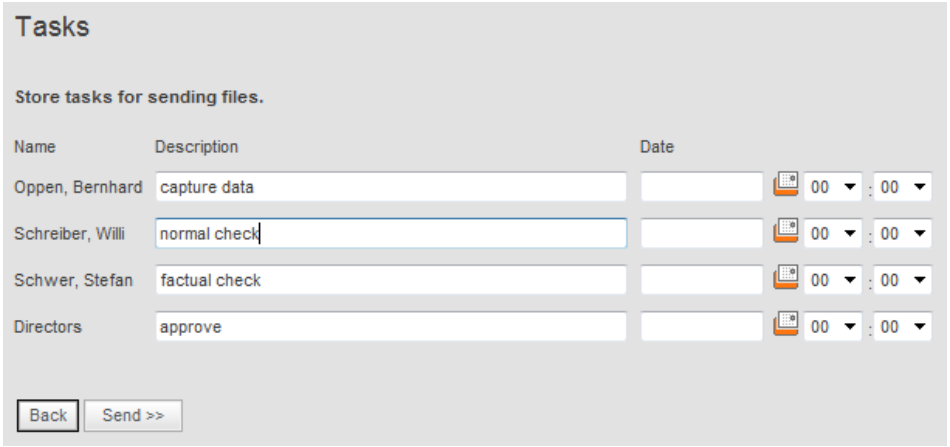

Fig. 67: Routing tasks

Following this, click the *Send* button to route the file to the initially defined process step.

11.1.4 Monitor routing

You will obtain an overview of all your sent contracts anytime in the *Sent Items* folder. If you have finished sending and you have enabled the *Return file to initiator after sending* option, these contracts will be stored in the *Finished sending* folder.

11.1.5 Save distribution lists

If you require regular identical routing, you can save this as a template once you have entirely defined it. To do this, click the *Save as...* button and, in the dialog that follows, specify a descriptive name.

Thereupon another *send option* appears in the form of a list that contains all saved routing items. All you select here is the completed routing, and then you click *Apply routing step*. The entire list is built immediately and you therefore do not need to re-create all individual steps.

> *Since distribution lists do not contain any file-specific information but only represent a recipient list including tasks, these can be used globally for DOCUMENTS files of all file types.*

11.1.6 Forward received DOCUMENTS files

If the next step is reached as part of routing, the DOCUMENTS file is placed in the Inboxes of the users defined in this step. When you open this DOCUMENTS file as a recipient, the assigned task will be visually highlighted above the tabs (Fig. 68).

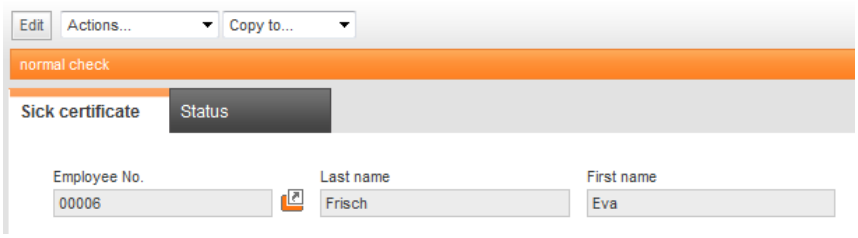

Fig. 68: Task as part of routing

Moreover, the *Status* tab displays the routing progress.

To forward the DOCUMENTS file to the next routing step, you need to select the *Forward* action after editing the task you were assigned.

The following dialog *Forward file* then contains various steps for selection (Fig. 69). In the default case, it is assumed that the file is *OK*. In this case, you can optionally enter a comment and forward the DOCUMENTS file to the specified recipient.

Whereas when you enable the *File not OK* option, new selection options will be available. The DOCUMENTS file can now be sent again to any system user to either forward it *unscheduled* or send a *prompt*. In both cases you can also address a new task to the selected recipient.

In both cases (*Confer with* and *Forward*) the defined distribution is interrupted and follows a different path ad hoc. The difference is that another step is added to the prompt but the DOCUMENTS file is then returned to the original step.

Whereas in case of a *forwarder* the current step is not restarted, but the next scheduled step is directly navigated to from the forwarded step.

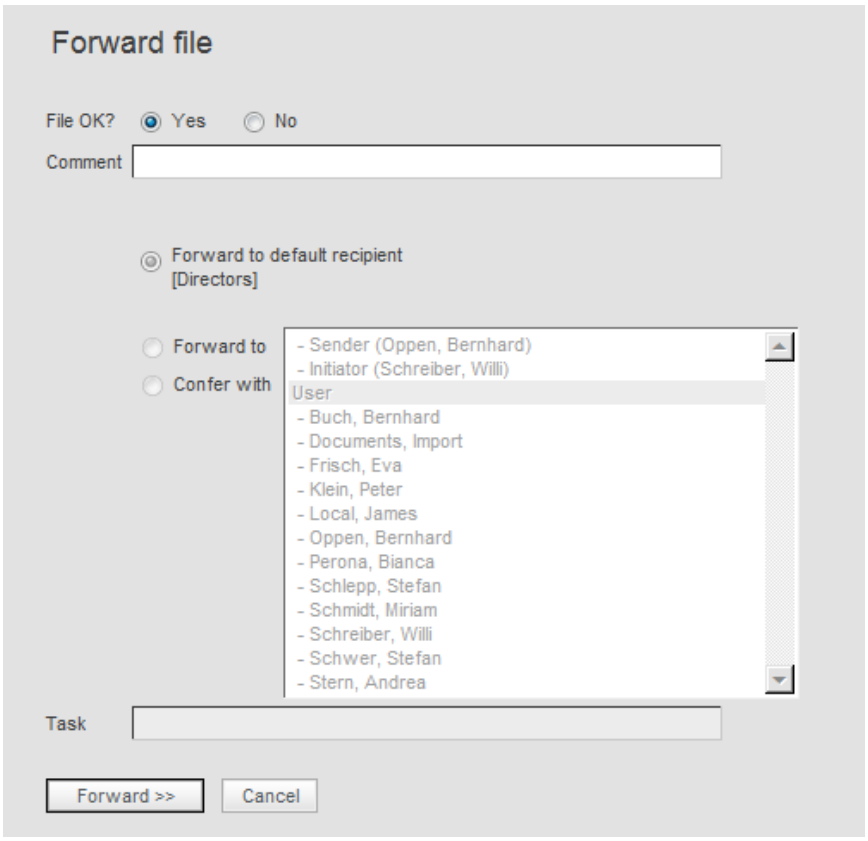

Fig. 69: Forwarding a contract

The sent DOCUMENTS file allows you to optionally display the process flow on a tab named *Monitor* (Fig. 70). This displays the entire history of the flow and you can see anytime which steps have already been walked through and which are still to follow.

Other settings of the file type are required for enabling this representation. These are described below in chapter 11.2.3.

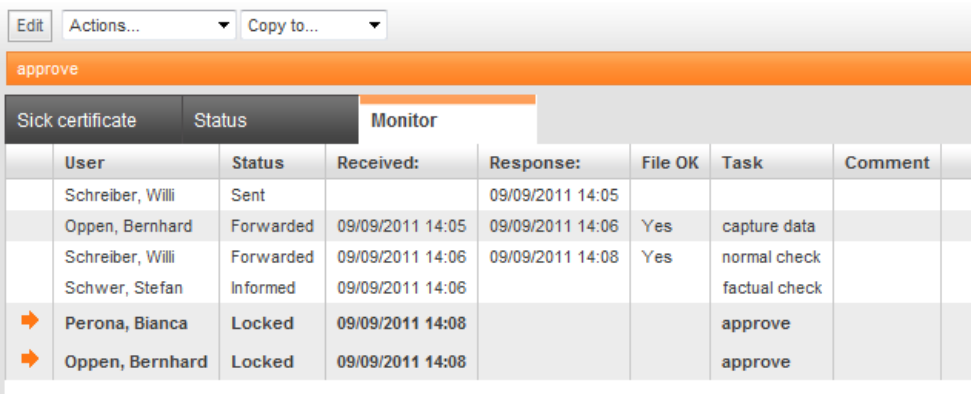

Fig. 70: Representation of routing in the monitor

11.2 Distribution lists in the DOCUMENTS Manager

In the DOCUMENTS Manager, distribution lists are configured and managed in the section of the same name below the *Documents* entry in the tree structure.

In this section all *distribution lists* are managed, i.e. both those created in the DOCUMENTS Manager and those saved by users in DOCUMENTS 4.

In the DOCUMENTS Manager's configuration dialog (Fig. 71), a distribution list is essentially composed of a technical *name*, *membership,* an optional *owner,* and a host of *routing steps*.

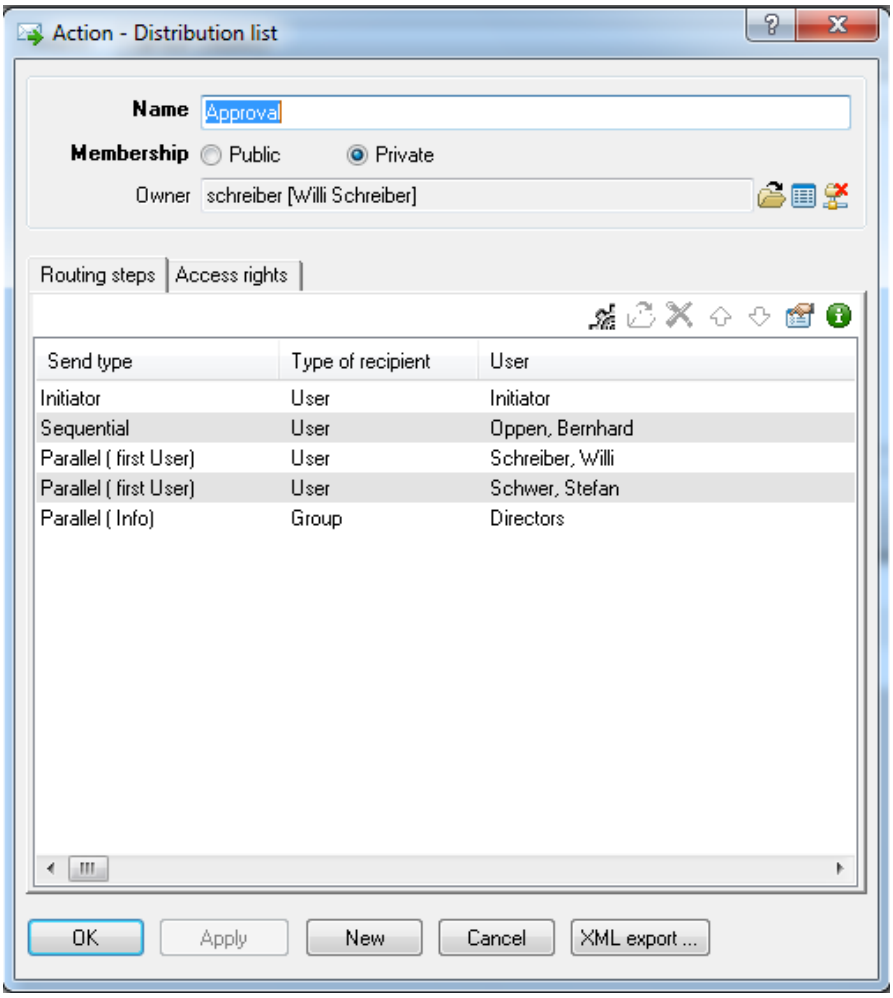

Fig. 71: Distribution list in the DOCUMENTS Manager

Membership controls who may use this distribution list. Each user is permitted to send this DOCUMENTS file based on a *public* distribution list, whereas *private* distribution lists can only be used by their *owners*. An owner can therefore not be specified with public distribution lists.

Distribution lists created by users in DOCUMENTS 4 are initially *private* ones. However, this property can be changed in the DOCUMENTS Manager. Thus, you can finally also use the Process Designer to create global distribution lists. All you need to do is correct *membership* for the workflow created in the Process Designer.

11.2.1 Routing steps

Each distribution list starts with an *initiator* step that is automatically created using the distribution list. With the exception of the *Return file to initiator after sending* checkbox, this step cannot be modified. This setting globally effects the entire distribution list, and is therefore read-only in the following steps.

More steps can be added via the New creation button on the *Routing steps* tab. Fig. 72 shows the configuration dialog of a routing step that follows.

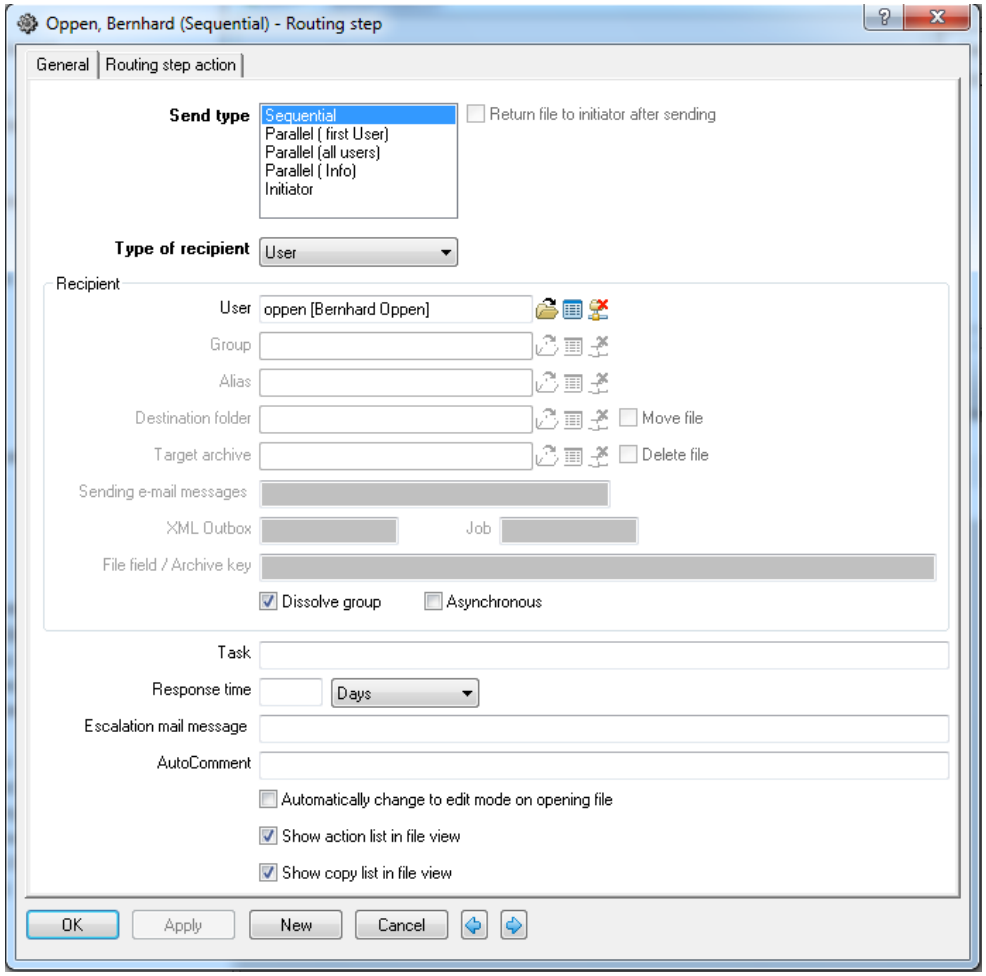

Fig. 72: Routing step of a distribution list

General

On the *General* tab, you need to initially select a *send type* that has already been described in chapter 11.1.1.

The *Type of recipient* controls the *Recipient* section below it. Depending on the selected type, the corresponding selection field (*User*, *Group*, *Alias*) becomes active in the Recipient section. If the DOCUMENTS file is to be exported to a *folder* or *archive*, exported via *XML* or sent via *e-mail*, this action will be executed directly when it occurs and directly continued with the next step.

When enabling the *Dissolve group* checkbox on sending to a group, a separate process step will be generated for each user of that group. Whereas when this option is disabled, only a single step will be used for the entire group.

The recipient types *User field*, *Alias field* and *Group field* are special features. When clicking one of these options, the *file field* will be activated.

You can then enter a user, alias or group here. In Web view the field of a DOCUMENTS file will then include a recipient. That recipient can also be determined at runtime.

The entered *task* is presented to the recipient when viewing the DOCUMENTS file on the Web. Additionally, the file is placed in the user's *Task* folder.

The *response time* indicates the interval available to the editor for this step. Once the time is exceeded, an *escalation mail message* is automatically sent to the entered recipient.

The *Automatically change to edit mode on opening* option will then be an advantage if the recipient needs to frequently modify field values to fulfill their task. Otherwise, the user can also manually switch to edit mode by pressing a button, when required.

Hiding the *Show action list in file view* and / or *Show copy list in file view* options is useful to enable the recipient a better overview within their step on the Web. These drop-down lists should only be displayed when required (e.g. for prompting options).

Routing step actions

A routing step can be allocated specific *routing step actions* in the DOCUMENTS Manager. These are reflected as a function on the DOCUMENTS file on the Web executed by the current user.

Actions allow further refining of what exactly the users can do in this step.

A new *routing step action* is created via the left button on the tab of the same name. Following this, the dialog shown in Fig. 73 opens.

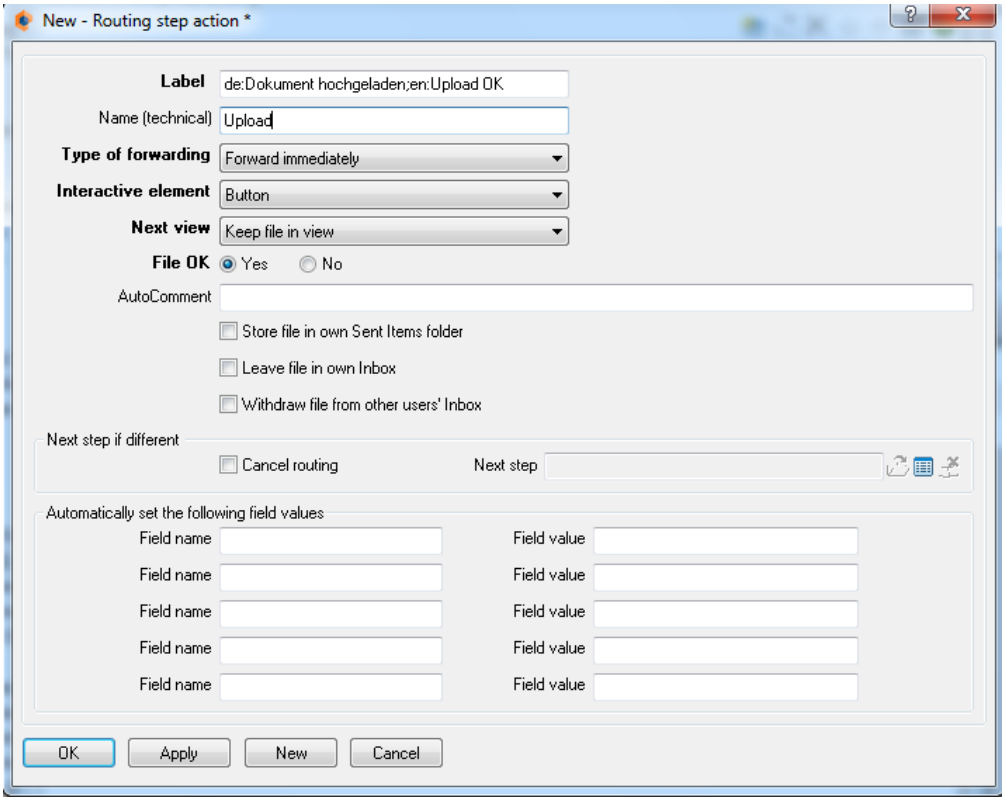

Fig. 73: Routing step action

In this dialog, specify an ergonomic *label* first. This can optionally be multilingual, and is used for the *interactive element* in DOCUMENTS 4, which controls whether the action is to be performed directly as a *function button* or whether it is given an entry in the *Actions* list.

Fig. 74 in this context shows the routing step shown in Fig. 73. Here a *function button* was selected as the *interactive element*. This can be seen in Fig. 74 with the "*Upload*" label. The user must only press that function button to route the DOCUMENTS file to the next process step.

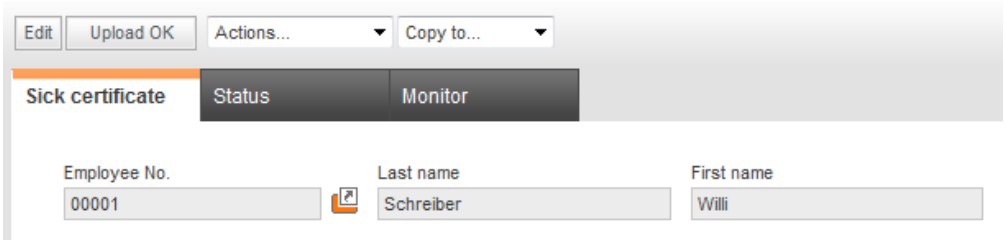

Fig. 74: File action using the interactive element "Function button"

The *Type of forwarding* can be selected as *Forward immediately*, *Forward through dialog* or *Do not forward*. Actions without forwarders remain on the DOCUMENTS file after executing them, while immediate forwarders jump to the next routing step.

The *Next view* defines DOCUMENTS 4 behavior from the current user's viewpoint after starting the action. The following options are available: *Keep file in view, Overview, Inbox* and *Next file*.

The *File OK* function and using an *auto comment* are immediately set within the process. Moreover, auto comments can use auto texts.

Below file view a new entry that contains these notes is generated in the monitor. Fig. 75 illustrates this in the second row.

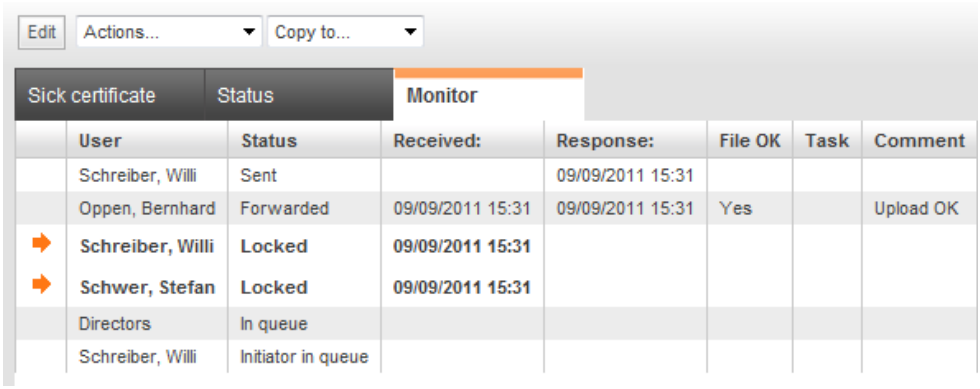

Fig. 75: Monitor including auto comment and "File OK" entry

If this action is to be used to deviate from the original sequence, you can select the *Cancel routing* option, for example. In this case, the distribution list immediately jumps to the endpoint without considering the steps in between.

The option to branch within the workflow is also available. In this case, you can an alternative *Next step*. Routing is then continued at this step. This facilitates implementing loops or jumps within this list.

Each action allows *automatically setting field values of the DOCUMENTS* file. The assignment can be performed both statically via fixed values and dynamically via auto texts. A frequently returning case is setting timestamps of an action or populating a recipient field through the current user.

11.2.2 Access rights

Using *access rights* (Fig. 76), you optionally define which users or access profiles may use this distribution list. Two rights, which can be enabled via checkbox, are distinguished here:

- *Create distribution list*: Without this right, users may not execute routing based on this distribution list.
- *Change distribution list*: Users who have this right may use the distribution list as a template and, moreover, manipulate or augment the existing steps for individual routing.

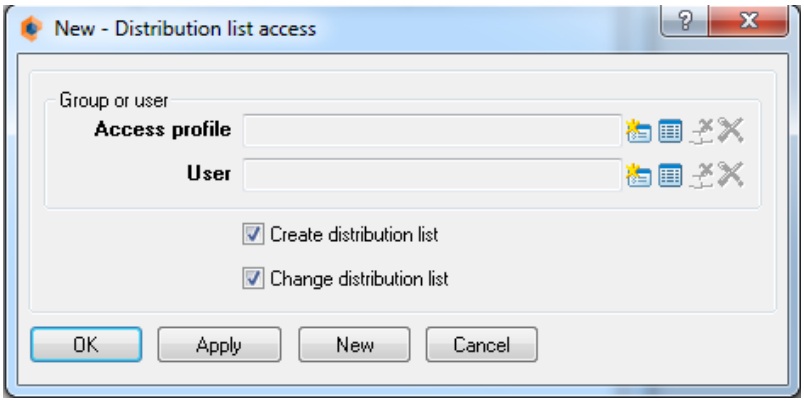

Fig. 76: Access rights on distribution lists

DOCUMENTS 4 rigorously uses a positive permission. Without allocating permissions, all users are allowed to execute the actions. Whereas once rights have been allocated, only the privileged users may exercise these rights. So, if different users or access profiles should have these rights, rights allocation must be repeated for all relevant users and access profiles.

11.2.3 Workflow settings on the file type

Specific settings for distribution lists and workflows must be configured for the respective file type. Fig. 77 shows these options. Open the configuration dialog of the desired file type, and click the Workflow tab to customize these options.

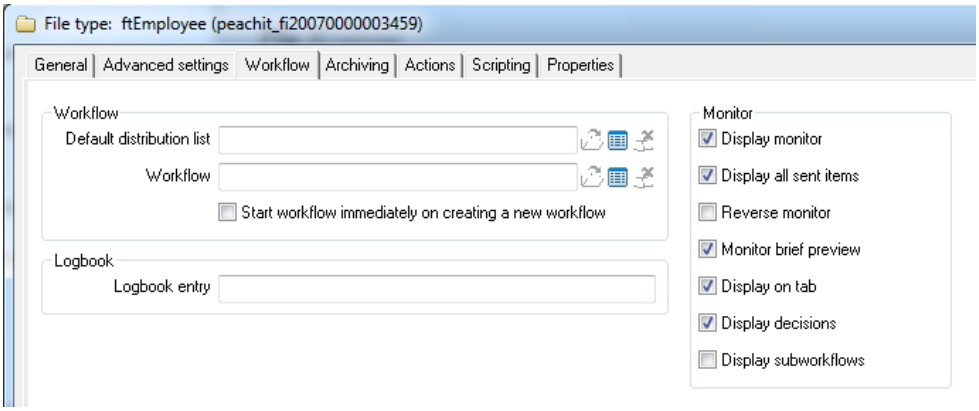

Fig. 77: Workflow settings on the file type

Initially, you can enter a *Default distribution list* or *Workflow* for the file type, and optionally *start it immediately on creating a new workflow*.

In the *Monitor* section you control options that have already been used in the above examples. The *Display monitor* checkbox is used to add visualization of the process steps to the DOCUMENTS files. Moreover, this can be displayed on a separate tab. Fig. 75 illustrates this behavior.

12. Workflows

Workflows represent functional extension of distribution lists. For instance, these allow branches, context-dependent decisions, escalations, or creating subworkflows.

Defining workflows requires extending the license; this is performed directly in Microsoft Visio using an interface (Fig. 78). The workflows defined there are synchronized in the DOCUMENTS Manager and are visible in the *Workflows* tree section. However, changes to their configuration cannot be made, because otherwise modeling in the DOCUMENTS Manager might differ from that from Microsoft Visio.

However, existing processes in the *Workflows* section can be selected as default workflows for a file type (see chapter 11.2.3).

Configuration and use of workflows is described in separate documentation.

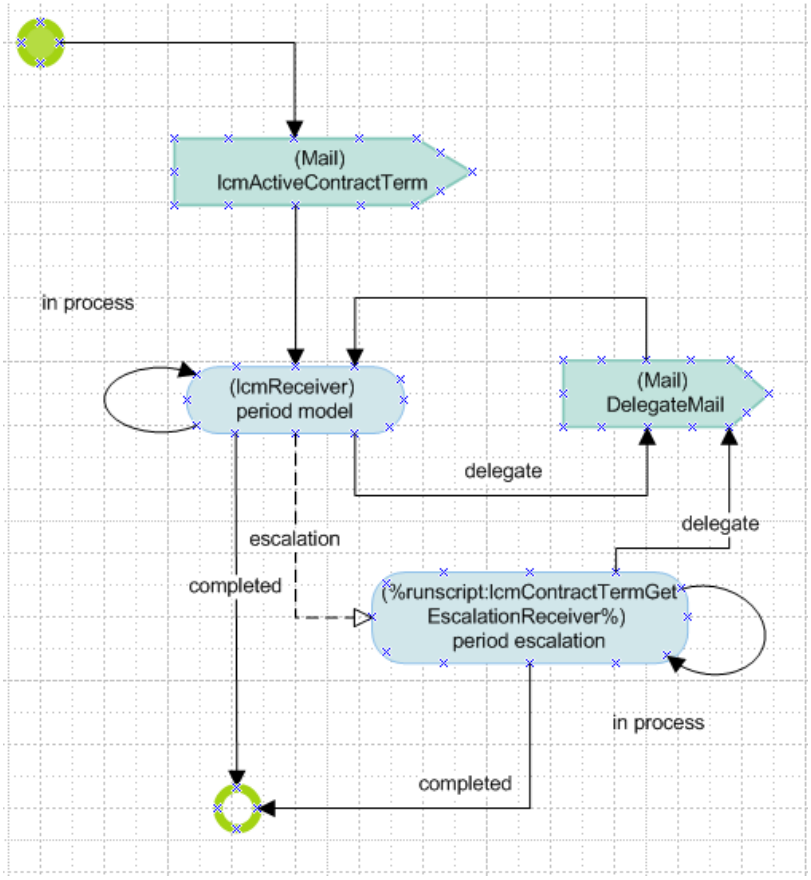

Fig. 78: Sample workflow in Microsoft Visio

13. Scripting

The DOCUMENTS 4 server includes an integrated *scripting engine*. Thus, it enables *server-side* execution of *JavaScript.*

These scripts are defined in the DOCUMENTS Manager. This is performed in the form of global script objects, which are initially created in a scripting library of the logged-in principal (Fig. 79).

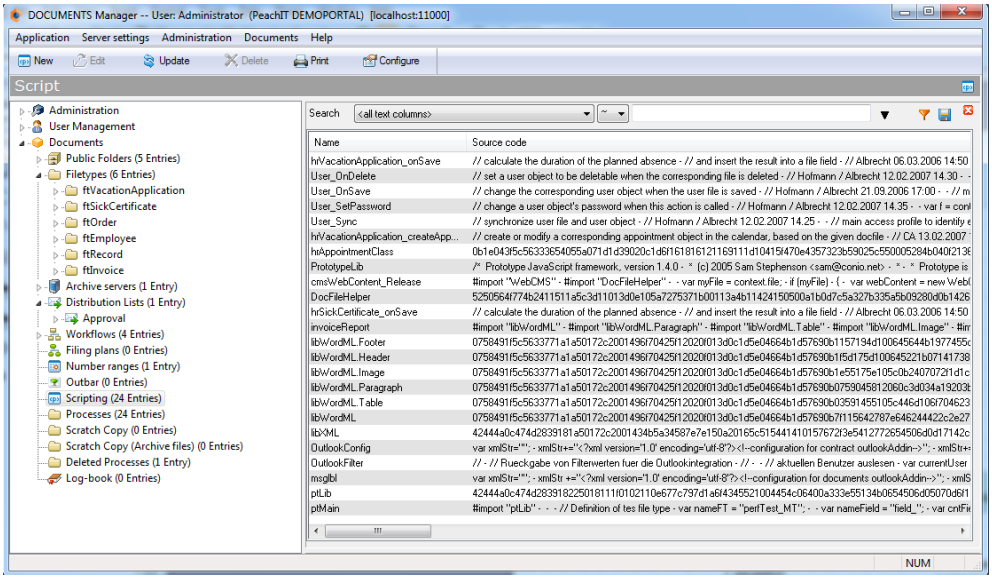

Fig. 79: Scripting library in the DOCUMENTS Manager

These are then embedded into predefined positions for use. Execution is performed according to the embedded position during runtime with specific actions. The following uses of server-side scripts are available here:

- Default actions of DOCUMENTS files. The script is integrated with a defined action of the file type:
  - On creating new files
  - On editing
  - On saving
  - After saving
  - On archiving
  - On deleting
  - As "Allowed actions" script: This controls globally for the DOCUMENTS file which actions are generally available to the user.

Fig. 80 shows for a file type in the DOCUMENTS Manager how assigning existing scripts to specific actions is implemented.

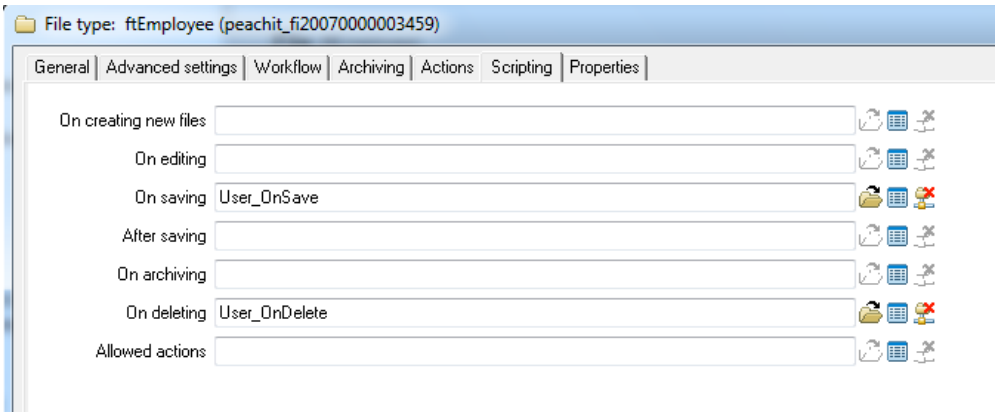

Fig. 80: Script actions of a file type

- As custom actions of the DOCUMENTS file, and of folders
- For describing conditions (guards) in workflows
- As send signal in workflows
- As part of field use actions in workflows
- For defining enumeration values of a field of the enumeration type

This allows implementing specific requirements in DOCUMENTS 4 such as:

- Drop-down lists that are dependent on the values of other fields
- Access to third-party databases for filling drop-down lists or field values
- Complex guard conditions that cannot be defined through simple expressions
- Automation of processes in the form of job-driven scripts
- Starting external DLLs to navigate towards third-party systems
- For calculating data

Fig. 81 shows the structure of a script based on a simple example which changes the password of a user via a dialog.

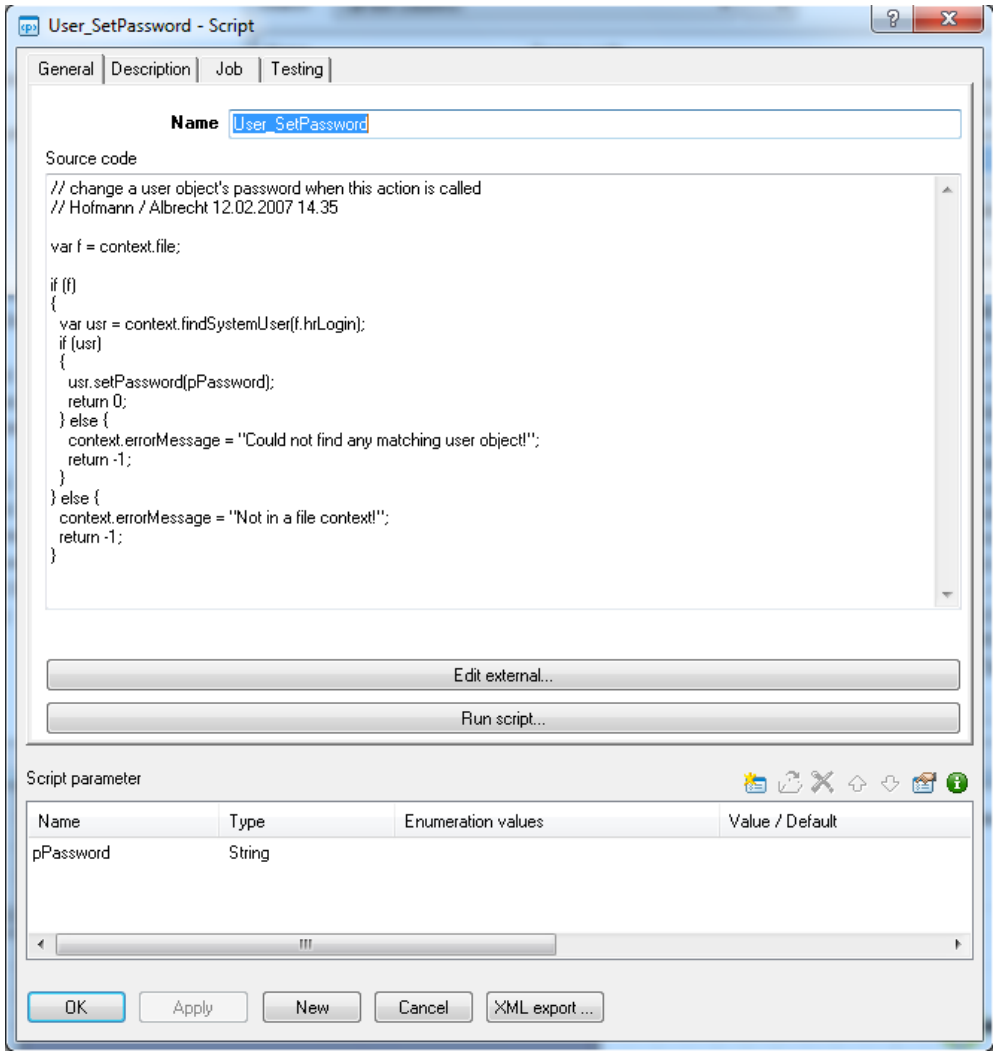

Fig. 81: Script in the DOCUMENTS Manager

The DOCUMENTS 4 scripting runtime environment facilitates accessing various file and field properties. Runtime constants can also be read, such as the currently involved user or the name of the current workflow step.

Due to the complexity of this topic, this is treated in separate documentation.

14. File Plans

File plans are an extension to DOCUMENTS 4 that require a license. If the corresponding license is not present, the File plans entry in the DOCUMENTS Manager will be invisible.

File plans replicate a tree structure in DOCUMENTS 4, thus enabling hierarchical storage of DOCUMENTS files. This is not dependent on a defined structure of public folders. Thus, file plans facilitate alternate grouping of DOCUMENTS files according to a unique storage structure.

The use of file plans includes both DOCUMENTS files and integrated archives.

Only the hierarchical structure is created in the file plan itself. DOCUMENTS files are grouped by using a field of the *File plan* type. Which file plan is to be used for the relevant file type is defined in this type. In DOCUMENTS 4 this field provides a pop-up dialog that lets the users define in which node within the file plan structure the respective DOCUMENTS file should be stored. Fig. 82 shows a sample file plan in the DOCUMENTS Manager including different levels. In this context, you can create any number of levels as well as any number of entries at a single level.

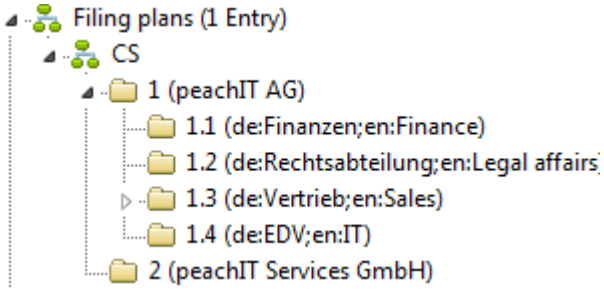

Fig. 82: File plan in the DOCUMENTS Manager

Because a file plan is not bound to a single file type, processes of any file types can be integrated with a file plan structure.

14.1 Define file plans

Select the *File plan* entry in the DOCUMENTS Manager tree structure, and then click the *New* button in the toolbar section. Following this, the (still unpopulated) dialog window shown in Fig. 83 opens.

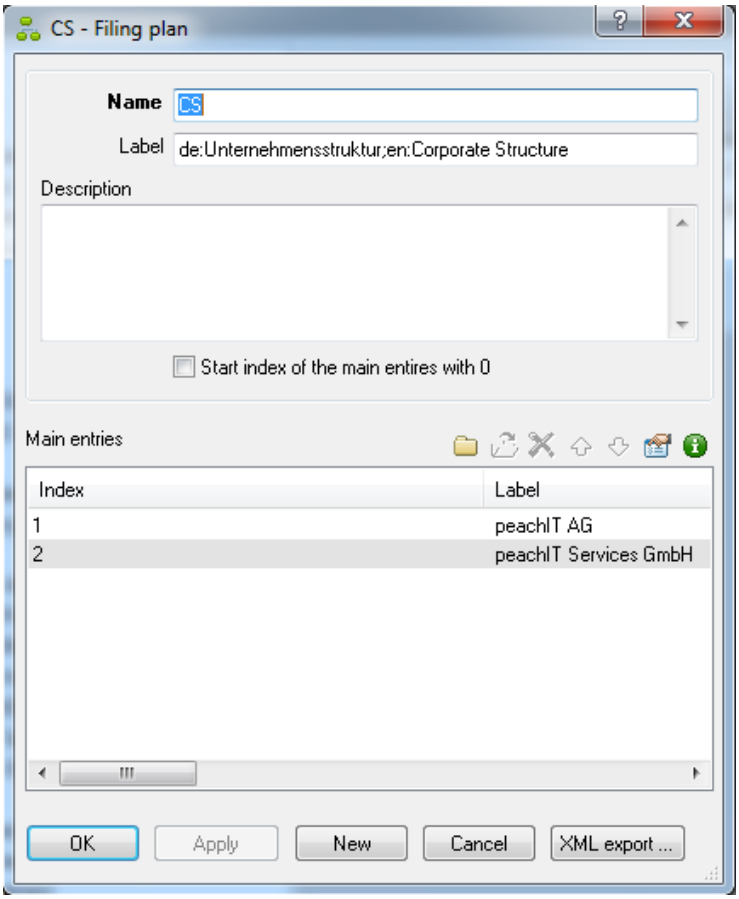

Fig. 83: Creating a file plan

Specify a technical *name* and an optionally multilingual *label* here first.

The result of enabling the *Start index of the main entries with 0* checkbox is that these will not start with 1 as the first file index, but zero-based.

In the bottom section of the dialog window, overwritten with *Main entries*, you create the top hierarchy level of your file plan. Please be aware that if you change the order of entries the index order will still be retained. In other words, the item in the first position is always assigned the smallest index. When, for instance, you swap entries 1 and 2, the original second entry will now be assigned the ordinal number 1.

You should therefore have defined the ultimate structure of your file tree already prior to storing the first DOCUMENTS file, because otherwise processes that already exist would be out-of-order.

You create a new *main entry* via the button on the left in the section of the same name. Fig. 84 shows the following dialog for creating a new main entry.

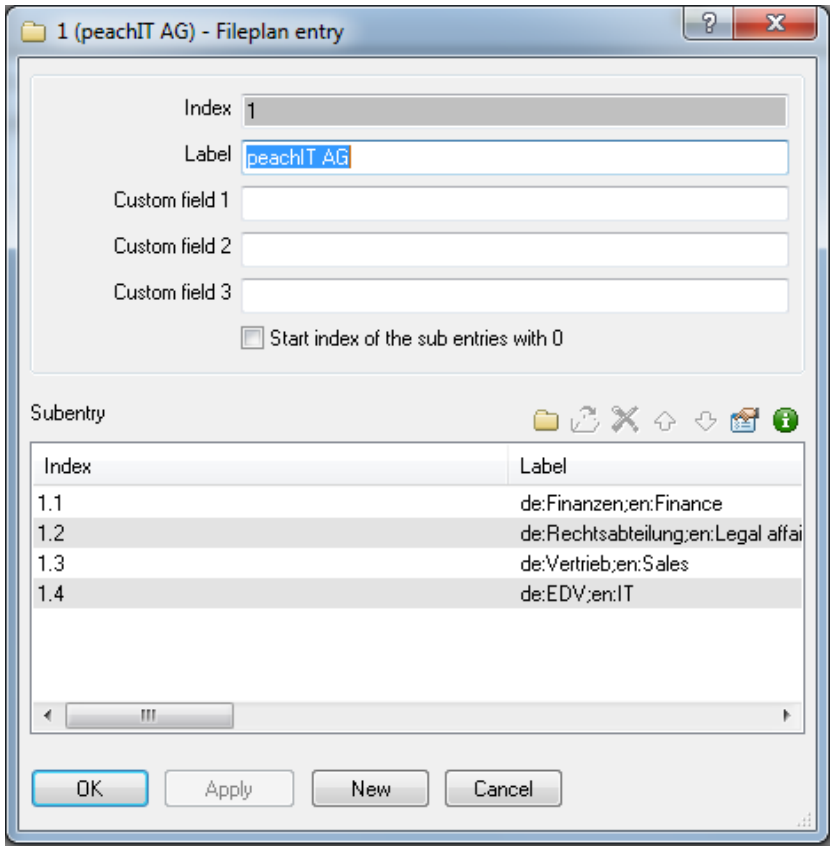

Fig. 84: Main entry of a file plan

The *index* of the entry is automatically allocated; it is read-only. This ensures that each entry is assigned a unique index that continues at the respective level.

Specify an optional multilingual *label* first. Moreover, the option to specify up to three *custom fields* is available. You can access the contents of these fields via auto text when linking file types or archives. For more information on their use, see chapter 14.4.

The bottom dialog section allows opening another level based on the relevant entry by defining *subentries*.

Creating *subentries* is analogous to the main entry: So, when clicking the corresponding button in the bottom section, the familiar dialog for entering a new entry opens (see Fig. 85).

Analogous to the previous dialog, you can also use subentries for a zero-based count.

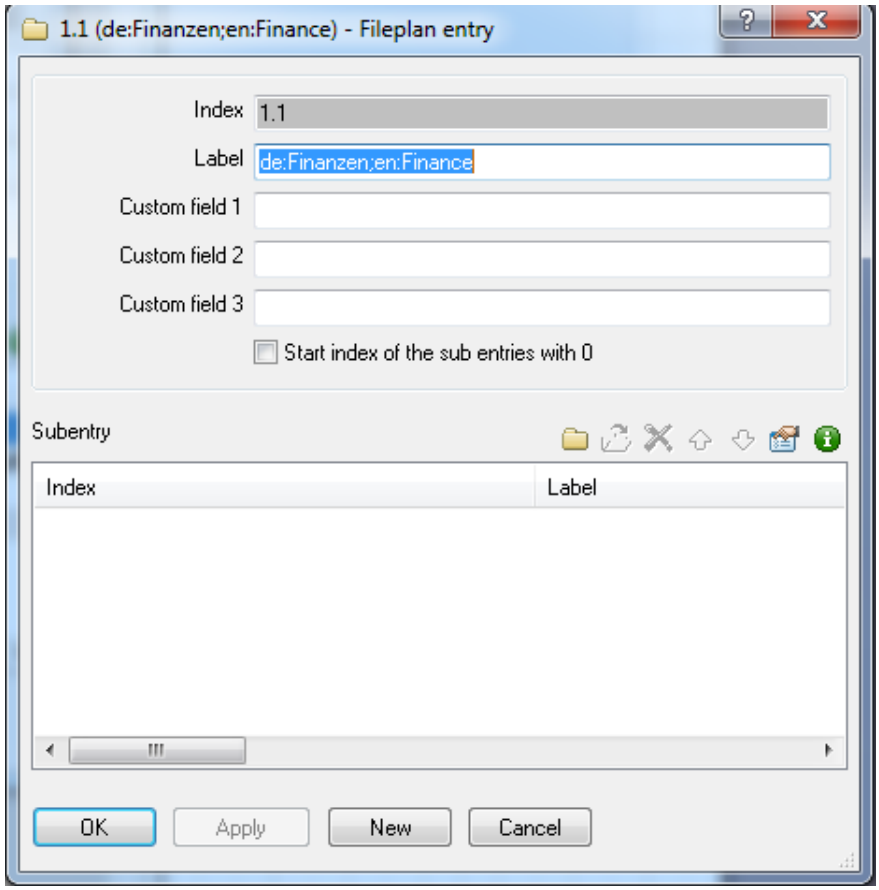

Fig. 85: Subentry of a file plan

The only difference to the main category is that the index used is now composed of two levels. The first number here refers to the related main entry, while the second number localizes exactly the relevant subentry at its level.

File plans themselves do not include permissions. Instead, permissions on file types, fields and tabs of the linked processes which might exist are reverted to. This also considers file class protection and its special type of *ACLs* (see separate documentation on this).

Another level of permissions will be added when the file plan is displayed on a separate *outbar*. For a description of the use of outbars, see chapter 16 below.

14.2 Integrating file plans

After creating the file plan structure, you will have to integrate the file plan with (at least) one file type or archive.

Without this assignment, the file plan cannot be used in DOCUMENTS 4. To do this, you need to extend at least one file type in the system with a field of the *File plan* type, which is configured for this file plan.

Therefore, open the file type whose processes you intend to enter in the file tree, and create a new field of the *File plan* type here. In the *Enumeration values* field you specify the technical *name* of your file plan; in this way, you establish the connection between file plan and file type. Fig. 85 shows the corresponding configuration dialog of a field. For the demo scenario peachIT, this example uses a file plan including departments to enable assigning DOCUMENTS files of the *Employee* type (*ftEmployee*) to it.

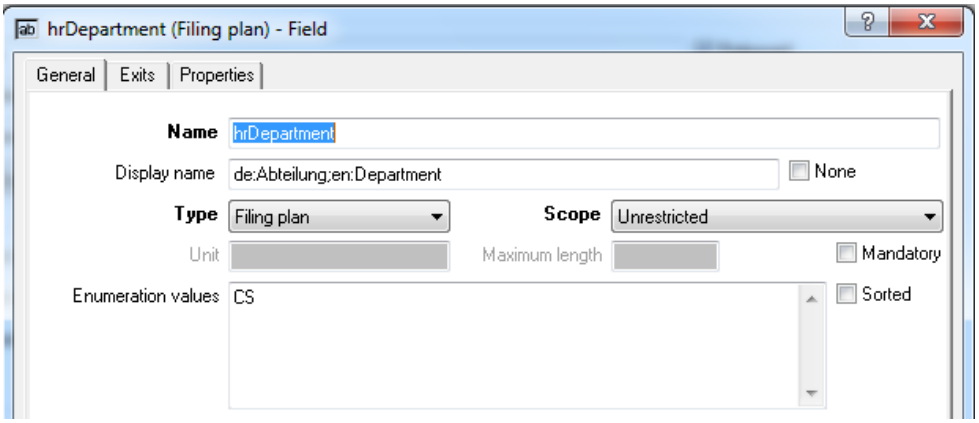

Fig. 86: Field of the "File plan" type

> *A special case is when you are using a DOCUMENTS file plan with an EASY ENTERPRISE.x schema and you want to display this in the outbar at a later time. You will then have to additionally either specify the name or technical key of a view in which the DOCUMENTS files can be searched for in the second text row under "Enumeration values". You may not add another schema in addition to that containing this file plan to this view.*

Finally, it is necessary that you define an *outbar* on which the file plan should be displayed. For detailed information on creating and using outbars, see chapter 16. Fig. 87 shows an example of what such an outbar might look like.

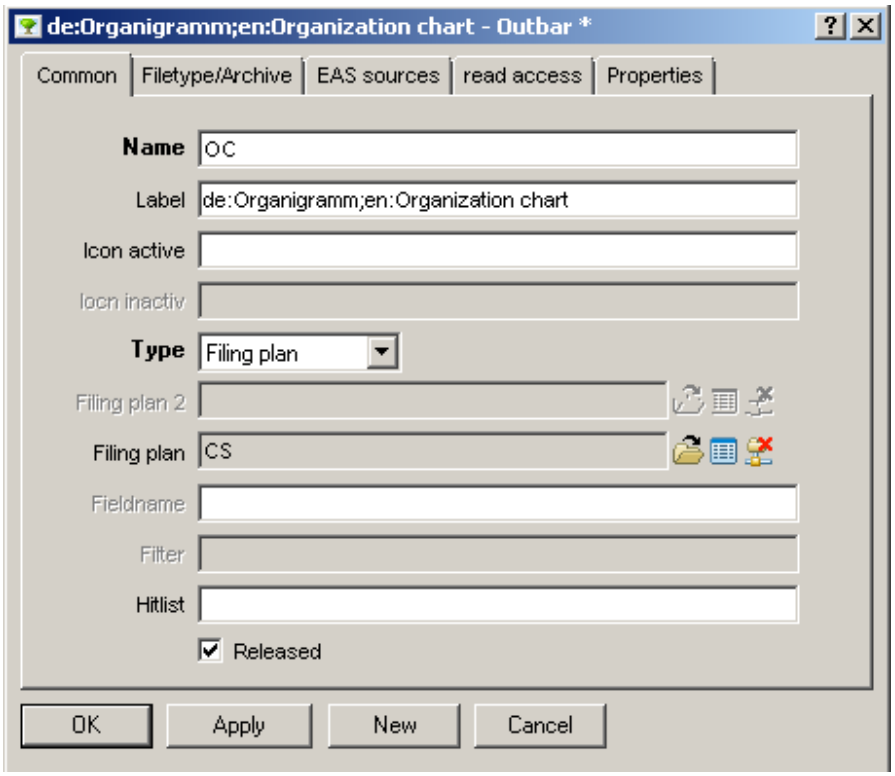

Fig. 87: Outbar for a file plan

To sum up, file plans are set up in three steps:

First, you need to create a *file plan* that sets the structure of DOCUMENTS files or archives to be stored. The second step is to create a *field* of the *File plan* type for the desired file type, and to integrate the file plan there to establish the link between structure and DOCUMENTS files. Finally, you need an *outbar* to integrate the file plan into DOCUMENTS 4 and to make it usable for the users.

14.3 Using file plans

Fig. 88 shows an outbar including file plan in DOCUMENTS 4 for the above sample from the `peachit` principal. Clicking a folder yields a search result with all DOCUMENTS files contained therein. If the relevant folder includes a level containing subfolders, the DOCUMENTS files from the subfolders will also be displayed in the search result.

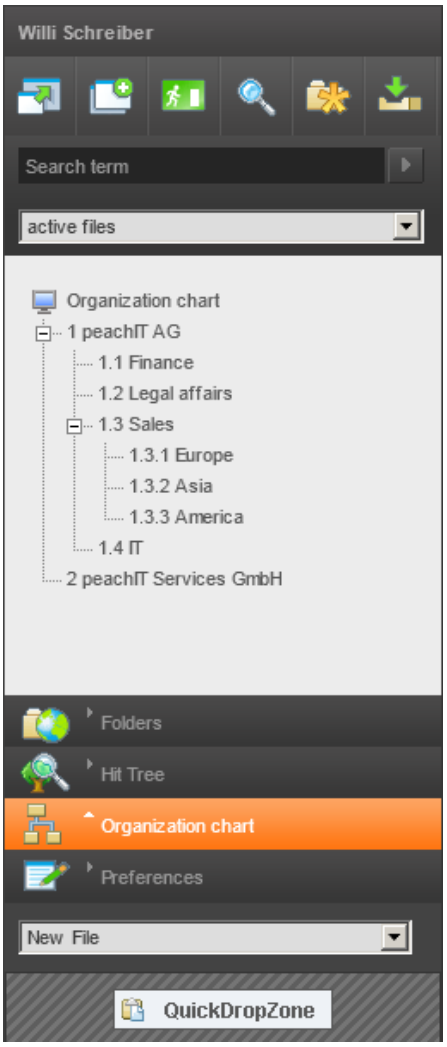

Fig. 88: Outbar including file plan

In DOCUMENTS 4, create a new DOCUMENTS file of the type that you have just linked with the file plan. The new field of the *File plan* type now lets you determine for this DOCUMENTS file in which file plan entry this is to be stored. Fig. 89 shows the field from the above sample.

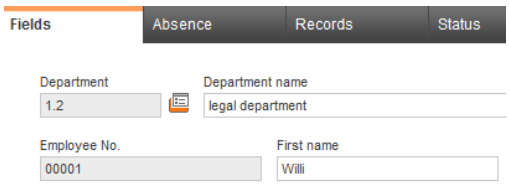

Fig. 89: File plan field

The content area of the field is read-only; you make an entry via the button to its right. When you click the icon, the file plan opens in a dialog and you can select the desired entry via the mouse (see Fig. 90). To open a deeper level, click the preceding plus sign. After selecting an entry this window is closed again and the index is transferred to the file plan field.

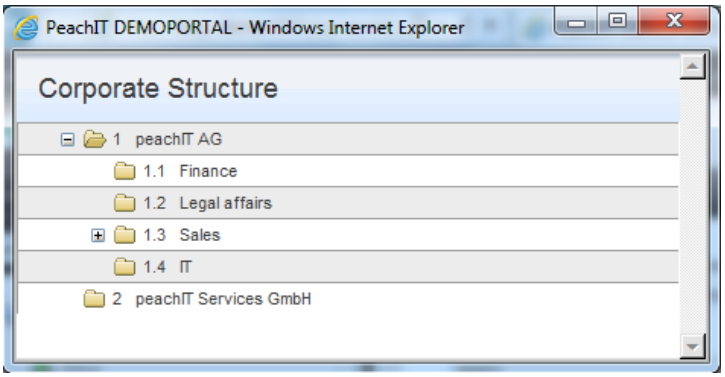

Fig. 90: Selecting the file plan

After saving the DOCUMENTS file, you will find it in the corresponding (parent) category in the file plan's outbar (Fig. 91).

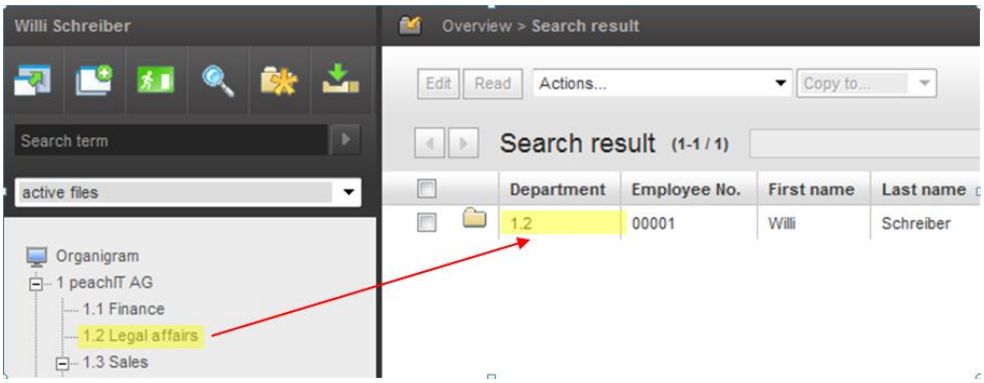

Fig. 91: Search result of a file plan

You can also use *a single* file plan for *multiple* ENTERPRISE.i archives (and compatible) inasmuch as you ensure that the technical name and index number of the file plan *file field* of the respective archive file type are identical in all cases (e.g. name in all archives named "Reference", and index number of this field everywhere is: `1008`).

In reverse, however, it is possible anytime to use multiple *different* file plans for a single file type. However, the use of the same file plan by multiple file fields for the same file type is illegal.

In ENTERPRISE.x a file plan can only be used concurrently by one schema; if necessary, combined with a file type. Different organization of this archive system eliminates the need for linking multiple identical archives (e.g. annual archives) with the file plan. Pools are selected for searches through the archive-side configuration of the view.

14.4 Specific file plan auto texts

Fields of the *File plan* type only incorporate the *index* of the selected entry, but not the *display name* stored in the file plan. This information can be used via specific auto texts in fields of the respective file types.

Fig. 92 and Fig. 93 below illustrate this in an example. For the file type using the file plan, create a new field of the *String* type. This should be write-protected (Fig. 92).

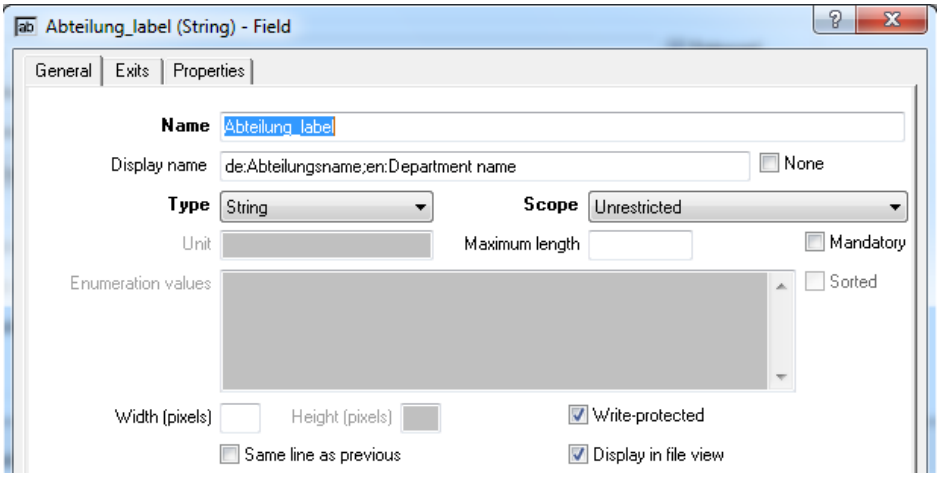

Fig. 92: Field for the display name of a file plan

You add another line to the already created field of the *File plan* type (Fig. 93) under *Enumeration values*. This line must have the following structure:

```
FieldName=%fp.Label%
```

The technical name of the new String field is entered as `field name` (in this example, `Department_Label`). Now this field is automatically populated with the display name of the file plan entry, once a file plan entry is selected for the DOCUMENTS file.

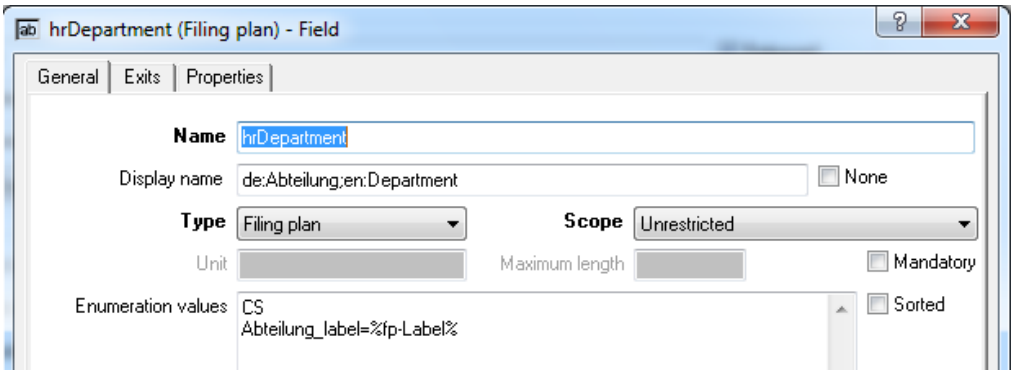

Fig. 93: File plan field including display name reference

The use in DOCUMENTS 4 then looks as follows, for example (Fig. 94):

Fig. 94: File Plan including index and label

In this context, the following specific auto texts for file plans are available:

Auto Text	Description
`%fp.Index%`	File plan index (as in the file plan field itself)
`%fp.Label%`	File plan entry display name
`%fp.Custom1%`	Content of custom field 1
`%fp.Custom2%`	Content of custom field 2
`%fp.Custom3%`	Content of custom field 3

15. Number Ranges

15.1 Use

Using *number ranges,* fields can be automatically populated with a generated sequential number. This, for instance, can ensure that each DOCUMENTS file of a specific type is allocated a unique key that is visible to the users.

Number ranges are composed of at least one counting item; they can be augmented with other static items.

The current number is saved with the number range, and incremented at each call.

The connection between number range and DOCUMENTS file is created by connecting a specific field to the number range.

So, two items are actually required to equip a DOCUMENTS file with such a counter: First, this is the number range itself; second, a field connected to it which is capable of including the current number for the DOCUMENTS file.

To create a new number range, in the DOCUMENT Manager's tree structure, select the corresponding entry and then, in the top left section of the application, click the *New* button.

In the following dialog from Fig. 95 you allocate a *name* for the number range, and define the *number structure*.

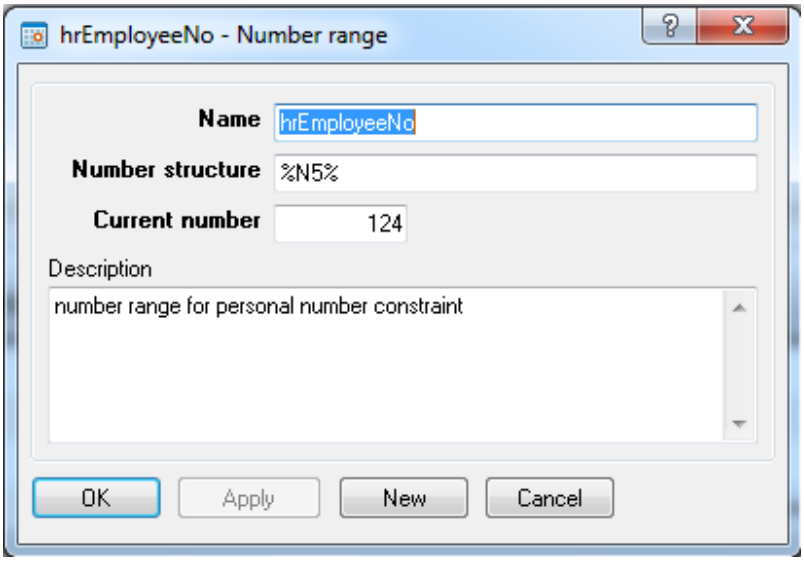

Fig. 95: Creating a number range

The *Current number* field lets you define a start number with which the number range should begin. The *description* is used for transparency; for example, it may contain a description of the number structure or the use of the number range.

To integrate a number range into a *file field*, open the relevant field dialog of the file type.

Here you actually need to integrate the *number range* in the Value / Default setting field. This is performed via the `%nr.NumberRangeName%` variable. In the example shown in Fig. 96, this is `hrEmployeeNo`.

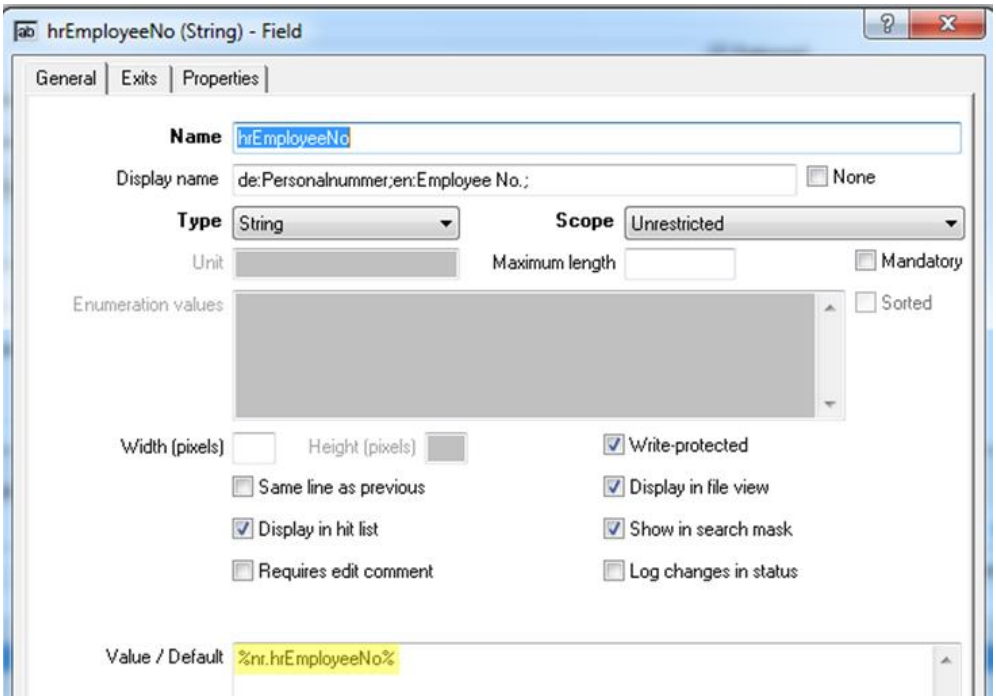

Fig. 96: Field dialog including integrated number range

15.2 Number range structure

The structure of a number range need not be limited to simple numbers. You can include annual, monthly or daily numbers, letters, etc. in the number ranges. Below you will find an overview of the variables that can be used in number ranges.

The definition of the rule for structuring the number must correspond to the preset syntax. Any text with the corresponding key terms can be used.

The following key terms are allowed here:

%N%	Sequential number
%Nx%	Sequential number with n fixed digits
%Y2%	Current year in two-digit format
%Y4%	Current year in four-digit format
%M%	Current month in two-digit decimal format
%D%	Current day in two-digit decimal format
%H%	Current hour in two-digit decimal format
%m%	Current minute in two-digit decimal format
%S%	Current second in two-digit decimal format

> ***Important!***
> *The key term %N% or %Nx% must occur in the type definition of the number range. All other key terms are optional.*

15.3 Examples

`INV-%N4%` returns a field value ***INV-005*** with the current count at 5, i.e. always the `INV-` constant followed by a four-digit number.

`INV-%N/%Y2%` in 2011 shows, at a current count of 197, the result ***INV-197/11***. Unlike the first example, no fixed five-digit number is defined here. Therefore, no leading zeros are preceding this. Thus, unlike above, the counter is well sorted from the outset, because without leading zeros 1 is followed by 10, 11, etc., and only then 2).

A number range in ***Nx*** format always generates a number with n digits, where this format is generated through leading zeros. Whereas in simple ***N*** format, only the current number is used.

When the last allowed value is reached in a number range in ***Nx*** format, an exception is made and the count continues as with number ranges without leading zeros. This also guarantees that each DOCUMENTS file is given a unique number. So, there is no "reset in case of overflow", as was the case with analogous mile counters in old automobiles, for example.

16. Outbars

Outbars are structural elements that in DOCUMENTS 4 set up their own tree structure. An outbar can optionally represent a file plan or a folder structure. Fig. 97 in this context shows an outbar named *Organization chart*, which includes a file plan structure.

Fig. 97: DOCUMENTS 4 outbar structure

You create a new outbar in the DOCUMENTS Manager by selecting the entry of the same name in the tree structure and clicking the *New* button.

The following configuration dialog is shown in Fig. 98. Specify a technical *name* here first, as well as an ergonomic *label* for the outbar. The label can be optionally entered in multiple languages.

Optionally, another graphic can be used by entering the name of a GIF file in the *Icon active* field.

In the *Type* selection list you define the use for the outbar. Either a *file plan* or a *folder* is available here for selection.

For an outbar of the *File plan* type, this must be selected in the field of the same name. Use the buttons to enter an existing file plan.

If this is an outbar of the *File plan* type and if, moreover, DOCUMENTS files of the same file type are stored herein, you can optionally enter a hit list. Thus, DOCUMENTS files can be displayed in the file plan, if necessary, other than displaying DOCUMENTS files in a public folder. Use the technical name of the desired hit list of the file type as the entry.

Enable the *Released* checkbox to make the outbar in DOCUMENTS 4 available to the users.

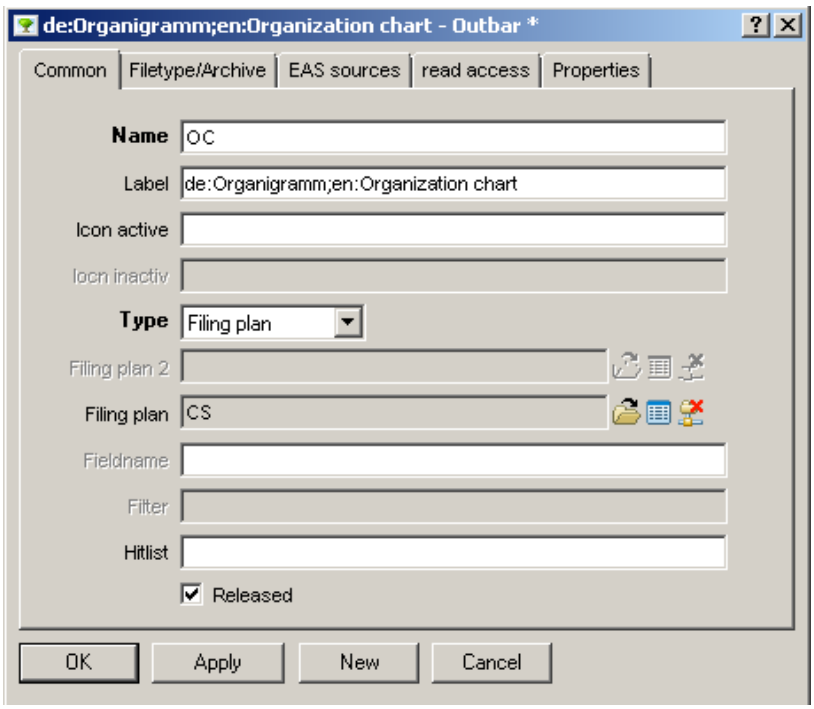

Fig. 98: Dialog for configuring an outbar

Depending on the selected type (*Folder* or *File plan*), the other tabs of the configuration dialog change. The available settings are explained in greater detail in the following subchapters.

16.1 Outbars of the "Folder" type

File plans of the *Folder* type are not bound to *file types* or *archives*. These simply display the contents according to the folder configuration. For this reason, you need to select only those folders to be displayed in the outbar from the *Folder* tab. The purpose of the outbar is only collected display.

The order of the displayed folders can be customized via the arrow buttons. Once assigned to an outbar, folders are no longer displayed in the DOCUMENTS 4 folder panel but only in the outbar itself.

Optionally, *read permissions* at access profile level can be granted for the outbar. Here the principle of a positive permission applies: When granting read permissions for specific access profiles, some of the named access profiles will no longer be automatically granted read permission. Without specifying permissions, the outbar is visible to all access profiles.

16.2 Outbars of the "File plan" type

For outbars of the *File plan* type, on the *File type / Archive* tab, you need to define which of these items should be added to the file plan after selecting the file plan. Above all, be aware while selecting that each selected *file type* (or *archive*) must have a field of the *File plan* type which links the DOCUMENTS files with the selected file plan.

When running DOCUMENTS 4 with an EAS archive, you can include *EAS sources* on another tab which, in addition to the archived DOCUMENTS files, also allows *including active DOCUMENTS files* via a checkbox.

Optionally, *read permissions* at access profile level can be granted for the outbar. Here the principle of a positive permission applies: When granting read permissions for specific access profiles, some of the named access profiles will no longer be automatically granted read permission. Without specifying permissions, the outbar is visible to all access profiles.

17. Processes, Working Copies and Deleted DOCUMENTS Files

The structure of a process is deduced from the underlying *file type* used as a template for creating a new DOCUMENTS file. When a user is saving the file, a process is created where the entered content is created under *Value / Default settings* for each field (Fig. 99).

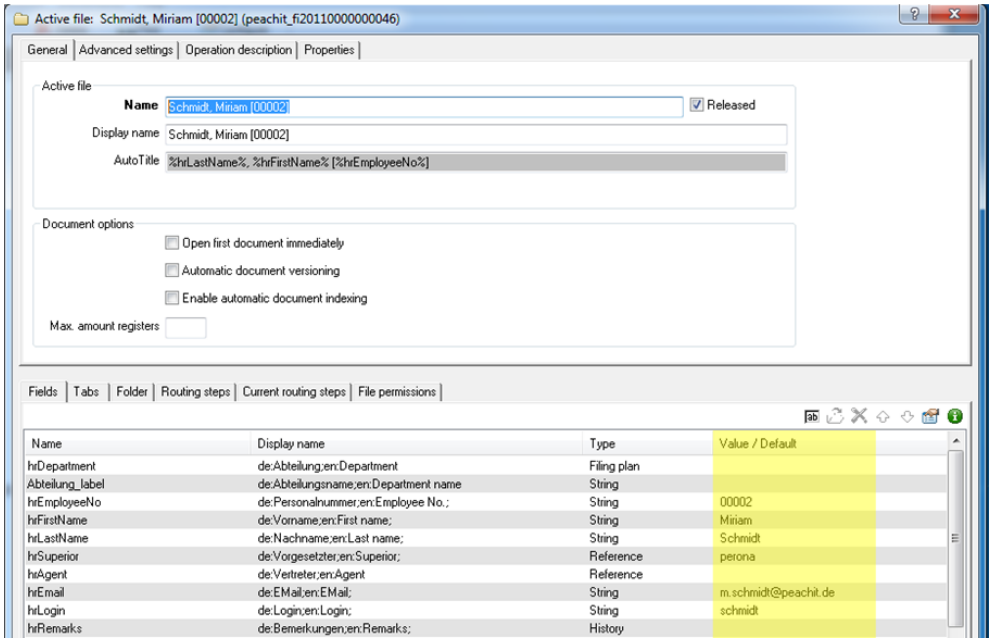

Fig. 99: Process with values

For multi-user operation, these processes are extended with a transaction concept, where *working copies* are used.

When a user opens a DOCUMENTS file via *read access*, that file will be displayed in the browser including the index fields. When, at that moment, other users are opening the same DOCUMENTS file, all users (leaving aside specific permissions here) will receive exactly the same contents.

Now, if a user sets the file to *edit mode*, they will receive exclusive write permissions on that file right at that moment. No other user can now also edit the file at the same time. Internally, DOCUMENTS 4 has now created a *working copy* of the file for the editing user. This working copy is modified by the user, but not the original DOCUMENTS file.

This allows other system users to view the current “As Is” state of the file during search/retrieval, but they cannot edit that file because the system is aware that user A is already editing the file.

Only when the exclusive user saves their changes to the DOCUMENTS file will the changes made to the working copy be actually written back to the genuine DOCUMENTS file. So, only from that moment will these changes be visible again to all other system users.

Whereas if one of the users discards their changes, all the system needs to do is delete the working copy, i.e. rollback to the genuine DOCUMENTS file is therefore not required.

Thus the concept of the working copy is not only a mechanism that guarantees transaction security, it is also an important performance feature. Fig. 100 illustrates this concept.

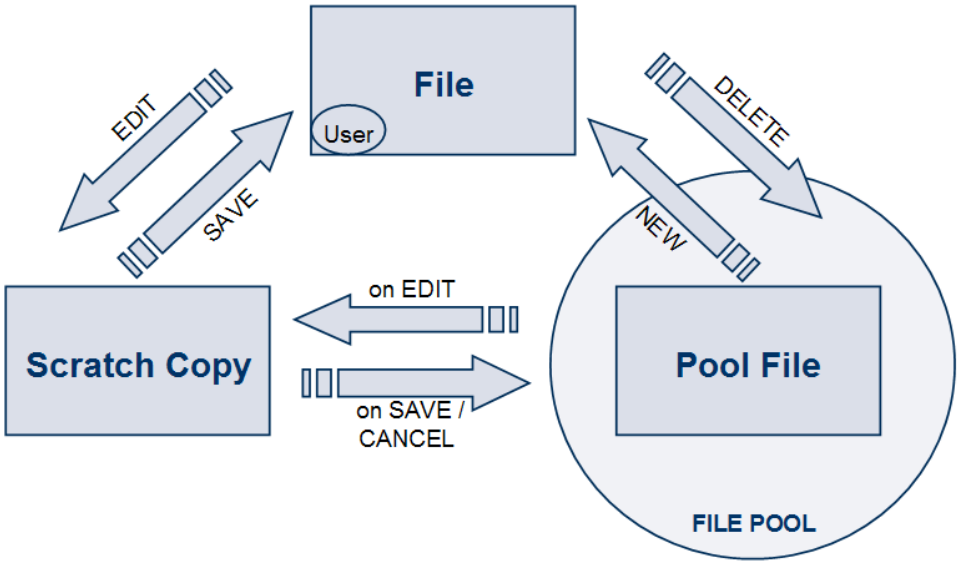

Fig. 100: Working (scratch) copy concept

Within the context of this concept, the functionality of the *Change files* button in the dialog of a *file type* becomes clear (see chapter 9)

On creating a process it is always the structure of the related file type that is used as a template. When making subsequent changes to this structure (e.g. adding more fields), each new DOCUMENTS file is in turn created on this template. Whereas DOCUMENTS files that already exist remain unchanged in their original format; they will not be automatically customized when making structural changes to the file type. Following this, different processes of a file type with a different structure are now temporarily available. Therefore, after making changes to the file type, you must execute the *Change files* function, which customizes all processes to the new structure of the file type by stacks.

This function may take several hours, depending on the number and complexity of existing DOCUMENTS files. Moreover, only processes for which no working copy exists at the time of making the change can be customized. For these reasons, you should make sure prior to executing this function that no working copies exist and that no new ones can be created during this process. To be on the safe side, you should therefore stop the Tomcat service. In this way you ensure that there are no users working with *DOCUMENTS 4* and that no users can login.

18. Logbook

This area manages *logbook* entries in which specific actions are logged. These actions to be logged are defined in two other places. Initially, on the *Documents-> Settings* menu, you generally define which global actions should be saved in the logbook. In the Settings dialog, the *Logbook* tab (Fig. 101) lets you define via various checkboxes which actions should be logged.

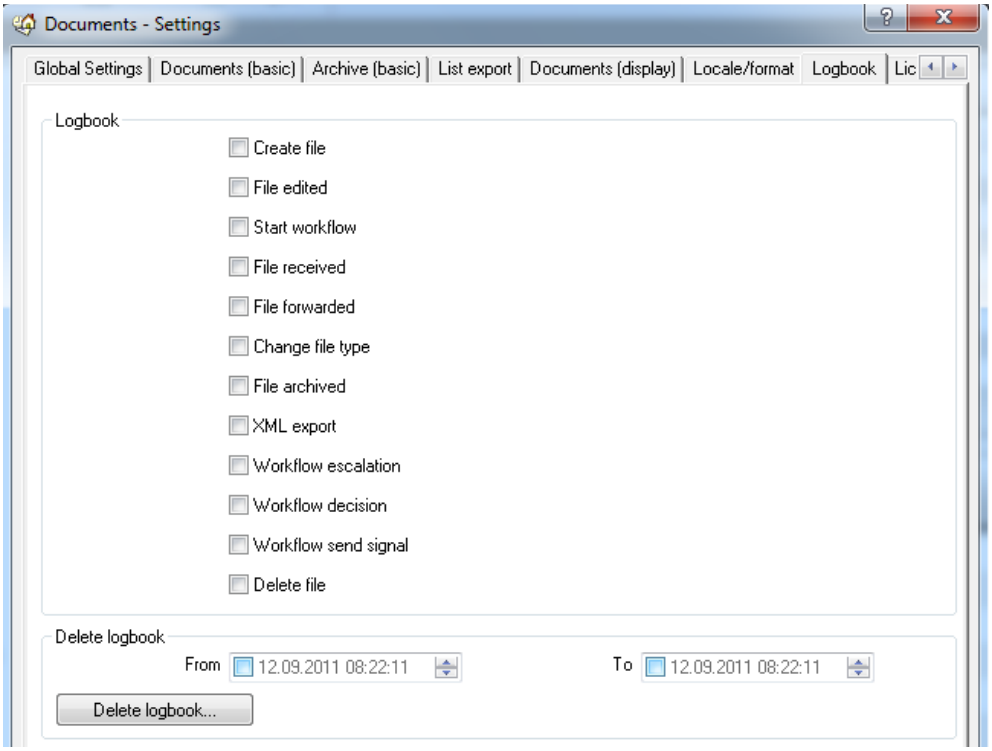

Fig. 101: Logbook entries of the principal

The *Delete logbook* area provides the option to remove the entries of a desired time period. First, determine the desired time interval, and then click the *Delete logbook* button.

Moreover, you can add the logbook entries at the level of individual processes. To do this, open the definition dialog for a file type, and go to the *Workflow* tab (Fig. 102). The *Logbook entry* field lets you define fields whose changes are to be logged in the logbook. Like auto titles of DOCUMENTS files, you can include field values here via the familiar replacement flags (`%NAME%`), and combine these with static texts. Including an ID or auto title comes in useful here to enable identifying the DOCUMENTS file in the logbook for which an action was performed.

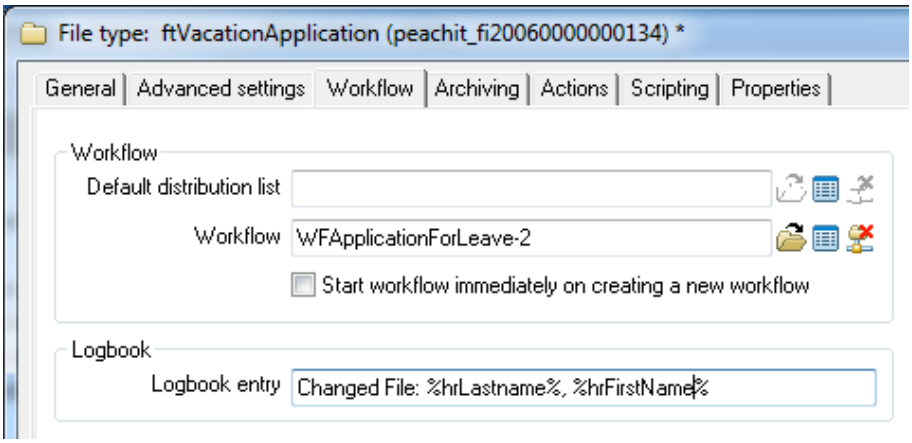

Fig. 102: Logbook entry for a field value

18.1 Enabling the logbook function

Statistical evaluations in DOCUMENTS 4 can be executed directly on the database. The data relevant to this is saved in the `DlcLogBook` table.

Unlike the status of a DOCUMENTS file, the information in the log table is retained even after deleting a DOCUMENTS file. You can perform statistical evaluations on lifetime and iteration of workflows.

You can also view the entries in the log table in the DOCUMENTS Manager anytime (Fig. 103). Entries may also be deleted there.

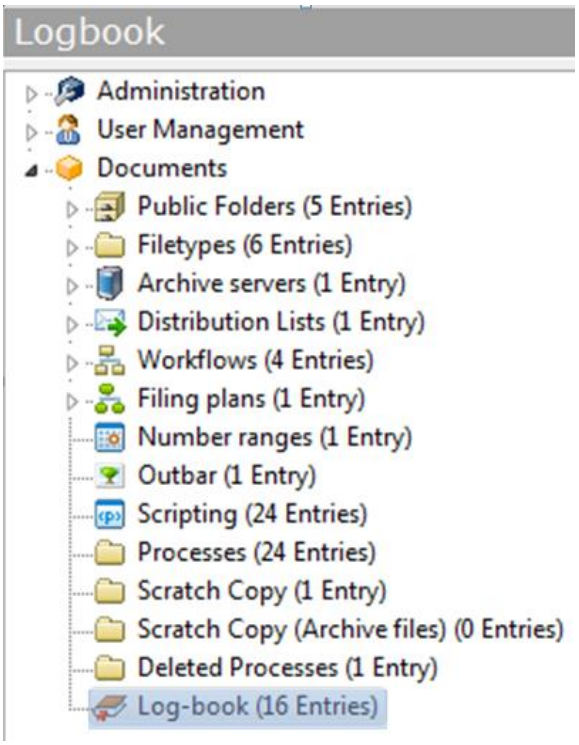

Fig. 103: Logbook position within the tree

However, a default tool for evaluating the log table is **not** available. Evaluations can be performed using various tools with access to ODBC data sources and SQL. We will later introduce a sample in Microsoft Excel in which the options of an evaluation are demonstrated.

18.2 Structure of the DlcLogBook table

The `DlcLogBook` table includes the following columns:

Name	Type	Description
m_oid	int	System column (Unique record ID)
c_ts	datetime	System column (creation timestamp)
m_ts	datetime	System column (modification timestamp)
c_user	int	System column (creation user)
m_user	int	System column (modification user)
principal	int	System column (principal)
Ts	datetime	Timestamp of creation of the entry
UserLogin	varchar(60)	User login of the responsible user (user who has edited the DOCUMENTS file, initiator of a workflow step, etc.)
TitleFile	varchar(255)	Title of the relevant DOCUMENTS file
IdFile	varchar(40)	Unique ID of the relevant DOCUMENTS file
FieldValues	varchar(255)	A logbook entry defined for the file type
TitleFileType	varchar(255)	Name of file type to which the DOCUMENTS file belongs
IdFileType	varchar(40)	Unique ID of file type to which the DOCUMENTS file belongs
IdWorkflow	varchar(40)	Unique ID of routing or workflow. All entries of a routing or workflow are allocated the same ID. This facilitates differentiating the individual routings when sending DOCUMENTS files multiple times.
MainContextName	varchar(60)	Name of workflow or distribution list that the DOCUMENTS file iterates through (e.g. purchase invoice)
IdMainContext	varchar(40)	Unique workflow / distribution list ID
IdStep	varchar(40)	Unique ID of the Step group. Example: If a

		group is assigned a task within a workflow, all individual steps will have the same IdStep.
ContextName	varchar(60)	Name of workflow step, e.g. Verify correctness
IdContext	varchar(40)	Unique workflow step ID
ActionCode	int	Which action was executed (table, see below) (ID)
ActionDescription	varchar(60)	Which action was executed (table, see below) (Description)
ActionDetail1	varchar(25)	Additional information. For example, comment on forwarding, etc.
ActionDetail2	varchar(60)	Additional information
ActionDetail3	varchar(60)	Additional information

The following actions are currently differentiated:

ActionCode	ActionDescription	Description
0	Create file	DOCUMENTS file was created
1	Start edit file	Edit mode was started
2	Cancel edit file	Edit mode was canceled
3	Save file	Edit mode terminated by saving
4	Archive file	DOCUMENTS file was archived
5	Delete file	DOCUMENTS file was deleted
6	Start workflow	DOCUMENTS file was sent / workflow was started
7	Receive file	A user has received the DOCUMENTS file as part of a routing / workflow
8	Forward file	A user has forwarded the DOCUMENTS file as part of a routing /workflow
9	Change filetype	The DOCUMENTS file's file type was changed
10	Workflow escalation	A workflow step escalated as part of the workflow
11	Workflow decision	The DOCUMENTS file iterated an automatic decision as part of the workflow
12	XML export	The DOCUMENTS file was exported as an

		XML file as part of the workflow
13	Workflow receive-signal	The DOCUMENTS file has received a receive signal as part of the workflow
14	Workflow finished	The workflow finished

18.3 Logbook sample

The `samples` directory contains a Microsoft Excel file including vb macros as a statistical evaluation sample (`DlcLogBook_Sample.xls`).

To be able to use the sample, you will have to create an ODBC data source with the name `Partnernet` on your local computer which will point to the corresponding database.

On the first sheet (Report) you define the parameters to be evaluated, defining which ODBC data source you want to access from which server (Fig. 104).

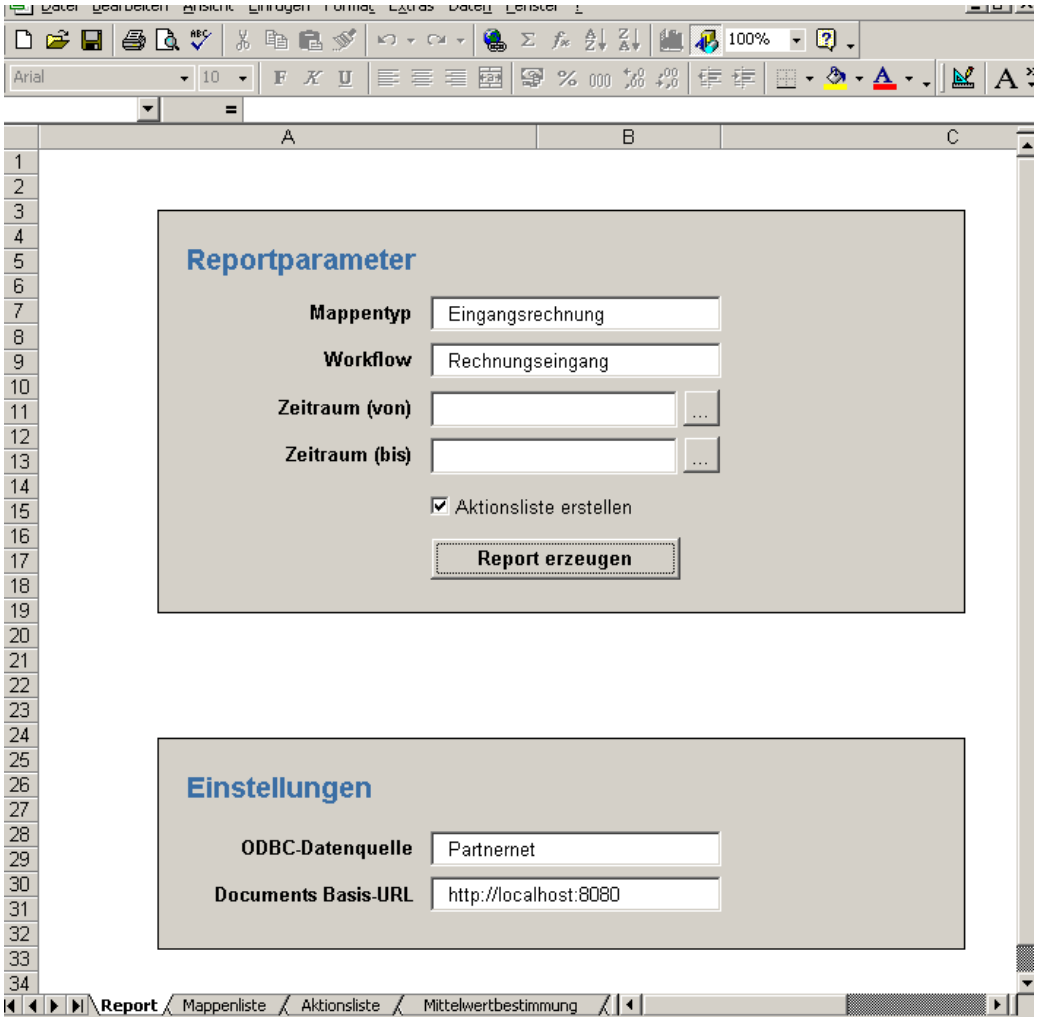

Fig. 104: Report in Microsoft Excel

Three evaluations are made. The second sheet displays the list of DOCUMENTS files with finished workflow as shown in Fig. 105:

	A	B	C	D
	Startzeit	**Endzeit**	**Laufzeit (h)**	**Mappentitel**
	20.08.2004 15:13:57	20.08.2004 15:16:37	0,044	Rech. von SQC, 726,16 Euro vom 24.02.2004
	20.08.2004 15:13:58	20.08.2004 15:15:50	0,031	Rech. von S_MUELLER, 15,90 Euro vom 13.01.2004

Fig. 105: File list of the report

The link in the file title allows navigating directly to DOCUMENTS 4.

The third worksheet (Fig. 106) displays all workflow steps belonging to the workflow and the file type:

	A	B	C	D	E	F	G	H
1	**Aktion**	**User**	**Startzeit**	**Endzeit**	**Laufzeit (min)**	**Mappentitel**		
2	Verteilung	local	20.08.2004 15:13	20.08.2004 15:14	0,900	Rech. von OKIA, 2340,87 Euro vom 24.02.2004		
3	Verteilung	local	20.08.2004 15:13	20.08.2004 15:14	0,817	Rech. von SQC, 726,16 Euro vom 24.02.2004		
4	Verteilung	local	20.08.2004 15:13	20.08.2004 15:14	0,700	Rech. von S_MUELLER, 15,90 Euro vom 13.01.200		
5	Verteilung	local	20.08.2004 15:15	20.08.2004 15:15	0,167	Rech. von SQC, 726,16 Euro vom 24.02.2004		
6	Freigabe	schreiber	20.08.2004 15:15	20.08.2004 15:15	0,467	Rech. von SQC, 726,16 Euro vom 24.02.2004		
7	Freigabe	schreiber	20.08.2004 15:15	20.08.2004 15:15	0,267	Rech. von S_MUELLER, 15,90 Euro vom 13.01.200		
8	Freigabe	oppen	20.08.2004 15:15	20.08.2004 15:16	0,683	Rech. von SQC, 726,16 Euro vom 24.02.2004		
9	Sachprüfung	schmidt	20.08.2004 15:15	20.08.2004 15:15	0,250	Rech. von SQC, 726,16 Euro vom 24.02.2004		
10	Sachprüfung	schmidt	20.08.2004 15:14	20.08.2004 15:15	0,900	Rech. von S_MUELLER, 15,90 Euro vom 13.01.200		
11	Sachprüfung	schreiber	20.08.2004 15:14	20.08.2004 15:15	0,283	Rech. von SQC, 726,16 Euro vom 24.02.2004		
12	Sachprüfung	oppen	20.08.2004 15:14	20.08.2004 15:16	1,833	Rech. von OKIA, 2340,87 Euro vom 24.02.2004		
13								

Mappenliste | **Aktionsliste** | Mittelwertbestimmung | Grafik

Fig. 106: Workflow steps

The runtime (time) of the step with the user was computed from the access and forwarding time.

Sheets 4 and 5 compute averages of the times and graphically display the computations as a pie chart (Fig. 107).

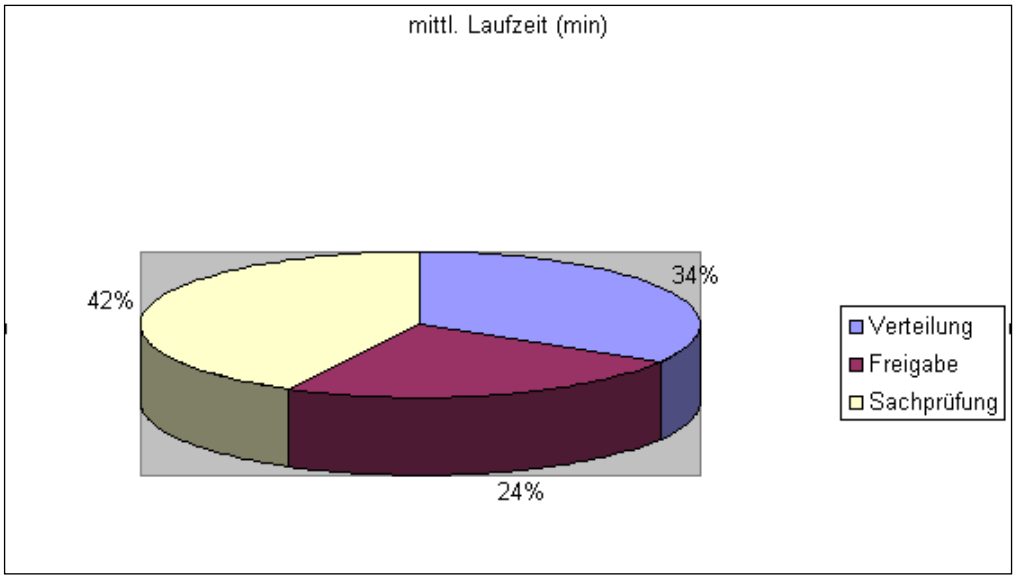

Fig. 107: Graphic evaluation

19. Drag & Drop Support (Dropzone)

The *Dropzone* is an *ActiveX* component that facilitates uploading documents via *Drag & Drop* in DOCUMENTS 4. When using the *Dropzone* all document tabs of the managed file types include, in addition to the popular *Upload* dialog for documents, a visually highlighted panel where documents can be stored using the mouse.

This function must initially be globally activated for the principal. To do this, in the DOCUMENTS Manager, open the "*Documents -> Settings*" menu item.

In the dialog, go to the *Global Settings* tab and enable the *Enable Drag & Drop* checkbox (Fig. 108).

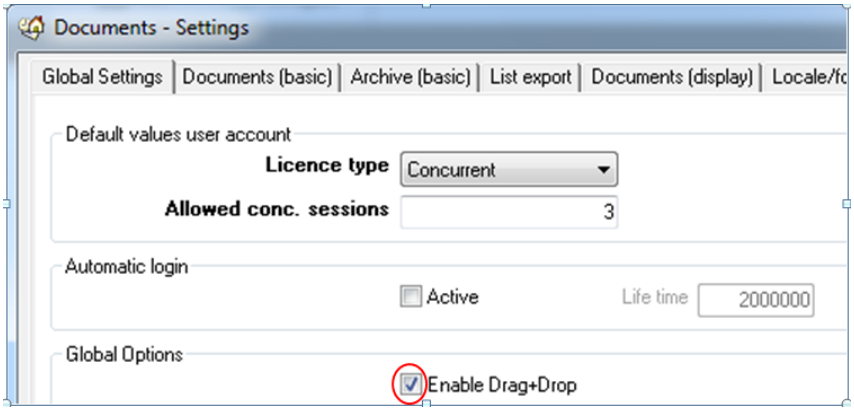

Fig. 108: Activating a Dropzone

After initialization or restarting the `Documents4Server` service, the *Drag & Drop* function can be used.

> *The Dropzone reverts to an ActiveX Control. This function can therefore be used only in combination with Microsoft Internet Explorer. Installation and execution of ActiveX components must be guaranteed via the browser's security settings.*

On the users' computers, the ActiveX component must be subsequently installed locally prior to initial use.

To do this, log in to DOCUMENTS 4 and edit a DOCUMENTS file that has a document tab. When you click that tab, the browser will prompt you to accept installation of a new ActiveX component (Fig. 109). This component supplies Drag & Drop functionality; accordingly, the feature can only be used when you accept installation of this component.

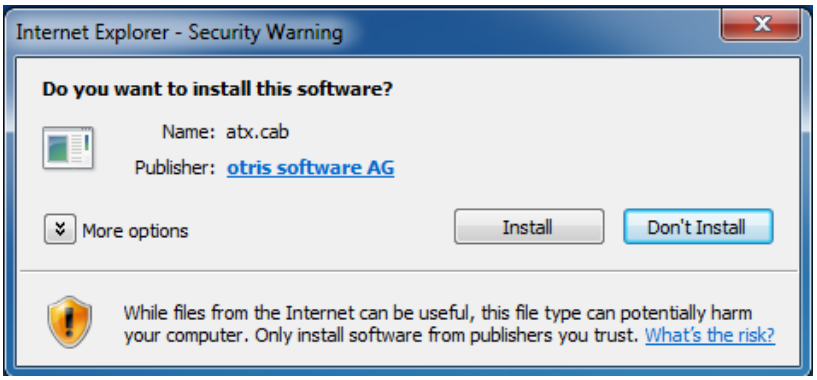

Fig. 109: Installing the Dropzone component

After successful completion of the installation, the Dropzone will be displayed on all document tabs, regardless of the respective file type (Fig. 110).

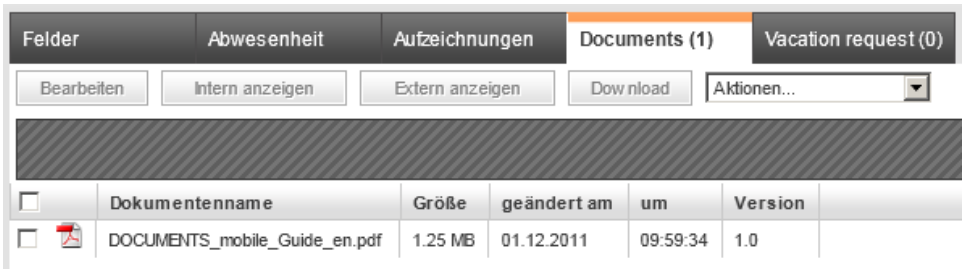

Fig. 110: Dropzone in DOCUMENTS 4

You can now upload documents directly from Explorer. Select the corresponding data files and drag them to the hatched area of the Dropzone, keeping the left mouse button depressed. If you release the mouse button, the documents will be automatically uploaded. To finally complete this process, you need to save the DOCUMENTS file.

Downloading a document to your local hard disk works exactly the same in reverse. To do this, simply drag the data file to be downloaded into a target directory that is open in Windows Explorer.

On dropping it, the data file is automatically downloaded to the target directory.

20. Menus

The menu bar's top section provides actions for administrative activities and for configuring parameters.

20.1 Application

Change password

For security reasons, please immediately specify a new password for the user `admin`. To do this, open the "*Application -> Change Password*" menu item (Fig. 111).

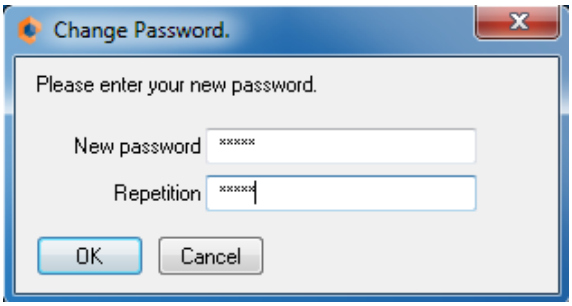

Fig. 111: Specify a password for the Portal Manager

Language selection English / German

The working language of the DOCUMENTS Manager can be switched at runtime. Alternatively, this can be done directly on login.

Recently used items

The recently used items are listed in a separate panel. When clicking an entry the item directly opens without the user having to search it in the tree structure.

Finish

After you finish working with the DOCUMENTS Manager, you close the application via the "*Exit*" entry from the "*Application*" menu.

20.2 Server settings

Server-side data export (jex)

This command generates principal backup as a JEX file. Unlike Portal backup from the Administration menu (see chapter 20.3), no documents are exported here. The JEX file only includes the structures from the DOCUMENTS Manager. To perform backup, you need to specify the file name and a storage path (Fig. 112).

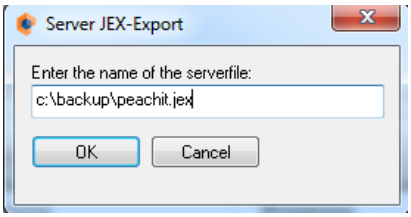

Fig. 112: Server-side JEX export

Server-side data import (jex)

To back up JEX export, please specify the complete path to the JEX file.

XML export

Specific DOCUMENTS Manager elements can be exported as an XML file by stacks when specifying an *export class*. The export class is named in this format:

```
ExportClass.xml
```

You will then have to specify a path to the store of the generated data file.

The following table shows an overview of the most important export classes and the exported objects:

Export Class	Description
`Fellow`	All editors
`DistributionList`	All distribution lists
`DlcFileType`	All file types
`DlcFile`	All processes
`DlcFileArchive`	All archive schemas / views
`PortalScript`	All scripts
`AccessProfile`	All access profiles
`Workflow`	All workflows
`DlcAlias`	All aliases
`Documents`	All elements of the following types: access profiles, aliases, scripts, file types, archives, folders, workflows, and number ranges

XML import

This command imports objects that have previously been exported via XML. This is limited not just to objects exported via the *XML export* command; individual objects such as file types, individual scripts, etc. are also considered. The configuration dialog of many of the objects also provides an XML export function via a button. Whereas import is performed centrally via the menu function described here. Thus, for instance, you can import scripts from one principal to another.

System parameters

This command opens a dialog including *system parameters*. Select the desired parameter from the top drop-down list. You can customize the value in the center pane. The bottom section of the dialog provides an explanation of the parameter's functionality. Fig. 113 shows the *server directory of the DOCUMENTS repository*, i.e. the storage location of uploaded documents on the server, in an example.

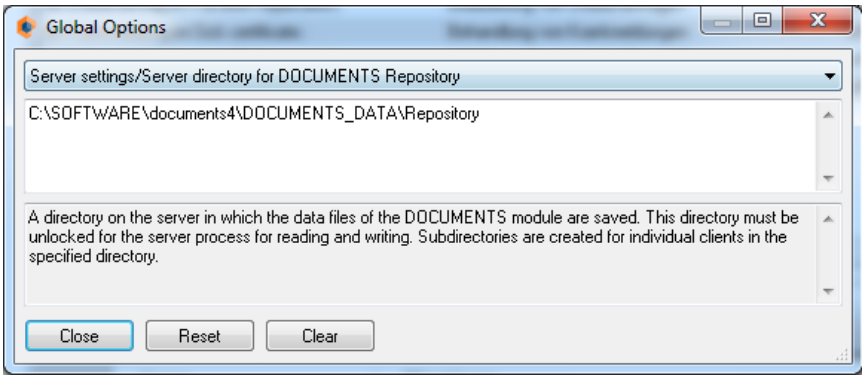

Fig. 113: System parameter including description of the function

Make sure that when changing entries you do not finish these by pressing the ENTER key. The actual entry may not be followed by another line. Otherwise, the parameter's interpretation fails.

20.3 Administration

20.3.1 Run maintenance operation

DOCUMENTS 4 provides a host of maintenance operations that allow checking specific system functions or changing them by stacks. To execute such an operation, you need to enter its ID in the dialog window. The following overview describes the most important maintenance operations:

`Build AclCache <file type name` OR `id>`

If you subsequently provide file class protection for a file type or change an existing one, you can rebuild the ACL cache for all DOCUMENTS files of that type using this operation.

`checkBinaries`

Checks the repository for missing attachments or broken links. Output comes in file format in the `\server` directory of the installation in the form of a table including error outputs.

`checkIndexBlob [filepath]`

A check is made on whether full text indexing of documents works error-free. When using the `filepath` option, you need to specify a server-side path to an indexable test file.

Without specifying the `filepath` option, a temporary text file will be generated and its indexing will be tested. This data will be deleted after the test.

`inboxEmailOff`

This operation turns *off* automatic e-mail notification for new DOCUMENTS files in the Inbox by stacks for all users.

`inboxEmailOn`

This operation turns *on* automatic e-mail notification for new DOCUMENTS files in the Inbox by stacks for all users.

`initDLC`

Grants all editors and users DOCUMENTS 4 access by stacks. This operation will be useful if you have previously created simple archive users who should be given DOCUMENTS access from a specific time onwards.

`lockNumberRanges`

This operation locks all number ranges (on number ranges, see chapter 15); it is above all of significance in conjunction with the *Change files* action of a file type (see chapter 17).

If a file type is using number ranges, the corresponding number range will be incremented by one digit on creating a new DOCUMENTS file. On using the *Change files* action, a new DOCUMENTS file is created by stacks for each existing one and the values of the old file are then transferred to the new one by stacks. This inevitably causes the number range to be always incremented on creating new DOCUMENTS files by stacks. So, a gap of the already existing number of DOCUMENTS files emerges in the number range through this action. The number range is ultimately used on creating new DOCUMENTS files, and then overwritten by the value of the existing file.

For this reason, you should always execute the *lockNumberRanges* operation prior to the *Change files* action if the file type is using a number range.

`unlockNumberRanges`

This operation unlocks all number ranges of the system.

`reIndex`

Rebuilds the full text index of all attachments / documents. Performing this operation may be necessary for two reasons:

- If a*utomatic indexing of documents on uploading* has not been enabled for a file type, the full text index can be built by stacks in this way.
- On performing Portal backup, the full text index might not be included with backup due to its possible size on the database. When reversing backup, the index can be rebuilt using this operation.

`setEasywareAuth:accessProfileFilter:on|off`

Turns EASYWARE authentication on or off for all users of the access profile.

The following example turns on EASYWARE authentication for all users of access profiles whose access profile name starts with `crm`:

```
setEasywareAuth:crm*:on
```

`showLicence`

Shows an overview of the existing license for the logged-in principal with information on licensed modules, plug-ins, and user licenses.

`testAccess`

This operation checks which permissions a user has on a specific DOCUMENTS file. Here all file permissions and an existing file class protection are considered. The following line shows the correct use of the notation:

```
testAccess <file id or filetype name> <login name>
```

The following example checks which permissions the *user* with the login name `schreiber` has on the file (the process) with the *id* `peachtit_fi20110000000046`:

```
testAccess peachit_fi20110000000046 schreiber
```

20.3.2 Perform portal backup

This command starts backup of the logged-in principal. This can either be backed up on the same system, or transferred to another system. The following data files are written on performing backup:

- *Normal contents*: The complete database (except indexing documents) is stored in the following data file: `Principal_Timestamp.zip`.
- *Documents*: The storage path of the documents uploaded in this principal is saved in the `Principal_documents_Timestamp.zip`: file.
- *Secure contents*: Replaces the normal contents on using `https`.

Taken together, the data files for *Contents* and *Documents* represent the entire system.

The storage location for created Portal backups can be viewed and configure on the *Server settings -> System parameters* menu.

The data from the database management system and the document storage are only combined in a Portal backup. There is no check for data integrity. For this reason, these backups do not replace a complete concept for regular backup.

A *portal backup* is directly executed without further prompting; it may take several hours, depending on the size of the principal.

20.4 Documents

The *Documents* menu provides key *settings* in a dialog. The following chapters explain the most important settings options on the different tabs. Information on individual options can be displayed anytime via context help.

20.4.1 Global settings

The *Global settings* tab (Fig. 115) is used to initially define *default values for new user accounts*. The *license type* set here will be automatically suggested when creating new user accounts. The default value *Concurrent* is useful because this license type (if present) is not limited to a number of user accounts.

Users can use multiple sessions at the same time in DOCUMENTS 4, and therefore edit different processes in parallel. In Fig. 114 the button to open another session has red borders.

Fig. 114: Global functions

The *Global settings* tab lets you set how many concurrent sessions a user is allowed.

Auto-login provides, on the DOCUMENTS 4 login page, provides a new checkbox for permanent login. The user can now work without login during the *lifetime*.

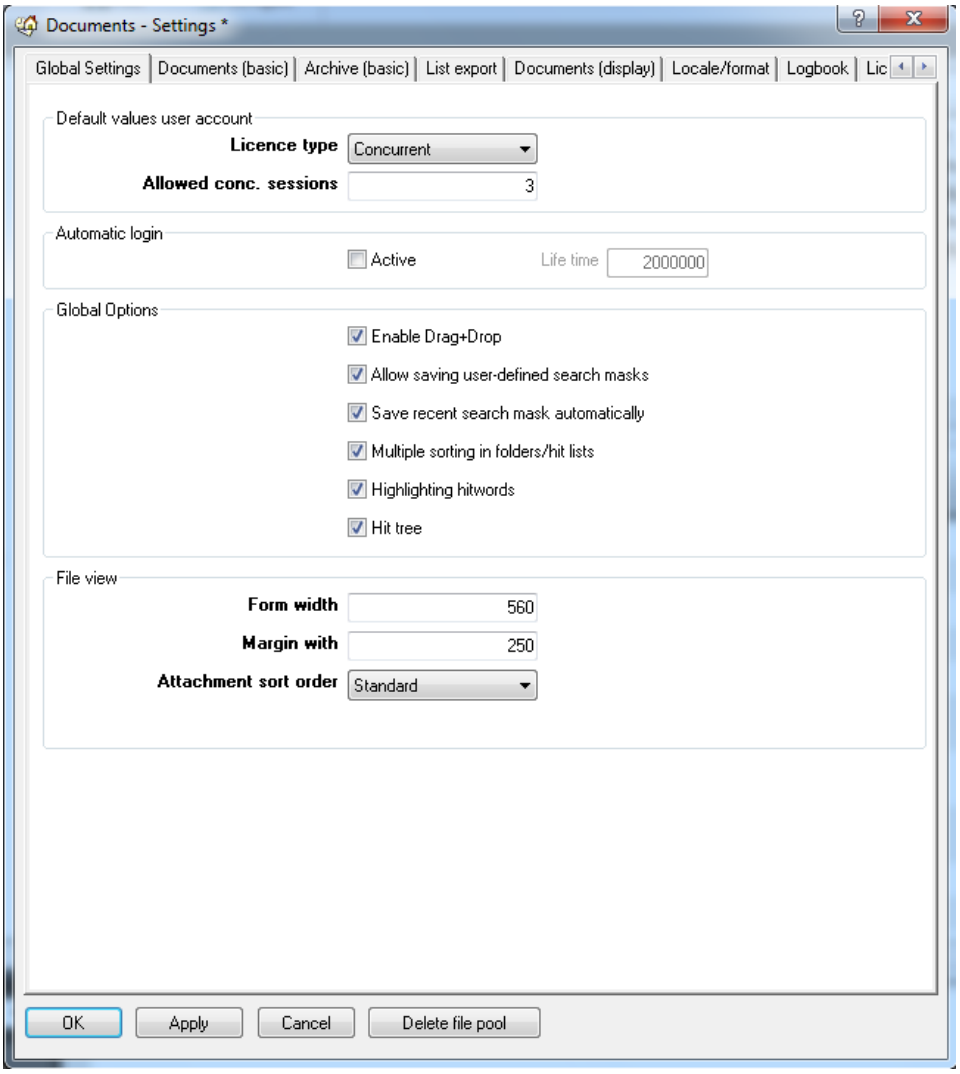

Fig. 115: Global DOCUMENTS settings

The following features can be activated in the *Global settings* area:

- *Enable Drag + Drop*: Activates the *Dropzone* (see chapter 19).
- *Allow saving user-defined search masks*: Users can save filter settings used in search masks at regular intervals, and open them when required.
- *Save recent search masks automatically*: The recently used filter criteria of a search are identically entered at the next call.
- *Multiple sorting in folders / hit list*: Grouping function in tabular overviews. The data is sorted by the column that is first marked and, within the result, by a second one, etc. column
- *Highlighting hitwords*: Highlights the hits found in the search result in color.
- *Hit tree*: Turns on the function, and creates a hit list tree outbar.

20.4.2 Documents (basic)

The *Encrypt repository* checkbox facilitates encryption in the storage directory of uploaded documents which can therefore not be opened outside of DOCUMENTS 4. This is an optional module subject to licenses. Encryption will therefore only be performed if a relevant entry is found in the license file.

20.4.3 List export

This tab allows configuring exports from DOCUMENTS 4 in the *CSV* or *XML* formats. Actions that optionally export all or only selected DOCUMENTS files in the desired format are provided for the active functions in folders, hit lists and search results.

20.4.4 Locale/format

The *Support multiple languages* checkbox initially generally enables multiple languages. When required, you can define for the enabled languages which *date format* and *decimal points* should be used, respectively, for display in DOCUMENTS 4.

Which languages are to be enabled is defined in the principal's configuration dialog (see chapter 5.2).

Subsequent changes to the *decimal points* are no longer permitted once DOCUMENTS files with numerical fields are present.

20.4.5 Logbook

Logging using the *logbook* is already described in detail in chapter 18.

20.4.6 Licenses

Provides a tabular overview of the *licenses*. This shows how many licenses are available, how many of them have already been allocated and are currently engaged.

20.5 Help

Besides the *Help* function, you can learn about version properties of the installed environment via the *About* entry.

21. Hit Trees

Hit trees can be optionally used in combination with the search function. For the search result, a tree structure is built that works in a similar manner to a file plan. A hit tree, however, is not subject to a fixed structure; it is dynamically built for the respective search result.

In their default behavior, search results are built as tabular hit lists and can therefore only be sorted by columns.

Hit trees additionally allow you to specifically group according to specific columns or field values, e.g. according to countries, regions, and cities. In many cases, this facilitates visualizing the search results to be significantly more transparent to users.

21.1 Basic settings in the DOCUMENTS Manager

To turn on the hit tree for a principal, you need to initially make two settings in the DOCUMENTS Manager.

Open the *Documents -> Settings* menu and, on the *Global Settings* tab, enable the *Hit tree* checkbox (Fig. 116).

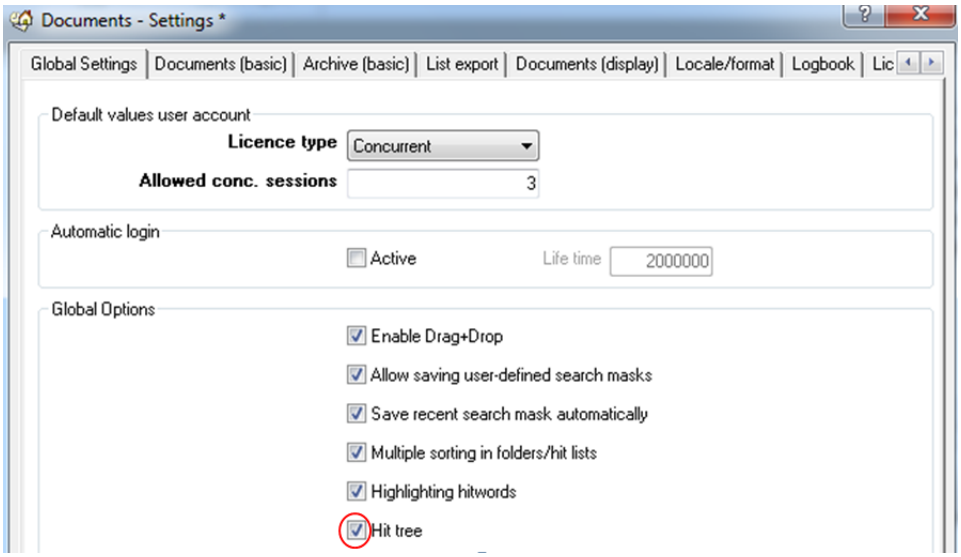

Fig. 116: Enable hit tree

Next, in the same dialog, go to the *Documents (basic)* tab and enable the *Search in multiple file types* checkbox (Fig. 117).

Fig. 117: Search in multiple file types

These customizations will only take effect when restarting the `DOCUMENTS4Server`. Alternately you can trigger reinitialization via `&pEvent=init` in the URL.

Overall, two variants are available for using hit trees: Either a link to a hit list can be used, or the hit tree is entered for the file type, including a fixed field structure.

21.2 Hit trees including hit lists

On the desired file type, create a new hit list (for hit lists, see chapter 9.4). You can optionally consider *all fields* here and correct them on the hit list afterwards, or you create a hit list that only includes fields where the *Show in hit list* property is enabled (Fig. 118). Make sure that the hit list contains all fields to be later displayed in the hit tree.

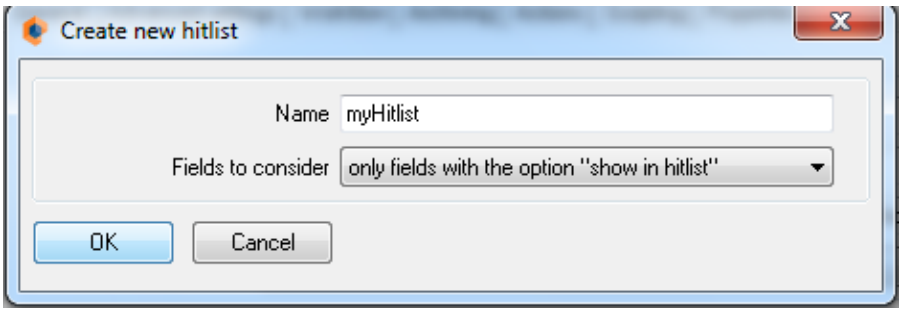

Fig. 118: Hit list for a hit tree

The following example of a hit list `myHitlist` is oriented to the file type `ftIivoice` (invoice) of the *peachIT* demo principal. Fig. 119 shows a possible structure of a hit list for use in the hit tree.

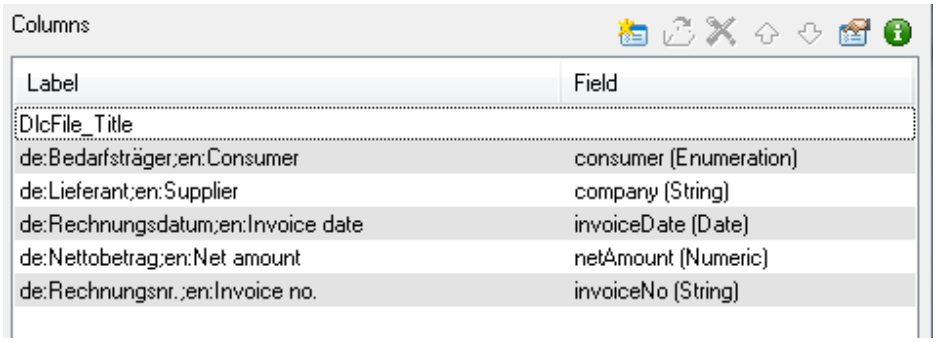

Fig. 119: Configuring a hit list

After creating the hit list, on the file type, you need to establish the link to the hit tree.

To do this, go to the *Properties* tab and right-click the dark section of the tab to create a new property with the following name:

```
treeSort_<Hit list name>
```

As a *value* of the property, the individual technical field names or columns to be used in the hit tree for sorting and grouping search hits must be listed comma-separated (without a following space). Through an additional "+" or "-" sign directly following the individual names, you can also influence whether the values in the hit tree are to be sorted in ascending or descending order. In any event, the corresponding hit list must also contain the specified columns.

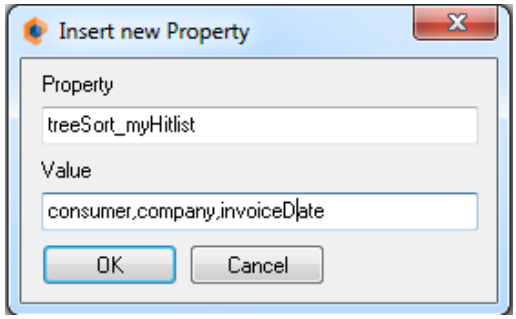

Fig. 120: Property of the file type

21.3 Displaying a hit tree in DOCUMENTS 4

Log in to DOCUMENTS 4 and start an advanced search. To do this, click the icon with the magnifying glass above the tree structure.

Next, choose the desired file type (in this example, this is *Invoice*) and make sure that you define this as the *main file type*.

This is achieved by simply clicking the name (not the checkbox!). The name will then be highlighted in bold and the search masks and hit lists valid for this file type will be shown in an enumeration list next to one another above the search form.

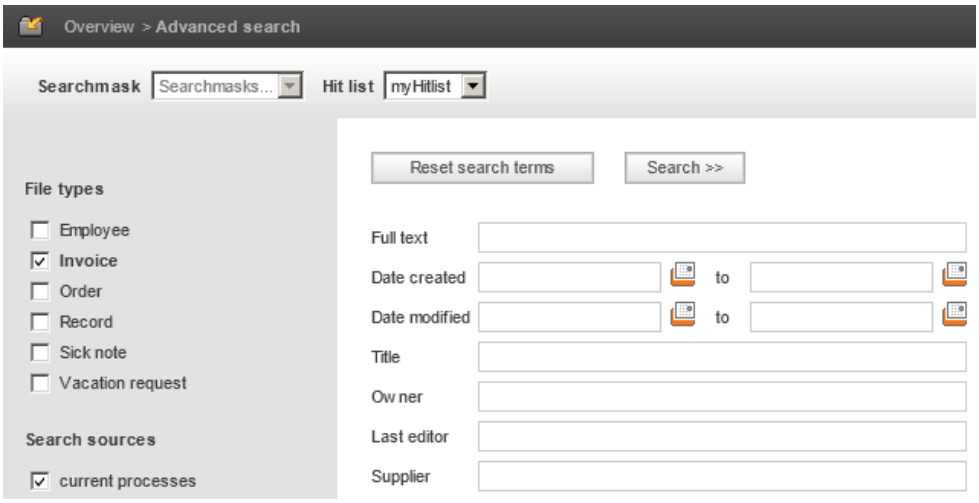

Fig. 121: Advanced search via hit list

When filling out the form with any search terms, you can already select the hit list via which a hit tree is to be generated. However, the hit list can also be switched while displaying the search result.

After initiating the search, the results will be displayed in the defined hit list (Fig. 122). If the hit list matching the hit tree has not previously been selected, you will have to do this now.

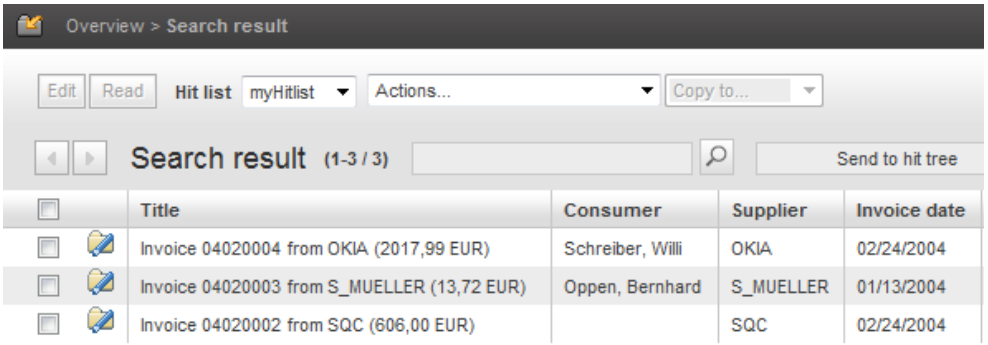

Fig. 122: Search result with hit list

Next, click the *Send to hit tree* button to generate it. In the tree pane, the switch to the new *Hit tree* outbar is made automatically (Fig. 123).

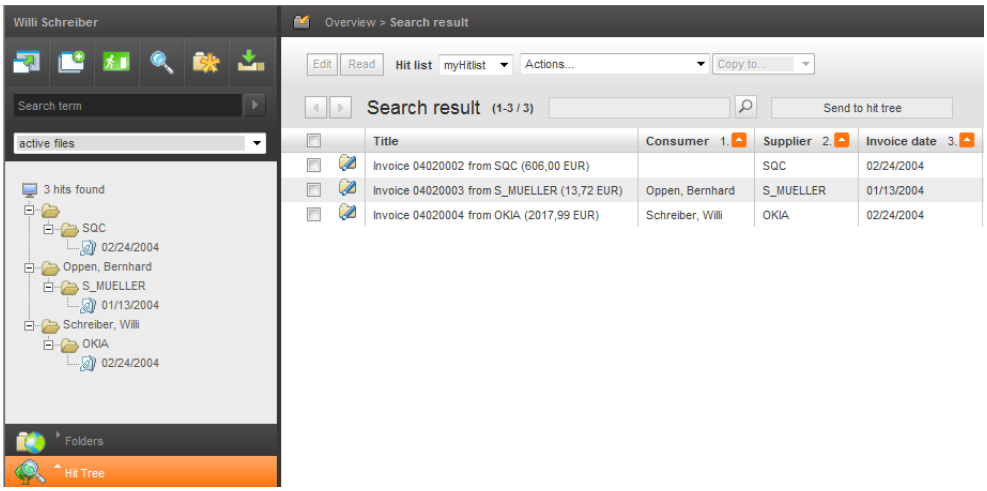

Fig. 123: Hit tree with grouped search result

In this example, the found invoices are initially grouped according to the *consumer*, then according to the *vendor* and finally according to the *invoice date*. The invoice date is entered at the bottom level, the so-called sheets. Clicking such a sheet directly opens the relevant DOCUMENTS file in the right pane.

The sort order of the search results can be reversed by the red icons on the hit list columns.

21.4 Hit trees without hit list

Hit trees can also be created without direct links with a hit list. For this, on the file type's *Properties* tab, you need to create a new property with the name `treeSort`.

As a *value* of the property, the individual technical field names or columns to be used in the hit tree for sorting and grouping search hits must again be listed comma-separated (without a following space). An additional "+" or "-" sign directly following the individual names can also be used here to specify the sort order.

21.5 Automatic hit tree

In some cases it might be a useful idea to automatically create a hit tree directly after an advanced search request. To achieve this, in the DOCUMENTS Manager, the *Documents /Settings* menu item's *Properties* tab lets you enter a principal-wide valid property named `impliedTreeSearch` with the value `true` (Fig. 124).

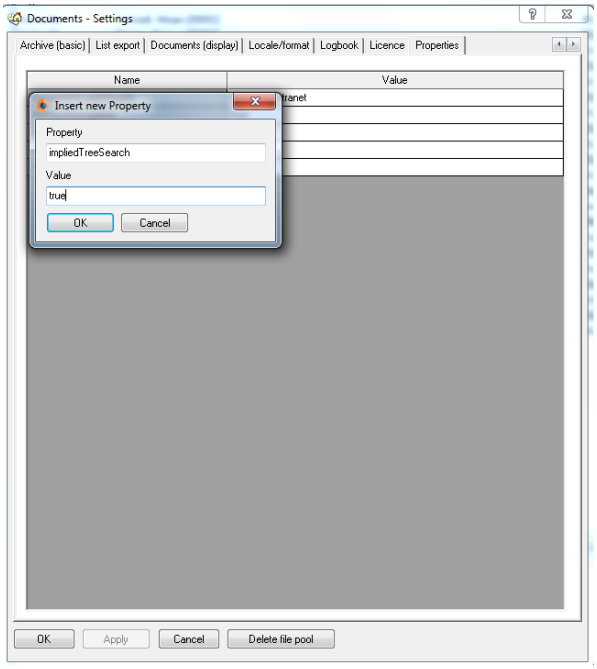

Fig. 124: Automatic hit tree

You will then have to restart the `Documents4Server` service or initialize the URL.

21.6 QuickView

To be able to use the hit tree in Quickview, append the `&_tv=1` parameter to the URL. Unlike normal hit tree display, Quickview directly opens the first DOCUMENTS file in the section on the right (Fig. 125).

Fig. 125: Hit tree in QuickView

22. Table of Figures