User and Solution Administration

Alfabet Reference Manual

Documentation Version Alfabet 10.15.3

Conventions used in the documentation

Convention	Meaning
Bold	Used for all elements displayed in the Alfabet interface including, for example, menu items, tabs, buttons, dialog boxes, page view names, and commands. Example: Click **Finish** when setup is completed.
Italics	Used for emphasis, titles of chapters and manuals. this Example: see the *Administration* reference manual.
Initial Capitals	Used for attribute or property values. Example: The object state Active describes...
All Capitals	Keyboard keys Example: CTRL+SHIFT
File > Open	Used for menu actions that are to be performed by the user. Example: To exit an application, select **File > Exit**
< >	Variable user input Example: Create a new user and enter <User Name>. (Replace < > with variable data.)
	This is a note providing additional information.
	This is a note providing procedural information.
	This is a note providing an example.
	This is a note providing warning information.

Table of Contents

Chapter 1: Introduction to the User and Solution Administration Functionalities

This reference manual provides information relevant for the Alfabet functionalities supporting user administration as well as other administrative tasks and configuration relevant for your Alfabet solution. The functionalities described in this reference should typically be accessed by users with an administrative user profile and should not be accessible to the general user community.

For security reasons, bookmarks cannot be created for administrative views. The **Create Bookmark** option will not be available for views that cannot be bookmarked.

The following information is available:

- Re-Enabling an Automated Assistant for a User
- Configuring and Managing Questionnaires
- Configuring the AlfaBot Capability
- Scheduling ADIF Jobs and Batch Jobs via the Job Schedule Functionality
- Managing Data Quality
- Reviewing and Correcting the Original Vocabularies

Chapter 2: Defining and Managing User Profiles

The **User Profiles Administration** functionality allows you to specify user profiles, add users to a user profile, and manage access to configured reports for a user profile.

User profiles are the basis of user administration in Alfabet and serve as the entry point when accessing Alfabet. Every user must log in with a user profile that has been assigned to him/her by a user administrator. Therefore, all users accessing Alfabet must be assigned at least one user profile. However, users may possess multiple user profiles in accordance with their responsibilities in the user community and in the enterprise as a whole. A user can switch to another permissible user profile at any point during a user session.

A user profile specifies the Alfabet functionalities available to a user, the visibility and editability of object classes and object class attributes, as well as the availability of associated capabilities including, for example, wizards and workflows.

The following should be defined for each user profile that has been created:

- The access permissions of the users logging in with the user profile.
- The view scheme that specifies the availability of functionality and visibility of data in the user profile. The view scheme must first be configured in the configuration tool Alfabet Expand by your solution designer. For more information, see the section *Configuring Class Settings for Object Classes and Object Class Stereotypes* in the reference manual *Configuring Alfabet with Alfabet Expand*.
- The users that have access permissions to the user profile.
- The configured reports that can be accessed by users logging in with the user profile.

If your enterprise is working with an external repository (for example, LDAP), the external ID can be captured for user profiles via the alternative editor `UserProfileWithExternalID_Editor`. This editor should be specified in the **Edit View** attribute for a custom class setting for the class `ALFA_USERPROFILE`.

The **User Profile Manager** explorer displays all user profiles  that have been created in either the **User Profiles Administration** functionality or the configuration tool Alfabet Expand. Click the **User Profile Manager** icon  to view and edit all existing user profiles or create new user profiles. The **All User Profiles** view displays the following information:

- **Name**: The name of the user profile.

  The preconfigured user profile `Admin` cannot be edited or deleted! This administrative user profile is necessary to provide full access to Alfabet functionalities and objects for user administrators and solution administrators. Please note that a user logged on with an administrative user profile has always edit permissions to all objects. The access permissions for an administrative user profile as well as a user defined as a mandate master take precedence over any other access permission concepts in Alfabet. A user is defined as a mandate master in the **User** editor.

- **Type**: The edit permissions defined for the user profile.
- **Use for Anonymous Users**: Displays a checkmark if the user profile is used by users of the type `Anonymous` to access Alfabet.
- **Guide Page**: The guide page or guide view displayed as the start page when Alfabet is accessed with the user profile.

- **View Scheme**: The view scheme that applies to the user profile when accessed by an external application.
- **Description**: A description providing information about the purpose of the user profile.
- **Administrative**: Displays a checkmark if the user profile is an administrative user profile.

Once the user profile has been created in the **User Profiles Administration** functionality, it will also be available in the **Admin** tab in the configuration tool Alfabet Expand. A custom online help can be assigned to the user in the context of the configuration tool Alfabet Expand. For more information, see the section *Making Functionalities Accessible to a User Profile* in the reference manual *Configuring Alfabet with Alfabet Expand*.

The names of user profiles will be available in the METAMODEL vocabulary and can be translated in the context of the vocabularies. The translated user profile name will be displayed in the masthead of the Alfabet user interface as well as in the **Change User Profile** menu in the <Alfabet User Name> menu in the main toolbar. Please note however that the translation of the user profile name will not be displayed in the **User Profiles Administration** and **User Administration** functionalities in the Alfabet user interface as well as the **User Profiles** node in Alfabet Expand. For more information about translating strings in the vocabularies, see the section *Modifying, Translating and Managing the Vocabularies* in the reference manual *Configuring Alfabet with Alfabet Expand*.

The following information is available:

Creating a User Profile

User profiles can be created in the **User Profiles Administration** functionality or the configuration tool Alfabet Expand. All existing user profiles regardless of where they are created are displayed in the **User Profiles Administration** functionality.

1) Go to the **User Profiles Administration** functionality and click the **User Profile Manager** icon  at the top of the explorer.
2) In the **All User Profiles** view, click **New** > **Create New Profile**. The **User Profile** editor opens.
3) Enter the necessary data, as needed:

   General tab:

   - **Name**: Enter a caption for the user profile. This is the name of the user profile that users will see when logging in to Alfabet.

     The names of user profiles will be available in the METAMODEL vocabulary and can be translated in the context of the vocabularies. The translated user profile name will be displayed in the masthead of the Alfabet user interface as well as in the **Change User Profile** menu in the <Alfabet User Name> menu in the main toolbar. Please note however that the translation of the user profile name will not be displayed in the **User Profiles Administration** and **User Administration** functionalities in the Alfabet user interface as well as the **User Profiles** node in Alfabet Expand. For more information

about translating strings in the vocabularies, see the section *Modifying, Translating and Managing the Vocabularies* in the reference manual *Configuring Alfabet with Alfabet Expand*.

- **Type**: If the user profile should have editing permissions, select `Read/Write`. If the user profile should only have viewing permissions, select `ReadOnly`.

  A warning message will be displayed in the **User Profile** editor if a user profile is specified as an administrative user profile and the **Type** attribute is set to `ReadOnly`.

- **View Scheme**: Click the **Drop-Down** button to select the relevant view scheme that applies to the user profile when accessed by an external application or via a hyperlink in an e-mail notification. The view scheme must first be configured in the configuration tool Alfabet Expand by your solution designer. For more information, see the section *Configuring Class Settings for Object Classes and Object Class Stereotypes* in the reference manual *Configuring Alfabet with Alfabet Expand*.

- **Guide Page**: Select an existing guide page or guide view from the drop-down list to assign it as the start page that is displayed when Alfabet is accessed with the user profile. The guide page/guide view contains links to Alfabet functionalities, thus defining the scope of access to functionalities for the user logged in with the user profile. Guide pages and guide views are configured with the tool Guide Pages Designer and must be uploaded to the Alfabet database in order to be available in the drop-down list in the **User Profile** editor. For more information about how to create, edit, and upload guide pages and guide views, see the reference manual *Designing Guide Pages for Alfabet*. For more information about whether to assign a guide page/guide view to the user profile or whether to configure access via menus, see the section *Making Functionalities Accessible to a User Profile* in the reference manual *Configuring Alfabet with Alfabet Expand*.

  The guide pages/guide views that are accessible to the user profile are specified in the **Details** tab.

- **Workflow Template**: Specify the workflow template that should be triggered for a user requesting assignment to this user profile. The workflow will be triggered when the user selects the user profile via the **Assign User Profile** option in the < **User Name** > menu in the Alfabet user interface. The checkbox **Enable User Self Administration** must be selected for the user profile request functionality to be activated.

- **Device Type**: Specify which type of device may be accessed by the user profile. Select `Browser` to indicate that Alfabet may only be accessed by the user profile in a Web-based browser, `App` to indicate that Alfabet may only be accessed by the user profile in the Alfabet Mobile Portfolio Manager which is available for mobile devices or `Unspecified` to indicate that Alfabet may be accessed by the user profile in both a Web-based browser an the Alfabet Mobile Portfolio Manager. Please note that if the user profile will access Alfabet on mobile devices, it is recommended that the visual layout of the object cockpits, guide pages/guide views, etc. Is optimized for mobile devices.

- **GUI Scheme**: The drop-down menu displays all GUI schemes configured by your enterprise. Select the GUI scheme that should be used to visualize the Alfabet interface. The configuration of GUI schemes is described in the section *Configuring GUI Scheme Definitions for the Alfabet Interface* in the chapter *Configuring the Visualization of the Alfabet Interface*. For more information about configuring user profiles for barrier-free accessibility, see the section *Configuring Barrier-Free User Profiles*.

- **Description**: Enter a description about the user profile.

- **Use for Anonymous Users**: Select the checkbox if the user profile should be used as the user profile that is used for anonymous users. You should only select the checkbox if you have defined the value `ReadOnly` for the **Type** attribute. For more information about anonymous access, see the section *Configuring User Authentication* in the reference manual *System Administration*.

  If you attempt to define a user profile of the type `Read/Write` as the user profile for anonymous users, the definition will be ignored. If you define more than one user profile as the user profile for anonymous users, the first user profile in the alphabetical order of user profiles will be used as the default user profile for anonymous users. All other definitions will be ignored. For information about how to define a Alfabet user of the type `NamedUser` or `Anonymous`, see the section Defining and Managing Users in the reference manual *User and Solution Administration*.

- **Enable User Self Administration**: Select the checkbox if users may self-assign this user profile via the **Assign User Profile** option in the < **User Name** > menu in the Alfabet user interface. Using the self-administration capability helps decrease administrative efforts. For more information about the options for user profile assignment, see the section *Configuring User Profile Request or Assignment for the User Community* in the reference manual *Configuring Alfabet with Alfabet Expand*.

  Please note that if you are implementing the self-administration capability, any user may assign this user profile to him/herself. It is recommended that you carefully consider which users may be assigning themselves a user profile via the self-administration capability. This is especially critical when configuring administrative user profiles, which grants `ReadWrite` access permissions to ALL objects in Alfabet!

- **Use WAI-ARIA**: Select the checkbox if the WAI ARIA specification should be implemented for barrier-free accessibility. If the **Use WAI-ARIA** is selected, the following changes will be automatically implemented to support barrier-free accessibility:

  - Conventional tabular datasets will be replaced by flat datasets with no hierarchical grouping in the table, no legend, and no cell coloring or icons in the cells. If a dataset is empty, the text " **No data provided** " will be read by the screen reader software. The **Export** > **MS File** and **MS PowerPoint** options will be available and the **Export** > **HTML** options removed. Please note that a special set of keyboard combinations are available for this dataset and are described in the section *Using Keyboard Shortcuts in Page Views* in the reference manual *Getting Started with Alfabet*.

    Please note that editable cells with JAWS® for Windows® software will only be read by the JAWS® for Windows® software if Alfabet is rendered in Google Chrome®. This functionality is not available if Alfabet is rendered in Microsoft Internet Explorer.

  - The auto-complete functionality that is typically available for edit search fields or combo-boxes in editors and filter fields will be automatically disabled.
  - The pop-up calendar that is typically available to select dates will be replaced by a simple data entry field for date fields.
  - The slide-in toolbar, secondary windows, and AlfaBot will not be available in the Alfabet user interface.
  - Placeholder texts displayed in editor fields and filter fields will be automatically disabled.

For detailed information about the configuration of barrier-free user profiles, see the section *Configuring Barrier-Free User Profiles* in the reference manual *Configuring Alfabet with Alfabet Expand*. For more information about how users can navigate the Alfabet user interface in the context of screen reader software and keyboard shortcuts, see the section *Barrier-Free Accessibility in Alfabet* in the reference manual *Getting Started with Alfabet*.

- **Is Administrative User Profile**: Select the checkbox if the user profile is an administrative user profile. An administrative user profile has access to all configured reports and standard views including those specified as administrative. In addition, the user logged in with an administrative user profile has Read/Write access to all objects and document folders that he/she is able to view according to the mandate settings. If the checkmark is not set, users will only have write permission for objects and document folders for which they have access permissions and will not have access to any standard views or configured reports that are specified to be administrative. The **Is Administrative User Profile** attribute will be set to `False` by default for all user profiles except the administrative user profiles. For an overview of all standard administrative views, see the section *Standard Views Accessible via Administrative User Profiles* in the reference manual *Configuring Alfabet with Alfabet Expand - Appendix*.

  A warning message will be displayed in the **User Profile** editor if a user profile is specified as an administrative user profile and the **Type** attribute is set to `ReadOnly`.

  It is recommended that express views are not sent via administrative user profiles. Per default, the XML attribute `EnableExpressViewForAdminProfiles` in the XML object **SolutionOptions** is set to "false" and therefore express views will not be sent via user profiles for which the **Is Administrative User Profile** attribute is set to `True`.

  If your enterprise wants express views to be sent via administrative user profiles, the XML attribute `EnableExpressViewForAdminProfiles` must be set to "true" in the XML object **SolutionOptions** available in the configuration tool Alfabet Expand.

- **Enable Automated Assistant**: Select the checkbox if the automated assistant capability should be enabled for the user profile. If the checkmark is not set, all assistants available for the user profile will be disabled. It is recommended that the **Automated Assistant Enabled** attribute is set to `False` for user profiles requiring barrier-free accessibility as well as administrative user profiles or super users providing help desk services, for example. For more information about how to use the automated assistant capability, see the section *Using the Automated Help Assistant* in the reference manual *Getting Started with Alfabet*. For more information about configuring the automated assistant capability, see the section *Providing Custom Online Help to the User Community* in the reference manual *Configuring Alfabet with Alfabet Expand*.

- **Enable Feedback Bot**: Select the checkbox if the user profile may access the Feedback Bot configured for your enterprise. The Feedback Bot allows users to provide feedback for a view, configured report, object cockpit, guide view, etc. For more information about the configuration of the Feedback Bot, see the section *Configuring the Feedback Bot*.

- **Enable Feedback for View**: Select the checkbox to activate the **Feedback for Current View** capability. This is relevant for user profiles responsible for reviewing and responding to feedback provided via the Feedback Bot.

  Feedback that has been provided for a view or report via the Feedback Bot can be displayed in the Alfabet user interface in a secondary view for those users responsible for reviewing and responding to the feedback. This allows the responsible users to navigate the Alfabet user interface and see the feedback for the relevant view where they currently are. A secondary view with the caption **Feedback for Current View** will be

displayed with a link if feedback has been provided for the view, configured report, object cockpit, guide view, etc. Clicking the link will open the Feedback Review Functionality in a new browser tab which displays all feedback in detail for the view. To implement the **Feedback for Current View** capability, the **Enable Feedback for View** checkbox must be selected in the **User Profile** editor for the relevant user profile and the **Enable Check Feedback for View** checkbox must be selected for the relevant user in the **User Settings** editor.

- **Custom Logo**: A custom logo can be added for a user profile to the header area in the Alfabet user interface.

  The **Custom Logo** field is only available in the `UserProfile_Editor_WithLogo` editor. To make the editor available in the **Users Administration** functionality, the `UserProfile_Editor_WithLogo` must be assigned to the relevant class setting of the `ALFA_USERPROFILE` class. The uploaded logo will be saved to the **Custom Logo** attribute of the GUI scheme assigned to the user profile. If no GUI scheme is assigned to the user profile, the logo will be assigned to the default GUI scheme.

Details tab:

- **Allowed Guide Views**: Specify all guide views in the guide page project that the user profile has access permissions to. You should include any guide view that is specified in the **Guide Page** attribute as well consider the guide views targeted by links in the **Guide Page** attribute. If a user attempts to navigate to a guide view that he/she does not have permission for, an informational message will be displayed.
- **Workflow Activities Explorer**: The configuration of guide views and guide pages may contain a hyperlinked text or button displaying the number of workflow steps that the user is responsible for and that when clicked opens the **Workflow Activities Explorer**. Instead of displaying the default **Workflow Activities Explorer**, a custom explorer can be configured to open instead. Select the custom explorer that should be opened in the context of the guide view/guide page configuration for the selected user profile. For more information about configuring the hyperlinked text or button displaying the number of workflow steps that the user is responsible for in guide views/guide pages, see the section *Adding Workflow, Assignment, Collaboration, and Microsoft Teams Meeting Links to the Guide View* in the reference manual *Designing Guide Pages for Alfabet*. For more information about configuring the **Workflow Activities Explorer** or a custom workflow explorer, see the chapter *Specifying the Customized Workflow Activities Explorer (WFS_Explorer) or a Custom Explorer* in the reference manual *Configuring Alfabet with Alfabet Expand*.

4) Click **OK** to save your data or **Cancel** to exit without saving the data. You will see the new user profile in the table.

Defining a User Profile for Anonymous Users

An anonymous user is someone that typically accesses Alfabet via an express view. All anonymous users will access Alfabet using the user profile that has been defined as the user profile for anonymous users.

The user type `Anonymous` is used only for users that are automatically generated during login with single-sign on mechanisms. If the **Type** attribute is specified as `Anonymous` for a user, the system administrator should consider whether that user should be changed to a user of the type `NamedUser`. Anonymous users have `ReadOnly` access permissions via the user profile specified for anonymous users. For information about how to define an Alfabet user of the type `Anonymous`, see the section *Configuring User Authentication* in the reference manual *System Administration*.

Please consider the following when defining the default user profile for access by anonymous users:

- Anonymous users may only access Alfabet with a user profile of the type `ReadOnly`. If you attempt to define a user profile of the type `ReadWrite` as the default user profile for anonymous users, the definition will be ignored.
- A user profile for anonymous users should NOT include such functionalities as **Document Application**, **Document Component**, etc. which are primarily functionalities to capture and edit data.
- If an anonymous user is sent an express view and that express view has been created for an object that is governed by a mandate, the anonymous user will not be able to access the view in Alfabet. Anonymous users may only access objects that are not governed by a mandate.
- Only one user profile can be selected as the user profile for access by anonymous users. If you define more than one user profile as the user profile for anonymous users, the first user profile in alphabetical order will be used as default user profile for anonymous users. All other definitions will be ignored.
- If no user profile is specified as the default user profile for anonymous users, anonymous users will have access to all `ReadOnly` user profiles.

To define a user profile as the default user profile for anonymous users:

1) Go to the **User Profile Administration** functionality and click the **User Profile Manager** icon  at the top of the explorer.
2) In the **All User Profiles** view, select the user profile that you want to define as the default for anonymous access.
3) In the toolbar, click the **Edit**  button. The **User Profile** editor opens.
4) Ensure that the following is specified:
   - **Type**: Select `ReadOnly`. If you attempt to define a user profile of the type `Read/Write` as the user profile for anonymous users, the definition will be ignored. If you define more than one user profile as the user profile for anonymous users, the first user profile in the alphabetical order of user profiles will be used as the default user profile for anonymous users. All other definitions will be ignored.
   - **Use for Anonymous Users**: Select the checkbox if the user profile should be used as the user profile that is used for anonymous users. You should only select the checkbox if you have defined the value `ReadOnly` for the **Type** attribute. For more information about anonymous access, see the section *Configuring User Authentication* in the reference manual *System Administration*.
5) Click **OK** to save your data.

Assigning Users to a User Profile

The **Users** page view allows you to assign an unlimited number of users to a user profile. Any user logging on with the user profile will have access to the functionalities configured for the user profile as well as the specified access permissions (`ReadOnly` or `Read/Write`) to the objects available to the user profile.

Please note that you should only assign users of the type `Anonymous` to a user profile specified for anonymous users. This would be a user profile for which the **Use for Anonymous Users** attribute is selected. If the user requires a different user profile, you must change the **Type** attribute of the user to `NamedUser`. For more information about specifying a user as a named user or an anonymous user, see the section Creating a User.

1) Go to the **User Profiles Administration** functionality and click the **User Profile Manager** icon at the top of the explorer.
2) In the **All User Profiles** view, select the user profile that you want to assign users to.
3) Click **Users**.
4) In the toolbar, click **New** > **Add Person**. The selector opens.
5) Select the user(s) that you want to add to the user profile and click **OK**. The users are assigned to the selected user profile.

To detach a user from the user profile, select the user in the **Users** view and click the **Detach** button. If you detach a user from a user profile and the user is currently working with Alfabet the user will be immediately logged out and needs to re-login. This is a security measure to ensure that a user immediately loses administrative access after being detached from administrative user profiles.

To edit the user information, select the user and click the **Edit** button.

Making Configured Reports Available to a User Profile

The **Reports** page view displays all reports that have been configured by your report designer in the configuration tool Alfabet Expand. The users with a specified user profile may add and thus execute the configured report in the **Reports** functionality or **Configured Reports** page view. Reports are configured in the tool Alfabet Expand.

Access permissions to a configured report may be limited to only the authorized user and authorized user group by means of the **Restrict to Authorized Access** attribute defined in the **Report** editor accessible in the **Reports Administration** functionality. In order for a user to have access to a configured report, the **Restrict to Authorized Access** attribute should NOT be selected in the **Report** editor. If a checkmark is set for the **Restrict to Authorized Access** field, a user will NOT be able to access the selected configured report unless he/she is the authorized user or member of an authorized user group. For more information, see the section Defining and Managing User Access to Configured Reports. Users with relevant access permissions can access the configured report in the **Reports** functionality in the **Search** module. For more information, see *Executing Your Configured Reports* in the reference manual *Getting Started with Alfabet*.

User access to configured reports can be specified for individual users, user groups, and user profiles. For an overview of the various access permission concepts for configured reports, see the section Understanding User Access to Configured Reports.

User profiles can be marked as administrative and configured reports can be marked as restricted to administrative user profiles. Configured reports restricted to administrative user profiles should only be added to user profiles that are specified as administrative. Configured reports that are not restricted to administrative user profiles can be added to both administrative and non-administrative user profiles.

1) Go to the **User Profiles Administration** functionality and click the **User Profile Manager** icon  at the top of the explorer.
2) In the **All User Profiles** view, select the user profile that you want to specify configured reports for.
3) Click **Reports**. All reports configured by your report designer are displayed. The following information is displayed:
   - **Name**: The caption of the configured report.
   - **Type**: Whether the configured report is based on a native SQL query (`SQL`), an Alfabet query (`Query`), graphic template (`Custom`), object profile/object cockpit (`ObjectView`), or URL (`Extern`).
   - **Object State**: The current object state of the configured report.
   - **User Decides**: A checkmark indicates that an individual user may decide whether or not to add the configured report to the **Reports** view available via the **Search** functionality when accessed via the selected user profile.
   - **Included**: A checkmark indicates that the configured report is automatically included in the **Reports** view available via the **Search** functionality when accessed via the selected user profile.
   - **Excluded**: A checkmark indicates that the configured report is automatically excluded for the **Reports** view available via the **Search** functionality when accessed via the selected user profile. Users with the selected user profile cannot view the configured report.
   - **Restrict to Administrative User Profiles**: A checkmark indicates that the configured report can only be accessed via administrative user profiles.
4) To specify the accessibility of a configured report for the selected user profile, select the configured report and click one of the following options in the **Edit** menu:
   - **Allow Users with User Profile to Add Report**: Select to allow an individual user with the selected user profile to decide if he/she wants to add the configured report from the **Reports** view available via the **Search** functionality. A checkmark is displayed in the **User Decides** column.
   - **Automatically Include Report for All Users with User Profile**: Select to automatically display the configured report for users with the selected user profile in the **Reports** view available via the **Search** functionality. A checkmark is displayed in the **Included** column.
   - **Allow No Users with User Profile to Access Report**: Select if the configured report should not be available to users with the selected user profile. Users with the selected user profile will not be able to add the configured report to the **Reports** view available via the **Search** functionality. If the user previously had access to this configured report with the current user profile, it will be crossed out in the **Reports** view and can no longer be executed with the selected user profile. A checkmark is displayed in the **Excluded** column.

Chapter 3: Defining and Managing Users

The **Users Administration** functionality allows you to define the users who will access Alfabet. Users of the type ype `NamedUser` or `Anonymous` are based on the person stereotype **User**. It is possible to change a user of the type `NamedUser` or `Anonymous` to a user of the type `NoAccess` that will not have access permissions to Alfabet. ReadOnly and ReadWrite access permissions are managed via the user profile that users are assigned to. For more information about the definition of user profiles, see the section Defining and Managing User Profiles.

Please note the following:

- Alfabet can be configured to automatically send onboarding emails with an automatically generated login password and login link to new users created by a user administrator in the **Users Administration** `(ADMIN_UsersOverview)` functionality. To enable this functionality, the XML attribute `SendUserOnboardingMail` in the XML object `SolutionOptions` must be set to `"true"`. The attribute is set to `"false"` per default. The specfication of a valid email address for new users will be mandatory if the XML attribute `SendUserOnboardingMail`is set to `"true"`. For more information, see the section *Configuring Onboarding Emails for New Users* in the reference manual *Configuring Alfabet with Alfabet Expand*.
- You can create users based on the stereotype **Contact** in the **Contact Administration** functionality. These users are created for documentation purposes only and will have no permission to access Alfabet. For example, an enterprise might want to manage roles associated with vendor personnel in order to manage vendor contacts, but would not want the vendor personal to have access to Alfabet. Ideally, new users that should have no access permissions should be created based on the person stereotype **Contact** in the **Contract Administration** functionality. A user of the type `NoAccess` will not be displayed in the **Contact Administration** functionality. For more information about creating a contact, see the section Creating a Contact

The **Users Administration** functionality displays all users defined in Alfabet. To limit the number of users displayed or find a specific user, specify any of the following filters and click **Update**.

- **Search Pattern**: Enter search criteria and select the attributes you want to search in the adjacent field. The search criteria will be valid for the selected attributes only.
- **User Profile**: Select a user profile to limit the results to only users that have the selected user profile assigned to them.
- **Show Internal Users With No Password**. Select the checkbox to limit the results to only users who are managed directly in Alfabet and that have no password assigned.
- **Show Only Users Marked for Deletion**: Select the checkbox to limit the results to only users that have been deleted from the associated external repository but still exist in the Alfabet database.

The following information is available for each user:

- **First Name**: The user's first name.
- **Name**: The user's family name.
- **User Name**: The user name that is used to log in to Alfabet.
- **Technical Name**: The technical name of the user.

- **Stereotype**: Displays **User**.
- **Deletion Requested**: A checkmark is displayed if the user has been deleted from an external repository but still exists in the Alfabet database.
- **Is User Assistant**: A checkmark is displayed if the user is responsible for answering user questions about the Alfabet solution. Users can request assistance about a standard or configured view in Alfabet via a **Request User Assistance** option in the Help menu available in the main toolbar in Alfabet. If triggered, an email with a link to the view will be sent to the user in the enterprise who has been specified as responsible for user assistance.
- **Skype ID**: The user's Skype ID.

  The **Skype ID** and **Skype Domain** attributes will only be displayed in the **User** editor and in **Users Administration** functionality if interoperability with Skype for Business Server® is activated. For more information about using the Skype capability to communicate with your colleagues, see the section *Skyping with Your Colleagues*. For more information about configuring integration with Skype, see the section *Configuring Interoperability with Skype for Business Server®* in the reference manual *Configuring Alfabet with Alfabet Expand*.

- **Skype Domain**: The user's Skype domain.
- **Last Login**: The timestamp displaying the last time the user was logged in to Alfabet.
- **Number of Failed Logins**: If the Alfabet Web Application is configured to restrict the number of failed login attempt, this column will be displayed with information about the number of current consecutive failed login attempts for the user. If the user logs in with the correct password prior to reaching the maximum number of allowed failed login attempts, the number in this column will be reset to zero. If the number of consecutive failed login attempts is equal to the maximum allowed number, the cell displaying the number will be colored red. The user will not be able to re-login until the user administrator resets the value via the **Action** > **Reset Failed Login Count** option.

  Entering a wrong current password on attempt to change a password is also counted as a failed login attempt. If the number of failed login attempts is reached while a user currently logged in to Alfabet changes his/her password, the user will be logged out and will not be able to re-login until the user administrator resets the failed login count.

  For more information about the restriction of failed login attempts, see the reference manual *System Administration*.

- **Number of Regenerated Passwords**: A counter is incremented each time a password is regenerated via email, either by a user administrator in the **Users Administration** functionality or because the user clicks the **I forgot my password** link on the login screen. The counter is reset to zero on login of the user with the last regenerated password. If the counter reaches a configured maximum number, the password regeneration functionalities are deactivated for the user unless the counter is reset via the **Action** > **Reset Regenerated Passwords Counter** option.
- **Alfabet-Managed User**: A checkmark is displayed if the user logs in via standard login with user name and password and login credentials are managed within Alfabet rather than via an external LDAP server. Most of the password-related user administration tasks such as clearing and regenerating passwords are only applied to users managed in Alfabet.
- **Executes Self-Reflective Events**: A checkmark is displayed if the user is configured to execute REST API service calls for self-reflective events. Only one user can be defined as the user who executes self-reflective events. To define a user to execute self-reflective events, click **Action** > **Set as User Allowed to Execute Self-Reflective Events**. This setting is relevant for the automated translation capability as well as for event management. For more information, see the sections *Configuring*

Automated Translation of Object Data and *Configuring Events* in the reference manual *Configuring Alfabet with Alfabet Expand*.

Once a user has been created, you can navigate to the user's object profile/object cockpit and view all objects that a user is defined as the authorized user of as well as those objects that the selected user has been assigned deputy responsibility or a role. In this way, you can manage the responsibility for objects in Alfabet if, for example, users leave or change positions in the enterprise.

The following information is available:

- Creating a User
- Removing Access Permissions to Alfabet from an Existing User
- Defining, Clearing, and Resetting a User's Password
  - Assigning a Password via System-Generated Email Notifications
  - Assigning a Password Without System-Generated Email Notifications
  - Prompting the User to Define a Password at First Login
  - Clearing an Existing Password
  - Deleting Stored Passwords
  - Defining a Password Expiration Date for a Single User
  - Enabling a User To Login after Failed Login Attempts
  - Resetting the Number of Password Regeneration Requests
- Changing the Login Mode Between Single Sign-On or LDAP and Standard Login
- Clearing the Context Settings for a User
- Setting a User as a Self-Reflective User to Execute Events
- Assigning a Mandate to a User
- Aligning the Alfabet User Repository with an External Repository
- Anonymizing User Data
  - Anonymizing Data of Selected Users
  - Excluding Users from Anonymization
- Removing a User from the User Community
- Assigning User Profiles to a Selected User
- Assigning a User Group to a Selected User
- Assigning Reports to a Selected User
- Reassigning a User's Objects to Another Authorized User or Deputy
- Reassigning Deputy Responsibility for a Selected User's Deputy Objects
- Reassigning Responsibility for a User's Role Objects
- Reassigning Responsibility for a User's Dashboards

- Reassigning or Closing a User's Assignments
- Closing a User's Sent Assignments
- Reassigning a User's Workflows to Another Workflow Owner
- Reassigning a User's Workflow Steps to Another Responsible User
- Creating a Contact

Creating a User

To create a new user based on the person stereotype **User**:

1) In the **Users Administration** functionality, click **New** > **Create New User**.
2) The **User** editor opens. Enter information in the fields, as needed:

   Basic Data tab:

   - **First Name**: Enter the user's first name. You may enter up to 128 characters.
   - **Name**: Enter the user's family name. This name is used for the definition of authorized users of objects, contact persons, owners of assignments, etc. You may enter up to 128 characters.
   - **Technical Name**: If necessary, enter a technical name for the user. If the Alfabet Web Application is configured to use the **Technical Name** attribute rather than the **Name** attribute for users, then your enterprise should define a technical name for all users.

     The Alfabet Web Application can be configured to use the user's technical name instead of the user name in the **Audit History** functionality, which tracks the modification of Alfabet objects. If a technical name is not defined for a user, then the user name (**Name**, **First Name**) of that user is used in the **Audit History** functionality. For general information about the audit functionality, see the section *Viewing the Change History of an Object* in the reference manual *Getting Started with Alfabet*. For information about how to configure the auditing functionality, see the section *Specifying History Tracking for an Object Class* in the reference manual *Configuring Alfabet with Alfabet Expand*.

   - **Phone**: Enter a telephone number where this user can be reached. The user can change or update the phone number in the **Personal Info** functionality available in the < **Alfabet User Name** > menu.
   - **Password Expiration Date**: Enter an expiration date for the password. The user's password will expire on this date and the user will need to define a new password to access Alfabet.
   - **Change Password**: If you select this checkbox, the user must change his/her password during the first login. The user can subsequently change his/her password as needed by means of the **Change Password** functionality in the **<UserName>** menu. To define the user password, see the section Defining, Clearing, and Resetting a User's Password.
   - **Type**: The user type specifies the available access permissions. Select `NamedUser`. Users of the type `NamedUser` must be assigned at least one user profile (with either `ReadWrite` or `ReadOnly` access permissions). You can assign `NoAccess` to remove the access permissions from a user.

The user type `Anonymous` is used only for users that are automatically generated during login with single sign on mechanisms. If the **Type** attribute is specified as `Anonymous` for a user, the system administrator should consider whether that user should be changed to a user of the type `NamedUser`. Anonymous users have `ReadOnly` access permissions via the user profile specified for anonymous users. For information about how to define an Alfabet user as a user of the type `NamedUser` or `Anonymous`, see the section *Configuring User Authentication* in the reference manual *System Administration*. For more information about specifying a user profile for anonymous users, see the section Defining a User Profile for Anonymous Users.

You can also create users of the stereotype **Contact** in the **Contact Administration** functionality. These users are created for documentation purposes only and will have no permission to access Alfabet. For example, an enterprise might want to manage roles associated with vendor personnel in order to manage vendor contacts, but would not want the vendor personnel to have access to Alfabet. Ideally, new users that should have no access permissions should be created based on the person stereotype **Contact** in the **Contract Administration** functionality. For more information about creating a contact, see the section Creating a Contact.

- **User Name**: Enter a name that the user must enter when logging in to Alfabet. Because the user name required for login is not case-sensitive, the user name will be saved to the database in uppercase letters.
- **Picture**: Click the arrow to upload a picture to Alfabet. The picture can be in GIF, PNG, or JPG formats and may not be larger than 16 KB. The user can change or update the picture in the **Personal Info** functionality available in the < **User Name** > menu. The picture will be displayed in the context of the **My Collaborations** functionality and in the main toolbar of the Alfabet user interface next to the < **UserName** > menu. For more information about the collaboration capability, see the section *Communicating with Your Colleagues via the Alfabet Internal Collaboration Functionality* in the reference manual *Getting Started with Alfabet*.

  Depending on your solution configuration, user pictures may be configured to be hidden if the XML attribute `EnableUserPersonalInfoPictureControl` in the XML object **SolutionOptions** is set to `False`. In this case, the **Picture** field will be hidden in this editor and the **Personal Info** dialog for all users in the enterprise.

- **General User Permissions**: Define the following user permissions for the user:
  - **Can Execute Batch Jobs**: Select the checkbox if the user may execute batch utilities designed to access the Alfabet database in stand-alone or remote mode. Access to batch utilities should only be allowed for administrators. For more information about batch utilities, see *About Batch Utilities for Alfabet* in the reference manual *System Administration*.
  - **Has Access to Diagram Designer**: Select the checkbox if the user may access the tool Alfabet Diagram Designer. If you select the checkbox, • 30492_UK_1011_multiThe Picture field available in the Personal Info dialog may be hidden for all users in the enterprise. A new XML attribute EnableUserPersonalInfoPictureControl in the XML object SolutionOptions allows the Picture field to be enabled (True) or disabled (False). If disabled, the Picture field will be removed from the editor. The picture is enabled per default.the **Open Diagram** button will be displayed in the diagram page views that the user has access permissions to. If you clear the checkbox, the **Open Diagram** button will be displayed in the diagram page views that the user has access permissions to. For more information about using the tool Alfabet Diagram Designer, see the reference manual *Designing IT Landscape Diagrams in Alfabet*.

- **Exclude from Anonymization**: Select the checkbox if the user may not be anonymized by means of the data anonymization capability.

- **Is User Assistant**: Select the checkbox if the user is responsible for answering user questions about the Alfabet solution. Users can request assistance about a standard or configured view in Alfabet via a **Request User Assistance** option in the Help menu available in the main toolbar in Alfabet. If triggered, an email with a link to the view will be sent to the user in the enterprise who has been specified as responsible for user assistance.

. The **Request User Assistance** option must be enabled in the XML object **SolutionOptions**. For more information, see the section *Enabling the User Assistance Functionality* in the reference manual *Configuring Alfabet with Alfabet Expand*. For a description of the **Request User Assistance** option, see the section *Requesting User Assistance for a View in the User Interface* in the reference manual *Getting Started with Alfabet*.

Mandates tab: Select a checkbox to assign one or more mandates to the selected user. Typically, a user will be assigned only one mandate. The user will be able to view only objects that have been assigned to his/her mandate. The first mandate checked in the **Mandates** tab of the **User** editor is, by default, the mandate automatically used at login. For an overview of the access permission concepts implemented in Alfabet, see the section *Understanding Access Permissions in Alfabet* in the reference manual *Getting Started with Alfabet*. A user can change mandates at anytime during their Alfabet session. For more information about how a user can change the mandate he/she is currently working with, see the section *Changing the Mandate That You Are Logged In With* in the reference manual *Getting Started with Alfabet*. For more information about the configuration of mandates, see *Configuring Access Permissions* in the reference manual *Configuring Alfabet with Alfabet Expand*.

Select the checkbox for the **Mandate Master** attribute to specify the selected user as a mandate master. A user that has the **Mandate Master** attribute checked can view any object regardless of its mandate.

API Permissions tab:

- **Has API V2 Access**: Select the checkbox if the user may send requests to the Alfabet RESTful API V2. These settings are only required for a user that is technically used by a RESTful client for authorization against the RESTful interface of Alfabet Web Application, version 2. The **Has API V2 Access** checkbox should be deselected for security reasons for all other users. For more information about the Alfabet RESTful API and the required settings in this tab to configure access to the Alfabet RESTful API, see the reference manual *Alfabet RESTful API*.

Alfabet Expand Permissions tab:

- **Has Access to Alfabet Expand**: Select the checkbox if the user may access the configuration tool Alfabet Expand. Access to Alfabet Expand should only be allowed for solution designers and administrators. The **Type** field in the **Basic Data** tab must be set to `NamedUser` to allow access to Alfabet Expand. You should specify the individual functionalities that the user can work with in the **Expand Access Options** field. For more information about the configuration capabilities available in Alfabet Expand, see the reference manual *Configuring Alfabet with Alfabet Expand*.

- **Alfabet Expand Access Options**: Select each checkbox to specify which functionalities the user should be allowed to access in the configuration tool Alfabet Expand. Only users of the type `NamedUser` may access Alfabet Expand, execute batch jobs, and access the Alfabet Diagram Designer. Please note the following:

Designation in Alfabet Expand Web	Designation in Alfabet Expand Windows
ADIF Designer	**ADIF** tab
Administrator	**Admin** tab
Business Function Designer	**Functions** tab
Class Designer	**Meta-Model** tab
Data Workbench Designer	**Data Workbench** tab
Diagram Model/Shape Designer	not relevant
Event Designer	**Events** tab
Guide Page Designer	**Guide Page Designer** option in **Managers Menu**
Icon Designer	**Icon** node in **Presentation** tab
Not relevant	**Managers Menu** containing **Help Manager**, **Database Manager**, etc.
Presentation Model Designer	**Presentation** tab
Publication Designer	**Publications** tab
Report Designer	**Reports** tab
Not relevant	**Survey** tab
System Administrator	not relevant
Utilities	**Meta-Model** menu and **Globalization** menu

Designation in Alfabet Expand Web	Designation in Alfabet Expand Windows
Workflow Designer	**Workflows** tab

Collaboration Data tab:

- **Email**: Enter the user's email address. The email address is necessary for notification emails sent in the context of various Alfabet functionalities including, for example, assignments and monitors. The user can change or update the email address in the **Personal Info** functionality available in the < **UserName** > menu.
- **Email Notification Language**: Select the language to be used for the text messages in automatically generated email notifications. The default language is English.

  For more information about the configuration of language versions for automatically generated email notifications, see the section *Configuring Text Templates for Email Notifications* in the reference manual *Configuring Alfabet with Alfabet Expand*.

- **Skype ID**: Enter the user's Skype ID to implement interoperability with Skype. A maximum of 128 characters is allowed. If interoperability with Skype is enabled for the user, a Skype presence status icon will be displayed next to the user's name in the **Attributes** section of object profiles/object cockpits and previews, allowing other users to contact the authorized user of an object should questions arise. Integration with Skype for Business Server® must be configured in order to implement Skype in the context of Alfabet.

  The **Skype ID** and **Skype Domain** attributes will only be displayed in the **User** editor and in **Users Administration** functionality if interoperability with Skype for Business Server® is activated. For more information about using the Skype capability to communicate with your colleagues, see the section *Skyping with Your Colleagues*. For more information about configuring integration with Skype, see the section *Configuring Interoperability with Skype for Business Server®* in the reference manual *API Integration with Third-Party Components*.

- **Skype Domain**: Enter the user's Skype domain to implement interoperability with Skype.
- **MS Teams User Name**: If integration with Microsoft® Teams is supported in your solution, enter the user's user name in MS Teams.
- **MS Teams User ID**: Enter the user's user ID in MS Teams.

3) Click **OK** to save the data or click **Cancel** to exit without saving your changes. The new user is displayed in the table and a checkmark is set in the **Alfabet-Managed User** column. This indicates that the user will log in with a user name and password and that the user password will be managed in Alfabet. Please note that the user has been created with a random encrypted password. You should reset a password for the user using the mechanisms that are described in the section Defining, Clearing, and Resetting a User's Password to generate a password known to the user.

Removing Access Permissions to Alfabet from an Existing User

You can revoke access to Alfabet for users of type `NamedUser` or `Anonymous` by changing the **Type** attribute to `NoAccess`. If a user with no access permissions attempts to log in, a message will be displayed explaining that access is not allowed.

New users that shall have no access permissions should be created based on the person stereotype **Contact** in the **Contract Administration** functionality. For more information about creating a contact, see the section Creating a Contact.

To revoke access permissions for a user:

1) In the **Users Administration** functionality, select the user that you want to revoke access permissions from and click the **Edit** button.
2) In the **Type** field, select `NoAccess`.
3) Click **OK** to save the data or click **Cancel** to exit without saving your changes. The new user is displayed in the table.

Defining, Clearing, and Resetting a User's Password

The following information is only relevant for users that are managed in Alfabet. Users that are managed in Alfabet must login to Alfabet via standard login. Their password is managed directly in Alfabet and not by an external LDAP source or single sign-on mechanism.

For information about the configuration of authentication mechanisms, see the section *Configuring User Authentication* in the reference manual *System Administration*.

If standard login is used for user authentication, users will log in to Alfabet with an Alfabet -specific user name and password. For this purpose, an initial password must be defined for the user by a system administrator. There are three ways to handle the initial login of a user:

- A mechanism can be triggered that generates a password for the user and sends two emails to the user. One email will inform the user about his/her user name and provide a link to a first login screen, and a second email will inform the user about the first login password. Upon the first login, the user will be prompted to change the password.
- An initial password can be generated manually by the user administrator and the Alfabet Web Application can be configured via the **User** editor to enforce a password change for the next login action. The information about the password and the user name can then be sent to the user by mechanisms not involving Alfabet. Upon the first login, the user will be prompted to change the password.
- No initial password is defined for the user, but the user administrator clears the password and configures the user's data to enforce a password change upon the next login. The information about the user name for login can then be sent to the user by mechanisms not involving Alfabet. Upon the first login, the user will be prompted to change the password.

A user can change his/her own password in the **Change Password** functionality in the **<UserName>** menu. The old password must be entered in addition to the new password to change the password.

The **Change Password** option will be disabled in the **<UserName>** menu for users logging in by means of an enterprise login (for example, by means of federated authentication or Windows sign-on).

If the user has forgotten the old password, the user administrator can clear the password and use one of the above-mentioned mechanisms to create a new one. Optionally, the Alfabet Web Application can be configured to display the **I forgot my password** link on the login screen. If the user clicks the link, he/she will receive two emails. One email will inform the user about his/her user name and provide a link to a first login screen, and a second email will inform the user about the first login password. Upon the first login, the user will be prompted to change the password.

For information about how to activate and configure the display of the **I forgot my password** link, see *Adding a Link to Request a New Password If Users Forget Their Passwords* in the reference manual *System Administration*.

In addition to the **Change Password** option in the **<UserName>** menu, a user can be forced to change his/her password after an expiration date has been reached. In this case, the user will be prompted to change his/her password during login. Password expiration can be configured in two ways:

- A system administrator can configure the Alfabet Web Application to enforce password change in specified intervals. For more information, see the section *Configuring Standard Login* in the reference manual *System Administration*.
- A password's expiration date can be specified in the **Password Expired Date** field in the **User** editor.

A system administrator can configure the Alfabet Web Application to enforce password criteria such as the use of special characters or a minimum password length and the number of recently used passwords that are stored in the Alfabet database. The user may not re-use a currently stored password. However, the user administrator can clear the list of recent passwords to allow their re-use.

If password criteria are enforced, it is no longer possible to allow a user to login without a password. On first login, the user will be prompted to set a password. User passwords can no longer be cleared by an administrator.

For information about the configuration of the Alfabet Web Application to enforce password criteria, see the section *Configuring Standard Login* in the reference manual *System Administration*.

The following information is relevant for password administration for the user administrator:

- Assigning a Password via System-Generated Email Notifications
- Assigning a Password Without System-Generated Email Notifications
- Prompting the User to Define a Password at First Login
- Clearing an Existing Password
- Deleting Stored Passwords
- Defining a Password Expiration Date for a Single User
- Enabling a User To Login after Failed Login Attempts
- Resetting the Number of Password Regeneration Requests

Assigning a Password via System-Generated Email Notifications

After the user is created, you can use the **Regenerate Password** functions to trigger the generation of an initial user password and automatically send emails informing the user about the login credentials.

This functionality includes sending emails to the user via the system. You must ensure that system emails are activated for your Alfabet installation. For more information about activating the sending of emails, see the section *Activating the Dispatch of Email Notifications in Alfabet* in the reference manual *System Administration*. For more information about specifying the message in the emails or configuring custom text templates to use in place of the standard text templates, see the section *Specifying Custom Text Templates for Password Generation* in the reference manual *Configuring Alfabet with Alfabet Expand* and the section *Text Templates for Activation of User Passwords* in the reference manual *Configuring Alfabet with Alfabet Expand - Appendix*.

To send a password via email to one or multiple selected users:

1) In the **Users Administration** functionality, select the user(s) for that you want to initialize login.
2) In the toolbar, select **Action** > **Regenerate Password**.

To send a password via email to all users that currently have no password assigned:

1) In the toolbar, select **Action** > **Regenerate Empty Passwords**.

To send a password via email to all users managed in Alfabet:

1) In the toolbar, select **Action** > **Regenerate All Passwords**.

If the user is currently not able to log in because the maximum number of failed login attempts has been reached, passwords cannot be regenerated unless the failed login count is reset. Resetting the failed login count can be done in combination with password regeneration in a single step:

1) In the **Users Administration** functionality, select the user for that you want to re-set the failed login count and at the same time regenerate the password.
2) In the toolbar, select **Action** > **Reset Failed Login Count and Regenerate Password**.

Assigning a Password Without System-Generated Email Notifications

You can define an initial password directly in the **Users Management** functionality and send it to the user via your own email account or any other mechanism outside Alfabet. In this case, the **Change Password** setting for the user should also be activated to force the user to change his/her password during the first login. This activation must be performed after the password has been reset.

1) In the **Users Administration** functionality, select the user for that you want to initialize login for.
2) In the toolbar, select **Action** > **Change Password** to change the selected user's password.
3) In the window that opens, enter the new password in the **New Password** field, and re-enter the new password in the **Confirm New Password** field.

For a new user without password, the field **Current Password** is deactivated, because it is not required. The functionality can also be used to change an existing password. The **Current Password** field is then active, and the current password must be entered prior to entering the new password.

4) Click **OK** to save your changes.

5) In the toolbar, click **Edit**  . An editor opens.

6) In the **Basic Data** tab, select the **Change Password** checkbox.

The checkmark will automatically be removed from the **Change Password** checkbox after the user has changed the password.

7) Click **OK** to save your changes.

8) Inform the user about his/her user name and password and the web address to use to access the Alfabet user interface.

Prompting the User to Define a Password at First Login

You can force a user to reset his/her password:

1) In the **Users Administration** functionality, select the user for that you want to initialize login for.

2) In the toolbar, click **Edit**  . An editor opens.

3) In the **Basic Data** tab, select the **Change Password** checkbox.

The checkmark will automatically be removed from the **Change Password** checkbox after the user has changed the password.

4) Click **OK** to save your changes.

5) Inform the user about his/her user name and the web address to use to access the Alfabet user interface.

Clearing an Existing Password

You can clear a user's password without knowing his/her current password:

This functionality is not available if the Alfabet Web Application is configured to enforce password criteria. For information about enforcing password criteria, see *Enforcing Passwords and Defining Rules for Password Specification* in the reference manual *System Administration*.

To clear the password of one or multiple users managed in Alfabet:

1) In the **Users Administration** functionality, select the user(s) that you want to clear the password for.

2) In the toolbar, select **Action** > **Clear Password** to clear the selected user's password. The current password will be cleared. Confirm the warning that follows by clicking **OK**.

To clear the password of all users managed in Alfabet:

1) In the toolbar, select **Action** > **Clear All Passwords**.

After having cleared the password, a new password should be defined for the user.

Deleting Stored Passwords

If the Alfabet Web Application is configured to store the defined number of passwords that the user has recently used in the Alfabet database, the user will not be allowed to re-use one of the stored passwords. The list of recently used passwords can be deleted from the Alfabet database for a selected user. This will allow the user to use previous passwords that have been used in the past.

1) In the **Users Administration** functionality, select the user for that you want to clear recent passwords.

   You can select several objects in the table at the same time by holding down the CTRL key while selecting.

2) In the toolbar, select **Action** > **Clear Recent Passwords** to delete the list of recently used passwords. Confirm the warning that follows by clicking **OK**.

Defining a Password Expiration Date for a Single User

Typically, a password expiration period is configured for all users by the system administrator, but a one-time password expiration date can also be defined for a single user by a user administrator. The user will be prompted to change his/her password on the first login on or after the defined date.

1) In the **Users Administration** functionality, select the user for that you want to define a password expiration date for.
2) In the toolbar, click the **Edit** button.
3) In the editor that opens, go to the **Basic Data** tab and enter an expiration date for the password in the **Password Expiration Date** field.
4) Click **OK** to save your changes.

Enabling a User To Login after Failed Login Attempts

The Alfabet components can be configured to count the number of consecutive failed login attempts for a user. If the defined number of allowed consecutive failed login attempts is reached, the user will be blocked and will not be able to log in.

Entering a wrong current password on attempt to change a password is also counted as a failed login attempt. If the number of failed login attempts is reached while a user currently logged in to Alfabet changes his/her password, the user will be logged out and will not be able to re-login.

Password regeneration, either by a user administrator or via the **I forgot my password** link on the login screen will also be disabled.

For information about the required configuration to block user login if a defined number of consecutive failed login attempts is reached, see *Tracking Login Actions in the Windows Event Log* in the reference manual *System Administration*.

After a successful login, the counter for consecutive failed login attempts will be reset to zero. If a user is blocked, the user administrator can reset the number of failed login attempts to zero in order to re-enable login for the user:

1) In the **Users Administration** functionality, select the user whose failed login attempts you want to reset.
2) In the toolbar, select one of the following: **Action** > **Reset Failed Login Count**. The counter for consecutive failed logins is reset to zero.
   - **Action** > **Reset Failed Login Count** to reset the login count only.
   - **Action** > **Reset Failed Login Count and Regenerate Password** to reset the login count, reset the number of regenerated passwords, and change the password of the user to a new generated password automatically sent via email to the user.

Resetting the Number of Password Regeneration Requests

The number of password regeneration emails that can be sent to a user without any login being performed by the user is restricted. Password regeneration emails are sent out if one of the following applies:

- A user administrator resets the password of the user in the **Users Administration** functionality.
- The user clicks the **I forgot my password** link in the login screen.

Each time a password is regenerated via email, a counter is incremented. The counter is reset to zero on login of the user with the last regenerated password.

In the **Users Administration** functionality, the **Number of Regenerated Passwords** column displays the count of unused regenerated passwords. Maximum counts that lead to inactivation of the password regeneration functionality are highlighted in red.

If the counter reaches a configured maximum number, the password regeneration functionalities are deactivated for the user unless the counter is reset by a user administrator:

1) In the **Users Administration** functionality, select the user whose unused regenerated password count you want to reset.
2) In the toolbar, click **Action** > **Reset Regenerated Passwords Counter**.

Changing the Login Mode Between Single Sign-On or LDAP and Standard Login

Users that are created via login using single sign-on mechanisms or authentication via an external LDAP server are regarded as users that are externally managed and not users that are managed in Alfabet. The functionalities available for standard login such as clearing and regenerating passwords are not applied to users that are managed externally. If you create a user directly in Alfabet, the user will be managed in Alfabet, and standard login can be configured for the user.

The information indicating whether a user is or is not managed in Alfabet is displayed in the **Alfabet-Managed User** column in the **Users Administration** functionality. A checkmark will be displayed (`= True`) in the **Alfabet-Managed User** column if the user was created by a user administrator in the **Users Administration** functionality. A checkmark will not be displayed (`= False`) in the **Alfabet-Managed User** column if the user was created via single sign-on mechanisms.

You can convert a user created via a single sign-on mechanism or via external LDAP login to a user managed in Alfabet and vice versa:

1) In the **Users Administration** functionality, select the user or multiple users that you want to convert to internal users.
2) In the toolbar, click **Action** > **Set as Alfabet-Managed User** or **Set As non-Alfabet-Managed User**.
3) Confirm the warning by clicking **Yes**.

Alternatively, the change can be performed in the user editor via the checkbox **Is Alfabet-Managed User**.

Clearing the Context Settings for a User

The settings specified for filters, selectors, columns, etc. that users define in Alfabet are saved in the user's context settings. When the user returns to the view or selector in the current or future session, the most recent settings will be automatically displayed.

These context settings can be cleared for a selected user so that the predefined values are removed.

1) In the **Users Administration** functionality, select the user whose user context settings you want to clear.
2) In the toolbar, select **Action** > **Clear User Context Settings**. The user context settings are removed.

You can select several objects in the table at the same time by holding down the CTRL key while selecting.

Setting a User as a Self-Reflective User to Execute Events

The **Set as User Allowed to Execute Self-Reflective Events** option is relevant for the automated translation capability as well as for event management. A checkmark is displayed in the **Executes Self-Reflective Events** column if the user is configured to execute REST API service calls for self-reflective events. Only one user can be defined as the user who executes self-reflective events. This setting is relevant for the automated translation capability as well as for event management. For more information, see the sections *Configuring Automated Translation of Object Data* and *Configuring Events* in the reference manual *Configuring Alfabet with Alfabet Expand*.

To define a user to execute self-reflective events, click **Action** > **Set as User Allowed to Execute Self-Reflective Events**.

Assigning a Mandate to a User

Some enterprises have a federated architecture. If this is the case in your enterprise, then mandates may be implemented to control the visibility of objects in Alfabet.

A mandate is a means to organize and structure the federated architecture of a holding company. The assignment of mandates to objects allows the holding company to structure the objects in the enterprise architecture in order to regulate visibility to objects across some or all subsidiaries. Only users explicitly assigned to a mandate will see objects with that mandate definition. An object that has not been assigned to a mandate is considered not to be owned by a mandate and is thus visible throughout the holding company to users with relevant access permissions.

The use of mandates in the Alfabet solution is optional. If mandates are implemented in an enterprise, the visibility of an object with a mandate assignment will take precedence over other concepts of access permissions in Alfabet. For example, an authorized user of an object must be assigned the relevant mandate to access the object that he/she is the owner of.

Within the context of mandates, the conventional rules governing access permissions apply. Thus, a user assigned to a mandate will only have Read/Write access permission to the objects made visible by the mandate if he/she has authorized access to the object as an authorized user, deputy, member of an authorized user group or discussion group or via rule-based access permissions, workflow contributor or assignee of an assignment.

Typically, a user will be assigned only one mandate. The user will be able to access any objects that have been assigned to his/her mandate(s).

A user that is assigned multiple mandates may log in to Alfabet using only one mandate at a time. The first mandate checked in the **Mandates** tab of the **User** editor is, by default, the mandate automatically used at login.

A user can change the mandate setting at anytime during the current session by clicking **<UserName>** > **Change User Mandate** in the upper-right corner of the interface an clicking the relevant mandate. The start page will be displayed, and the new mandate applied to the interface.

You can specify a user to be a mandate master. The mandate master has access to all mandates in your enterprise and therefore can log into Alfabet and view any object regardless of its mandate.

For an overview of the access permission concepts implemented in Alfabet, see the section *Understanding Access Permissions in Alfabet* in the reference manual *Getting Started with Alfabet*. A user can change mandates at anytime during their Alfabet session. For more information about how a user can change the mandate he/she is currently working with, see the section *Changing the Mandate That You Are Logged In With* in the reference manual *Getting Started with Alfabet*. For more information about the configuration of mandates, see *Configuring Access Permissions* in the reference manual *Configuring Alfabet with Alfabet Expand*.

To assign a mandate to a user:

1) In the **Users Administration** functionality, select the user that you want to assign a mandate to.
2) In the toolbar, click the **Edit**  button. The **User** editor opens.
3) Go to the **Mandates** tab and enter the information into each field, as required. Select a checkbox to assign one or more mandates to the selected user. Typically, a user will be assigned only one mandate. The user will be able to access any objects that have been assigned to his/her mandate.
4) Select the **Mandate Master** checkbox to assign all mandates in your enterprise to the selected user. A mandate master can log into Alfabet and view any object regardless of its mandate.
5) Click **OK** to save your changes or **Cancel** if you do not want to save them.

Aligning the Alfabet User Repository with an External Repository

It may be that the Alfabet users are sourced from an external repository such as LDAP. Because information is subject to change in the external repository, the user information in Alfabet will typically be resynchronized and updated regularly per batch job by a system administrator. For more information about resynchronization of data with an external source, see the section *Integrating Data from External Sources* in the reference manual *System Administration*.

If a user is deleted from an external repository and is also defined in the Alfabet database, the user must be manually removed from Alfabet. Before you delete the user, you should ensure that all objects that the user is responsible for have been reassigned to another user.

A **Person** selector (`PersonSelector`) is available that includes a **Deletion Requested** field which allows users to be searched for based on the `DeletionRequested` property. To implement the **Person** selector (`PersonSelector`) with the **Deletion Requested** field, the following line must be added to the XML object **GeneralViewMap** available in the configuration tool Alfabet Expand:

```
<MapEntry Type="Selector" Source="StandardSelector:PERS_SelectorDef"
Target="PersonSelector:PERS_SelectorDef_PersonDeletionRequested"/>
```

For more information about configuring the XML object **GeneralViewMap**, see the section *Configuring the Alfabet User Interface to Map Standard Configuration Objects to Custom Configuration Objects*.

If you delete a user from the Alfabet database, you will irrevocably delete the user and user information from the Alfabet database. You will also delete the relationship of that user to objects, assignments, user groups, and configured reports. Therefore, you should review and, if necessary, reassign responsibilities for objects for which the user is defined as an authorized user, deputy, or role as well as reassign any assignments, and workflow activities that the user is responsible for. Responsibility for such objects can be reassigned in the respective page views available in the object profile of the user who is to be deleted. Please note that in the case of workflows, user information (such as who has performed a workflow step) is maintained even for users who have been deleted from the Alfabet database.

1) In the **Users Administration** functionality, select **Deletion Requested** in the filter field next to the **Search Pattern** field and enter `True` in the **Search Pattern** field in order to find the users deleted from an external repository and click **Update**. All users that have been deleted from the external repository but remain in the Alfabet database will be displayed.

2) Select a user and click the **Navigate** button to navigate to the object profile in order to review and, if necessary, reassign responsibilities for objects that the user is defined as an authorized user, deputy, or role as well as reassign any assignments, and workflow activities that the user is responsible for.

3) Return to the **Users Administration** functionality by clicking the browser Back button.

4) In the **Users Administration** functionality, select the user you want to delete and click the **Delete** button.

Anonymizing User Data

The data anonymization capability ensures data transparency and accountability across the enterprise as well as supports the enterprise to meet requirements of the General Data Protection Regulation (GDPR)s. By means of pseudonymization, for example, user data can be replaced with an artificial identifier to ensure anonymity if the user leaves the enterprise, or all sensitive data can be replaced with artificial identifiers in the development or test environment.

Anonymization can be performed for data of the type `String`, `Text`, `URL` and `Picture`. A solution designer must enable the anonymization function for the object class `Person` in Alfabet Expand to activate the anonymization options available for user administrators. The solution designer also configures individual anonymization methods for the object class properties that may be anonymized. User data can either be substituted with a random string, substituted with the `REFSTR` of the user, or set to `NULL` during anonymization.

For details about the configuration required to anonymize data and the consequences of anonymization, see *Anonymizing Data* in the reference manual *Configuring Alfabet with Alfabet Expand*.

The following information is available:

- Anonymizing Data of Selected Users
- Excluding Users from Anonymization

Anonymizing Data of Selected Users

To trigger anonymization of the data of selected users:

Anonymizing data is a sensible process that might disrupt database integrity. It cannot be reverted! **Always back up the Alfabet database prior to triggering data anonymization!**

If the anonymized user is the **Last Update User** or **Creator** of any configuration object subordinate to the **Classes** explorer node in the **Meta-Model** tab, the connections of all currently running Alfabet components with the Alfabet database will be terminated and the database will be locked during the anonymization process. The Alfabet components need to be restarted afterwards.

1) In the **Users Administration** functionality, select the user(s) that you want to anonymization.
2) In the toolbar, click **Action** > **Anonymize User**.
3) Confirm the warning by clicking **Yes**.

Excluding Users from Anonymization

Single users can be excluded from anonymization to make sure that the data of that user cannot be anonymized:

1) In the **Users Administration** functionality, select the user that you want to exclude from anonymization.
2) In the toolbar, click **Edit**  . The **User** editor opens.
3) Select the checkbox **Exclude From Anonymization**.
4) Click **OK** to save your changes.

Removing a User from the User Community

You can remove any user from the user community. To do so, you must delete the user from the Alfabet database.

If you delete a user from the Alfabet database, you will irrevocably delete the user and user information from the Alfabet database. You will also delete the relationship of that user to objects, assignments, user groups, and configured reports. Therefore, you should review and, if necessary, reassign responsibilities for objects for which the user is defined as an authorized user, deputy, or role as well as reassign any assignments, and workflow activities that the user is responsible for. Responsibility for such objects can be reassigned in the respective page views available in the object profile of the user who is to be deleted. Please note that in the case of workflows, user information (such as who has performed a workflow step) is maintained even for users who have been deleted from the Alfabet database.

To delete a user from the Alfabet database:

1) In the **Users Administration** functionality, select the user you want to delete.
2) In the toolbar, click the **Delete**  button.
3) Confirm the warning by clicking **Yes,** or click **No** to exit without deleting the selected object(s).

Assigning User Profiles to a Selected User

The **Assigned User Profiles** page view displays all user profiles assigned to the selected user.

User profiles are the basis of user administration in Alfabet and serve as the entry point when accessing Alfabet. Every user must log in with a user profile that has been assigned to him/her by a user administrator. Therefore, all users accessing Alfabet must be assigned at least one user profile. However, users may possess multiple user profiles in accordance with their responsibilities in the user community and in the enterprise as a whole. A user can switch to another permissible user profile at any point during a user session.

A user profile specifies the Alfabet functionalities available to a user, the visibility and editability of object classes and object class attributes, as well as the availability of associated capabilities including, for example, wizards and workflows.

A user that has been defined as an anonymous user (**Type** = `Anonymous`) may only be assigned to a user profile specified for anonymous users. If a different user profile needs to be assigned to an anonymous user, you must first change the user to a named user (**Type** = `NamedUser`) in the **Person** editor. For more information about specifying a user as a named user or anonymous user, see the section Defining and Managing Users in the reference manual *User and Solution Administration*.

To assign a user profile to the selected user:

1) In the **Users Administration** functionality, select the user that you want to assign a user profile to and click the **Navigate**  button.
2) Click **Assigned User Profiles**.
3) In the toolbar, click **New** > **Assign User Profiles**. The object selector opens and displays all user profiles created for your enterprise.
4) Select a user profile and click **OK**. The user profile is displayed in the table.

To detach a user profile from the selected user, click the user profile in the **Assigned User Profiles** page view and click the **Detach** button.

Assigning a User Group to a Selected User

The **User Group Memberships** page view allows you to assign a user group to the selected user.

A user group bundles a set of users. Once a user group is assigned to an object, it is referred to as the object's authorized user group.

The rules for propagation or inheritance of user group access permissions are configured by your solution designer in the configuration tool Alfabet Expand. For more information, see *Configuring Access Permissions* in the reference manual *Configuring Alfabet with Alfabet Expand*.

1) In the **Users Administration** functionality, select the user that you want to assign a user group to and click the **Navigate** button.
2) Click **User Group Memberships**.
3) In the toolbar, click **New** > **Assign User Groups**. The object selector opens and displays all user groups.
4) Select the user group that you want to assign the user to and click **OK**. The user group is displayed in the table.

To detach a user group from the selected user, click the user group in the **User Group Memberships** page view and click the **Detach** button.

Assigning Reports to a Selected User

The **User's Reports** page view displays all configured reports that have been assigned to the selected user. The configured reports defined here will be displayed to a user accessing the **Reports** view available via the **Search** functionality.

Access permissions to a configured report may be limited to only the authorized user and authorized user group by means of the **Restrict to Authorized Access** attribute defined in the **Report** editor accessible in the **Reports Administration** functionality. For a user to have access to a configured report, the **Restrict to Authorized Access** attribute should NOT be selected in the **Report** editor. If a checkmark is set for the **Restrict to Authorized Access** field, a user will NOT be able to access the selected configured report unless he/she is the authorized user or member of an authorized user group. For more information, see the section Defining and Managing User Access to Configured Reports. Users with relevant access permissions can access the configured report in the **Reports** functionality in the **Search** module. For more information, see *Executing Your Configured Reports* in the reference manual *Getting Started with Alfabet*.

User access to configured reports can also be specified for user profiles and user groups. For more information, see the sections Making Configured Reports Available to a User Profile and Defining the User Profiles That Can Access a Configured Report.

1) In the **Users Administration** functionality, select the user that you want to assign a configured report to and click the **Navigate**  button.
2) Click **User's Reports**. All configured reports that have been assigned to the selected user are displayed. The following information is displayed for each configured report:
   - **Caption**: The caption of the configured report.
   - **Description**: A description describing the purpose of the configured report.
   - **Type**: Displays whether the configured report is based on a native SQL query (`SQL`), an Alfabet query (`Query`), graphic template (`Custom`), object profile/object cockpit (`ObjectView`), or URL (`Extern`).
3) To make a configured report available to the selected user, click **New** > **Add Report**.
4) In the selector, select one or more configured reports to add to the user and click **OK**.
5) To view a configured report, select the configured report in the **User's Reports** view and click the **Open Report** button.

Reassigning a User's Objects to Another Authorized User or Deputy

In the **Authorized User Objects** page view you can view the objects that a selected user has authorized access to.

Please note that changes made in the **Authorized User Objects** page view are NOT updated in the `LAST_UPDATE_USER` and `LASTUPDATE` properties. The `LAST_UPDATE_USER` and `LASTUPDATE` properties are only updated in the Alfabet database when changes are made in an object's editor. For more information about the history tracking capability, see the section *Viewing the Change History of an Object*.

1) In the **Users Administration** functionality, select the user whose object responsibilities you want to reassign and click the **Navigate**  button.
2) Click **Authorized User Objects** to open the view. The view displays all objects that you have authorized access to.
3) To limit the set of objects in the view, select the object class whose objects you want to define in the **Select Class** field and click **Update**.
4) You can assign a new authorized user, authorized user group, or deputy to an individual object that you select in the table:
   - To add a deputy to an object, select the object in the table and click **Edit** > **Set Deputy for Selected Object(s)**. Although multiple deputies can be assigned to an object, you can only define one deputy at a time.
   - To change the deputy for an object, select the object in the table and click **Edit** > **Substitute Deputy for Selected Object(s)**.
   - To remove all deputies from an object, select the object in the table and click **Edit** > **Remove Deputies for Selected Object(s)**.

- To assign a new authorized user to an object, select the object in the table and click **Edit** > **Substitute Authorized Access for Selected Object(s)**.
- To change the authorized user or authorized user group for an object, select the object in the table and click **Edit** > **Change Authorized Access for Selected Object**.
- To assign an authorized user group to an object, select the object in the table and click **Edit** > **Add User Groups to Selected Object(s)**.
- To make any of the changes above to all objects displayed in the view, select the respective option in the **Edit** menu targeting all objects.

Reassigning Deputy Responsibility for a Selected User's Deputy Objects

A deputy is an Alfabet user granted Read/Write access permissions to an object in order to act on behalf of the authorized user.

In the **Deputy Objects** page view, you can view all objects that a selected user has deputy responsibility for. Additionally, you can remove deputy responsibility from the user and assign another user as the new deputy of the respective object. The original deputy will no longer have authorized access to the object and the object will no longer be available in his/her **Deputy Objects** page view in the **Personal Info** functionality.

1) In the **Users Administration** functionality, select the user whose deputy responsibilities you want to reassign and click the **Navigate**  button.
2) Click **Deputy Objects**.
3) In the **Select Class** field, select the class of objects in the drop-down list that you want to display in the table and click **Update**.
4) The following options are available:
   - To change the deputy for an object, select the object in the table and click **Edit** > **Substitute Deputy for Selected Object(s)**.
   - To remove all deputies from an object, select the object in the table and click **Edit** > **Remove Deputies for Selected Object(s)**.
   - To make any of the changes above to all objects displayed in the view, select the respective option in the **Edit** menu targeting all objects.

Reassigning Responsibility for a User's Role Objects

In Alfabet, some objects have roles assigned to them.

A role defines the functional relationship or responsibility that a user or organization has to an object. A role is based upon a configured role type that is configured for an object class. Roles are defined for informational purposes only and provide detail about users or organizations that may have information about or a stake in the object. The definition of a role for an object does not impact access permissions.

In the **Role Objects** page view, you can view all objects for which a selected user has a role assigned. Additionally, you can remove a role object from a selected user and reassign it to another user. A user can be assigned a functional role for an object in the *Responsibilities Page View* for the relevant object.

1) In the **Users Administration** functionality, select the user whose roles you want to reassign and click the **Navigate**  button.
2) Click **Role Objects**.
3) In the **Select Class** field, select the class of objects in the drop-down list that you want to display in the table and click **Update**.
4) The following options are available:
   - To change the role for an object, select the row in the table and click **Edit** > **Substitute Responsibility for Selected Role(s)**.
   - To remove all roles from an object, select the row in the table and click **Edit** > **Remove Selected Responsibilities**.
   - To make any of the changes above to all objects displayed in the view, select the respective option in the **Edit** menu targeting all objects.

Reassigning Responsibility for a User's Dashboards

In Alfabet, some users have dashboards assigned to them. The **User's Dashboards** page view allows you to manage the ownership of dashboards in case the dashboard owner has left the company or changed roles. This view allows you to assign another user as the owner of a selected dashboard, thereby removing the current user as the dashboard owner.

1) In the **Users Administration** functionality, select the user whose dashboards you want to manage and click the **Navigate**  button.
2) Click **User's Dashboards**.
3) The following options are available:
   - To change the ownership of a dashboard, select the row in the table and click **Edit** > **Substitute Responsibility for Selected Dashboard(s)**.
   - To remove the ownership of a dashboard from the selected user, select the row in the table and click **Edit** > **Remove Selected Dashboards**.
   - To make any of the changes above to all dashboards displayed in the view, select the respective option in the **Edit** menu targeting all objects.

Reassigning or Closing a User's Assignments

The **User's Assignments** page view displays all assignments that are assigned to the selected user. Here you can reassign individual assignments or reassign all assignments, if necessary. Depending on the configuration of your Alfabet solution, an email notification may be sent to the assignee informing them that they have been assigned a task for the selected object.

1) In the **Users Administration** functionality, select the user whose assignments you want to reassign or close and click the **Navigate**  button.

2) Click **User's Assignments**.

3) The following options are available:

- To reassign a selected assignment to another user, click the assignment(s) in the table and click **Edit** > **Reassign Selected Assignment(s) to Another User**. Select the user to whom you want to assign the assignment to and click **OK**. The assignment is removed from the **User's Assignments** page view and will appear in the new responsible user's **My Assignments** functionality.
- To reassign a selected assignment to the originator of the assignment, click the assignment(s) in the table and click **Edit** > **Reassign Selected Assignment(s) to Originator**. The assignment is removed from the **User's Assignments** page view and will appear in the originator's **My Assignments** functionality.
- To close one or more assignments, click the assignment(s) in the table and click **Edit** > **Close Selected Assignment(s)**. After you have confirmed the warning by clicking **Yes**, the status of the selected assignments is set to **Closed** and the assignments are removed from the **User's Sent Assignments** page view as well as the **Sent Assignments** functionality.
- To make any of the changes above to all objects displayed in the view, select the respective option in the **Edit** menu targeting all objects.

Closing a User's Sent Assignments

The **User's Sent Assignments** page view displays all assignments that have been sent by a user that do not have the status **Closed**. Here you can close individual assignments or close all assignments, if necessary.

1) In the **Users Administration** functionality, select the user whose assignments you want to close and click the **Navigate**  button.

2) Click **User's Sent Assignments**.

3) The following options are available:

- To close one or more assignments, click the assignment(s) in the table and click **Edit** > **Close Selected Assignment(s)**.
- To close all assignments, click **Edit** > **Close All Assignment(s)**.

4) Confirm the warning by clicking **Yes**. The status of the selected assignments is set to **Closed** and the assignments are removed from the **User's Sent Assignments** page view as well as the **Sent Assignments** functionality.

Reassigning a User's Workflows to Another Workflow Owner

The **User's Workflows** page view displays all workflows that the selected user is the workflow owner of. You can reassign the ownership of a workflow to another user. The following information is displayed:

- **Workflow**: Displays the name of the workflow. The workflow's name is automatically the same as the workflow template that the workflow is based on.
- **Workflow ID**: Displays the identification number of the workflow.

- **Current Workflow Step**: Displays the name of the workflow step that is currently being performed in the workflow. You must expand the workflow to see the workflow step.
- **Workflow Step ID**: Displays the identification number of the current workflow step.
- **Object ID**: Displays the identification number of the object that is being processed in the current workflow step.
- **Object Name**: Displays the name of the object that is being processed in the workflow step.
- **Start Date**: Displays the timestamp showing when the workflow/workflow step was started.
- **End Date**: Displays the timestamp showing when the workflow/workflow step was completed.
- **Workflow State**: Displays the current state of the workflow/workflow step. For more information about the meaning of the various values and how to resolve errors, see the section *Understanding Event Values and Resolving Workflow Errors*.

1) In the **Users Administration** functionality, select the user whose workflows you want to reassign and click the **Navigate**  button.
2) Click **User's Workflows**.
3) Select the workflow that you want to assign to another user and click the **Change Owner**  button.
4) In the selector, select the person you want to reassign ownership of the workflow to and click **OK**. The workflow is reassigned and will be displayed in the selected user's **My Workflows** functionality.

Reassigning a User's Workflow Steps to Another Responsible User

The **User's Workflow Activities** page view displays all workflow steps that the selected user is responsible for. You can reassign individual workflow steps to another user. The following columns are displayed in the view:

- **Workflow**: Displays the name of the workflow that the workflow step belongs to.
- **Workflow ID**: Displays the identification number of the workflow that is being processed.
- **Object Name**: Displays the name of the object that is being processed in the workflow step.
- **Object ID**: Displays the identification number of the object that is being processed in the current workflow step.
- **Active Step**: Displays the name of the workflow step that is currently being performed in the workflow.
- **Step ID**: Displays the identification number of the workflow step that is being processed.
- **Previous Step**: Displays the name of the workflow step that was performed prior to the current workflow step.
- **Enter Date**: Displays the timestamp showing when the workflow advanced to the current workflow step.
- **Remaining Days**: If a deadline has been configured, displays the number of days left before the workflow step expires.
- **Ready**: Displays a checkmark if the current workflow step has been performed and is ready to be confirmed and completed.

- **Locked By**. Displays the name of the user who is currently working on the workflow step.
- **Current Step Comments**: Displays a checkmark if a comment has been made for the current workflow step. To access the comment, select the workflow step, click the **Navigate** button and click **Active Step Activities** to view the comments in the **Message** column.
- **Previous Step Comments**: Displays a checkmark if a comment has been made for the previous workflow step. To access the comment, select the workflow step, click the **Navigate** button and click **Activities of Preceding Steps** to view the comments in the **Message** column.

1) In the **Users Administration** functionality, select the user whose workflow steps you want to reassign and click the **Navigate** button.
2) Click **User's Workflow Activities**.
3) Select the workflow step that you want to delegate and click the **Delegate** button.
4) The **Delegate Workflow Step** editor opens. Define the following fields, as needed:
   - **Substitute All Current Workflow Step Owners**: Select if all users currently responsible for the selected workflow step should be replaced by the user(s) that the workflow step is being delegated to.
   - **Revoke Responsibility from Current User**: Select this checkbox if you want to remove the responsibility for the workflow step from the user that is currently responsible. Clear this checkbox if you want to include the user who is currently responsible for the workflow step to remain responsible.
   - **Comment**: Provide an explanation about why the workflow is being delegated to another user. The explanation will be displayed in the **Message** column of the **Workflow Step Event Trace** page view.
   - **Delegates**: Click the **Add Person** button to select one or more persons to delegate the workflow step to.
5) Click **OK** to save your changes. The responsible user will find the workflow step in the **Workflow Activities Explorer** or **My Workflow Activities** view.

Creating a Contact

The **Contact Administration** functionality allows you to create contacts in Alfabet. Contacts are users of the stereotype **Contact** that are created for documentation purposes only and will have no permission to access Alfabet. For example, an enterprise might want to manage roles associated with vendor personnel in order to manage vendor contacts, but would not want the vendor personnel to have access to Alfabet. Ideally, new users that should have no access permissions should be created based on the person stereotype **Contact** in the **Contract Administration** functionality.

The table displays all existing contacts (users based on the stereotype **Contact**). To limit the contacts displayed in the view, enter search criteria in the **Search Pattern** field and click the **Update** button. The following columns are displayed:

- **First Name**: Displays the contact's first name.

- **Name**: Displays the contact's family name.
- **User Name**: Displays the email address of the contact.
- **Stereotype**: Displays **Contact**.

To create a new contact:

1) In the **Contacts Administration** functionality, click **New** > **Create New Contact**.
2) In the **Contact** editor, define the following fields:
   - **First Name**: Enter the contact's first name. You may enter up to 128 characters.
   - **Name**: Enter the contact's family name. You may enter up to 128 characters.
   - **Email**: Enter the contact's email address. The email address is mandatory for a user based on the stereotype **Contact**. The email address must be unique and may not be reused for another user.

     Once the **Contact** editor is closed, the email address will be displayed in the **User Name** field in the **Contract Administration** functionality and will overwrite the `USER_NAME` property in the Alfabet database. If a new user is created and the same email is entered in the **Email** field for the new user, a message will be displayed saying that the user with the name <email address> already exists.

   - **Phone**: Enter a telephone number where this Contact can be reached.
   - **Exclude from Anonymization**: Select the checkbox if the user may not be anonymized by means of the data anonymization capability.
   - **Email Notification Language**: Select the language to be used for the text messages in automatically generated email notifications. The default language is English.
   - **Picture**: Click the arrow to upload a picture to Alfabet. The picture can be in GIF, PNG, or JPG formats and may not be larger than 16 KB.
3) Click **OK** to save the data or click **Cancel** to exit without saving your changes.

Chapter 4: Defining and Managing User Groups

An authorized user group is a user group that has been assigned to an object. Like the object's authorized user, all users in the authorized user group have Read/Write access permissions to the object. A user group may have an unlimited number of subordinate user groups.

A solution designer may configure the Inheritance and/or propagation of access permissions for authorized user groups. If the inheritance of access permissions has been specified, then all user groups in the user group hierarchy that are descendant to a user group should have the same access permissions to that object. If the propagation of access permissions has been specified, then all user groups in the user group hierarchy that are ascendant to a selected user group should have the same access permissions to that object.

The rules for propagation or inheritance of user group access permissions are configured by your solution designer in the configuration tool Alfabet Expand. For more information, see *Configuring Access Permissions* in the reference manual *Configuring Alfabet with Alfabet Expand*.

For more detailed information about access permissions in Alfabet, see the section *Understanding Access Permissions in Alfabet* in the reference manual *Getting Started with Alfabet*.

The **User Group Administration** functionality allows you to define the user groups that constitute your Alfabet community. Here you can create new user groups, structure the user group hierarchy, assign users to user groups as well as define which Alfabet query-based reports or external reports should be accessible to a user group.

The **User Groups** explorer displays all existing user groups  . Any user group, whether a root or descendant group, may contain an unlimited number of users.

To create a root-level user group:

1) In the **User Group Administration** functionality, click the **User Groups** node  at the top of the explorer.
2) In the toolbar, click **New** > **Create User Group**. You will see the **User Group** editor.
3) Enter the information in the following fields, as needed:

   Basic Data tab:

   - **ID**: Alfabet assigns a unique identification number to each object in the inventory. This number cannot be edited.
   - **Name**: Enter a unique name for the user group.
   - **Description**: If necessary, enter a meaningful description that will clarify the purpose of the user group.
4) Click **OK** to save the user group or **Cancel** if you do not want to save it.

If you delete a user group, it will be irrevocably deleted from the Alfabet database. If you delete a user group that has subordinate groups, all sub-groups with their subordinate objects will move to the top-level in the explorer tree and thus remain in the database. If you delete a user group that has users assigned to it, those persons will remain in the database.

The following information is available for user groups:

- Assigning Users to a User Group
- Assigning Subordinate User Groups to a User Groups
- Assigning Configured Reports to a Selected User Group

Assigning Users to a User Group

The **Users** page view allows you to add existing users to the selected user group.

Changes made to user group membership (such as adding, reassigning, or removing users) may not be immediately applied to user permissions in Alfabet. This allows the change to be undone if a user is erroneously assigned to a user group. Changes to user group membership will be updated to Alfabet in regular cycles.

Any person that you want to assign to a user group must first be defined in the **Users Administration** functionality. For more information, see Defining and Managing Users in the reference manual *User and Solution Administration*.

To assign users to a selected user group:

1) In the **User Groups Administration** functionality, double-click the user group in the explorer tree that you want to assign users to.
2) Click **Users**.
3) In the toolbar, click **New** > **Add Person** in order to add an existing person to the user group.
4) The object selector opens. Enter search criteria, as needed, and click **Search**.
5) Click **OK** to assign the person to the user group or **Cancel** to exit without saving your changes.

Assigning Subordinate User Groups to a User Groups

The **Sub-Groups** page view allows you to define sub-groups for the selected user group. A user group can have an unlimited number of sub-groups.

Depending on the specific customer configuration, inheritance and/or propagation may apply. In the case of inheritance, if a user group that has authorized access to an object also has sub-groups, then all members of the sub-group also have Read/Write access rights to the object. In the case of propagation, if a user group that has authorized access to an object also has superordinate user groups, then all members of the superordinate groups also have Read/Write access rights to the object.

A user group may have an unlimited number of subordinate user groups.

The rules for propagation or inheritance of user group access permissions are configured by your solution designer in the configuration tool Alfabet Expand. For more information, see *Configuring Access Permissions* in the reference manual *Configuring Alfabet with Alfabet Expand*.

For more detailed information about access permissions in Alfabet, see the section *Understanding Access Permissions in Alfabet* in the reference manual *Getting Started with Alfabet*.

To define sub-groups for a user group:

1) In the **User Groups Administration** functionality, double-click the user group in the explorer tree that you want to define sub-groups for.
2) In the user group's profile, expand the **User Group Structure** workspace and click **Sub-Groups**
3) In the toolbar, click either.
   - **New** > **Create User Group** to define a new sub-group for the user group. A window opens in which you can enter the necessary information.
   - **New** > **Move Existing User Group Here** in order to add an existing user group as a sub-group to the selected user group. The object selector opens. Enter search parameters, as needed, and click **Search**.
4) Click **OK** to assign the sub-group to the selected user group or **Cancel** to exit without saving your changes.

Assigning Configured Reports to a Selected User Group

The **Reports** page view allows you to add one or more configured reports to a selected user group. The users in the user groups may add and thus execute the configured report in the **Reports** functionality or **Configured Reports** page view. Reports are configured in the tool Alfabet Expand.

The attribute **Restrict to Authorized Access** should NOT be selected in order for the user to have access to the configured report. If the attribute **Restrict to Authorized Access** is set to active, the user will NOT be able to access the selected configured report unless he/she is the authorized user or member of an authorized user group. Authorized access and the **Restrict to Authorized Access** attribute are defined in the **Report** editor, which is accessible in the **Reports Administration** functionality. For more information, see Defining and Managing User Access to Configured Reports.

User access to configured reports can be specified for individual users, user groups, and user profiles. For an overview of the various access permission concepts for configured reports, see the section Understanding User Access to Configured Reports.

To define user group access for a selected user group.

1) In the **Users Groups Admin** functionality, select the user group that you want to assign a configured report to and click the **Navigate**  button.
2) Click **Reports**.
3) Select the report that you want to define access for and click one of the following in the toolbar:
   - To allow an individual user in the selected user group to add the configured report to the **Configured Reports** functionality or **Configured Reports** page view, select **Edit** > **Allow Users to Add Report**. A checkmark is displayed in the **User Decides** column. The user may add the configured report to his/her list of configured reports.
   - To automatically add the configured report for all users in the selected user group to the **Configured Reports** functionality or **Configured Reports** page view, select **Edit** > **Automatically Include Report for All Users in User Group**. A checkmark is displayed in the **Included** column. The user will automatically see the configured report in his/her list of configured reports.

- To exclude this configured report from all users in the selected user group so that they cannot add the configured report to the **Configured Reports** functionality or **Configured Reports** page view, select **Edit** > **Allow No Users in User Group to Access Report**. A checkmark is displayed in the **Excluded** column. The user cannot access the configured report. If the user previously had access to this configured report, it will be displayed in red and cannot be executed.

Chapter 5: Defining and Managing User Access to Configured Reports

The **Reports Administration** functionality allows you to define the authorized access to the configured reports that have been created, configured, and structured in report folders by your report designer in the configuration tool Alfabet Expand.

The creation of configured reports is a very complex process. Although basic report generation, configuration, and structuring functionalities are available in the **Reports Administration** functionality, these features are for demonstration purposes only and should not be used to create configured reports for the production environment. For more information about creating configured reports, see the chapter *Configuring Reports* in the reference manual *Configuring Alfabet with Alfabet Expand*.

There are many different types of configured reports in Alfabet. For an overview of how to work with the different kinds of visualizations available in configured reports, see the section *Analyzing Your Data in Page Views and Reports* in the reference manual *Getting Started with Alfabet*. The following explains the values that may be displayed in the **Type** column of the view.

Report Type	Description
`Query`	Configured report based on an Alfabet query. The output is a tabular report. Tabular reports display search results in a table. Depending on the configuration, search parameters can either be defined by the user or exclusively defined by the solution designer who configured the report.
`NativeSql`	Configured report based on a native SQL query. The output is a tabular report.
`Custom`	Configured report displaying graphic representation of data based on Alfabet queries or native SQL queries. The Alfabet queries defining the content of the configured report are specified in a report assistant. Graphic reports include cluster maps, trees and tree maps, layered diagrams, data-capture matrices and charts.
`ObjectView`	Configured report displaying an object profile or object cockpit configured for objects in a selected object class.
`Extern`	Configured report opening a URL. This type of report displays data in the Alfabet database or external data generated by third-party reporting tools. The report could display either a static website or dynamic content generated by means of a script or reporting tool that searches the Alfabet database via an SQL-based query. Optionally, an external report may be configured to provide ReadOnly access to views via hyperlinks. Depending on the configuration of the link syntax, access may be via a named user or an anonymous user, and the view may open with a different user profile than the one that was used for log in. For more information, see the section *Links to Alfabet Views from External Applications* in the reference manual *System Administration*

When you open the view, you will see a table listing all configured reports that have been configured by your report designer in the configuration tool Alfabet Expand. The configured reports are structured in sections in the

table that correspond to the report folders created in Alfabet Expand. The name, structure, and number of sections will depend on your individual solution configuration. You can click a column header to sort the configured reports in alphanumeric order based on the value of the columns. When configured reports are structured in report folders, the configured reports are sorted within the folders and the folder structure is maintained.

If necessary, you can limit the objects in the table. To do so, enter search criteria in the **Search Pattern** field and click **Update**. The search pattern is used to match the name, caption or description of the configured reports or report folders. If a configured report or report folder matches the search criteria, the complete folder hierarchy of the configured report or report folder will be displayed. Objects not matching the search criteria but that are in the same hierarchy are also displayed. If a report folder matches the search criteria, the configured reports and report folders subordinate to the report folder will also be displayed. To return to the view that displays all reports structured in report folders, you must clear the **Search Pattern** field and click **Update**.

Each column in the table is explained below:

- **Caption**: The caption of the configured report/report folder. The caption is displayed to the user in the **Configured Reports** functionality and in the **Configured Reports** page view.
- **Description**: The description of the configured report/report folder provided by the creator of the configured report/report folder. The description is displayed to the user in the **Configured Reports** functionality and in the **Configured Reports** page view.
- **Type**: Whether the configured report is based on a native SQL query (`SQL`), an Alfabet query (`Query`), graphic template (`Custom`), object profile/object cockpit (`ObjectView`), or URL (`Extern`).
- **Name**: The name of the configured report/report folder. The name is used to identify the report or report folder. Two reports may have the same caption but must differ in their technical names.
- **Object State**: The current object state of the configured report. Only configured reports with the state `Active` are available to users in the **Configured Report** views.
- **Apply to Class**: For reports that are applied to a class, the name of the object class is displayed in this column. Reports of the type `Extern` or `Query` that are applied to a class are available via the **Configured Reports** page view of the object profile of the specified class.
- **Base Classes**: The base class specified in the **Base Classes** attribute for the report in the configuration tool Alfabet Expand. This attribute defines all classes that are required to run the report.
- **Restrict to Administrative User Profiles**: A checkmark is displayed in this column if the configured report can only be accessed via administrative user profiles.

  In the editor for editing the configured report in the **Reports Administration** functionality, the **Restrict to Administrative User Profiles** option can only be set for configured reports that have a report view defined. Report views can only be added to a configured report in Alfabet Expand.

The following information is available:

- Understanding User Access to Configured Reports
- Restricting Access to the Configured Report's Authorized User/Authorized User Groups
- Defining the User Profiles That Can Access a Configured Report
- Defining the User Groups That Can Access a Configured Report
- Managing the Visibility of the Report for Individual Users

- Resetting the User Context Settings for a Tabular Configured Report

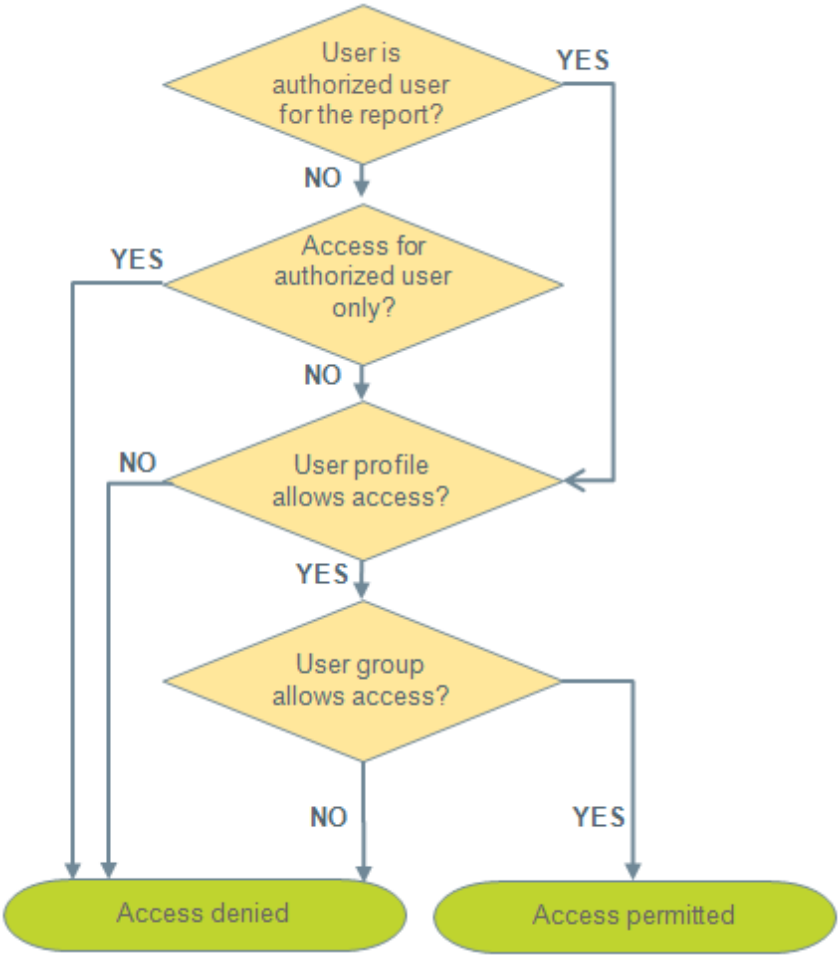
Understanding User Access to Configured Reports

The **Reports Administration** functionality allows you to administrate user access permissions to configured reports. You may explicitly assign user profiles, user groups, and users to access each configured report. The following access permissions can be defined:

- Access permission for users with a specified user profile.
- Access permission for users in a specified user group.
- Access permission for an authorized user and authorized user groups only.

The access permission concepts described in this section only define access to configured reports in the **Configured Reports** functionality or the **Configured Reports** page view available in relevant object profiles. A configured report can be accessed by a user independent of the access permissions configured in the **Reports Administration** functionality if the report is accessible to the user via the configured Alfabet interface that the user has access to. A configured report could be available via a customized button in the toolbar of an object profile, as a view in an object profile, or embedded in a wizard or workflow.

The figure below provides an overview of the rules governing access to reports:



Whether a user may access a configured report depends on the combination of all access permissions configured for a report. The following rules govern access:

- A user can access a configured report when the user profile he/she is currently logged in with allows access to the configured report AND when at least one of the user groups that the user is a member of permits access to a configured report.
- If the user profile that the user is currently logged in with allows access to the configured report, and the user is a member of multiple user groups, the user can access the configured report if one of the user groups permits access to the configured report even if another user group explicitly excludes access to a configured report.
- If user groups are specified but the user is not member of any of the user groups, the user can access the report if the user profile allows access.

  Users not assigned to a user group can be excluded from access without changing the access rules that apply to users that are members of user groups. For more information, see Restricting Access to the Configured Report's Authorized User/Authorized User Groups.

- If no user groups are specified, access permission will depend only on the access permissions for the user profile.
- If access is restricted to an authorized user and/or authorized user groups, this authorized user and/or members of authorized user groups can only access the configured report if the user profile he/she is currently logged in with AND at least one of the user groups he/she is member of permits access to the configured report.
- If access is restricted to an authorized user and/or authorized user groups, the access permissions for user profiles and user groups are only evaluated for the authorized user and/or members of the authorized user groups. All other users will not be able to access the configured report.
- Configured reports can be defined to be accessible with administrative user profiles only (user profiles for which the option **Is Administrative User Profile** is enabled). The user can only access these configured reports if both access permissions for the user profile and a user group is granted and the current user profile is an administrative user profile.
- If a user has access permission to a configured report, the report results displayed will also depend on the access permissions for objects in Alfabet:
  - The objects found in a configured report may be controlled by mandates. As a result, the report will only display those objects that correspond to the user's current mandate or that have no mandate assignment.
  - The objects found in a configured report may be controlled by conventional Alfabet access permissions. The user may navigate to and edit any objects to which he/she has Read/Write access. The user may navigate to and see a ReadOnly view for any objects to which he/she has ReadOnly access.

The user can access all reports for which he /she has access permissions in the **Configured Reports** functionality. Configured reports that are assigned to an object class can also be accessed in the **Configured Reports** page view in the object's profile. Whether a configured report can be added automatically or whether the user can explicitly decide to include the configured report in the **Configured Reports** functionality or **Configured Reports** page view is configured in the **Reports Administration** functionality as part of the definition of access permissions for a specified user, user groups, and user profiles.

If multiple user profiles are assigned to a user, the reports available in the **Configured Reports** functionality or **Configured Reports** page view may differ depending on the following:

- The report will be automatically added to the **Configured Reports** functionality or **Configured Reports** page view if either the user profile or one of the user groups that the user is a member of is defined to automatically add the configured report. The **Open Report** button is only activated if the automatically added configured report is accessible if both user profile and user group memberships grant access to the configured report. The user cannot detach the configured report from the **Configured Reports** functionality or **Configured Reports** page view.
- If a report is automatically added to the view of user profile A and must be explicitly added for user profile B, the report will be displayed automatically in the **Configured Reports** functionality or **Configured Reports** page view of user profile A. The configured report will only be displayed in the views for user profile B when the user explicitly selects the report for display.
- If a user explicitly adds a configured report to the **Configured Reports** functionality or **Configured Reports** page view when logged in with one user profile, the report will also be visible for all user profiles defined to allow explicit selection of reports.
- If the user detaches a report from the **Configured Reports** functionality or **Configured Reports** page view when logged in with one user profile, the report will also be detached for all user profiles that are defined to allow explicit selection of reports. However, the report will still be visible in the **Configured Reports** functionality or **Configured Reports** page view for those user profiles for which the report is automatically added to the table.
- If the access permissions change so that access permissions are no longer granted to a configured report that is already added to the **Configured Reports** functionality or **Configured Reports** page view, the report will still be displayed in the view but the **Open Report** button is deactivated. The configured report can be detached from the **Configured Reports** functionality or **Configured Reports** page view.

The availability of the configured report in the **Configured Reports** functionality or **Configured Reports** page view also depends on the settings of attributes of the configured report:

- If the **Report State** attribute is changed to `Plan` or `Retired` for a configured report, the report cannot be explicitly added by the user to the **Configured Reports** functionality or **Configured Reports** page view. If it is already added either automatically or explicitly by the user, it will still be displayed but the **Open Report** button will be deactivated.
- If the **Selector Behavior** attribute of the configured report is set to `NotVisible`, the configured report cannot be explicitly added by the user to the **Configured Reports** functionality or **Configured Reports** page view. If it is added automatically or already added explicitly by the user, it will be displayed, and the user will have access to the configured report if all required access permissions apply.

Restricting Access to the Configured Report's Authorized User/Authorized User Groups

You can restrict access to the configured report so that only the users defined as the configured report's authorized user and authorized user groups may access the configured report.

If access is restricted to authorized users or an authorized user group, at least one authorized user or authorized user group must be defined for the configured report. Otherwise, the restriction to authorized access has no effect.

1) In the **Reports Administration** functionality, select the report that you want to define user access to.

2) In the toolbar, click the **Edit**  button. The **Report** editor opens.

3) To restrict access to the configured report so that only the users defined as the configured report's authorized user and authorized user groups can access the report, select the checkbox next to the **Restrict to Authorized Access** field.

4) In the **Authorized Access** tab, define the relevant authorized user and authorized user groups of the configured report.

   If access is restricted to authorized users or an authorized user group, at least one authorized user or authorized user group must be defined for the configured report. Otherwise, the restriction to authorized access is ignored.

5) Click **OK** to save your changes or click **Cancel** to exit without saving your changes.

By default, users that are not assigned of any of the existing user groups can access all reports. If you want to restrict access to a report to users that are assigned at least to one user group with access permission to the report, you can use the restriction to authorized access to configure this behaviour:

1) Select the checkbox of the **Restrict to Authorized Access** field of the report.
2) In the **Authorized Access** tab, select all available user groups.

As a result, all members of all user groups are authorized users, while the users that are not assigned to the user group are not authorized and access is not granted.

Defining the User Profiles That Can Access a Configured Report

In the **User Profiles** page view, access permissions for access to configured reports can be defined for all available user profiles. The users in a specified user profile may add and thus execute the configured report in the **Configured Reports** functionality or **Configured Reports** page view. Reports are configured in the tool Alfabet Expand.

User access to configured reports can be specified for user groups and user profiles. A user can access a configured report if both the user profile he/she is currently logged in with and at least one of the user groups he/she is assigned to grant access to the configured report. For an overview of the various access permission concepts for configured reports, see the section Understanding User Access to Configured Reports.

Access permissions to a configured report may be limited to only the authorized user and authorized user group by means of the **Restrict to Authorized Access** attribute defined in the **Report** editor accessible in the **Reports Administration** functionality. For a user to have access to a configured report, the **Restrict to Authorized Access** attribute should NOT be selected in the **Report** editor. If a checkmark is set for the **Restrict to Authorized Access** field, a user will NOT be able to access the selected configured report unless he/she is the authorized user or member of an authorized user group. For more information, see the section Defining and Managing User Access to Configured Reports. Users with relevant access permissions can access the configured report in the **Reports** functionality in the **Search** module. For more information, see *Executing Your Configured Reports* in the reference manual *Getting Started with Alfabet*.

The **User Profiles** page view displays all user profiles that have been defined for your enterprise.

Configured reports can be defined to be accessible with administrative user profiles only (user profiles for which the option **Is Administrative User Profile** is enabled). The user can only access these configured reports if the current user profile is an administrative user profile. You must not assign a user profile that is not administrative to a configured report that is restricted to administrative user profiles.

1) In the **Reports Administration** functionality, select the report that you want to define user access for and click the **Navigate**  button.
2) Click **User Profiles**. All user profiles configured for your enterprise are displayed. The following information is displayed:
   - **User Profile**: The name of the user profile.
   - **Administrative**: A checkmark indicates that the user profile is an administrative user profile. Reports with the attribute Restrict to Administrative User Profiles set to True should only be included for administrative user profile.
   - **User Decides**: A checkmark indicates that an individual user may decide whether or not to add the configured report to the **Configured Reports** view available via the **Search** functionality when accessed via the selected user profile.
   - **Included**: A checkmark indicates that the configured report is automatically included in the **Configured Reports** view available via the **Search** functionality when accessed via the selected user profile. The configured report cannot be removed from the table by the user.
   - **Excluded**: A checkmark indicates that the configured report is automatically excluded for the **Configured Reports** view available via the **Search** functionality when accessed via the selected user profile. Users with the selected user profile cannot view the configured report.
3) To specify the accessibility of a configured report for the selected user profile, select the configured report and click one of the following options in the **Edit** menu:
   - **Allow Users with User Profile to Add Report**: Select to allow an individual user with the selected user profile to add the configured report in the **Configured Reports** view available via the **Search** functionality. A checkmark is displayed in the **User Decides** column.
   - **Automatically Include Report for All Users with User Profile**: Select to automatically display the configured report for users with the selected user profile in the **Configured Reports** view available via the **Search** functionality. A checkmark is displayed in the **Included** column.
   - **Allow No Users with User Profile to Access Report**: Select if the configured report should not be available to users with the selected user profile. Users with the selected user profile will not be able to add the configured report to the **Reports** view available via the **Search** functionality. If the user previously had access to this configured report with the current user profile, it will be crossed out in the **Configured Reports** view and can no longer be executed with the selected user profile. A checkmark is displayed in the **Included** column.

Defining the User Groups That Can Access a Configured Report

The **User Groups** page view displays all user groups that have been defined for your enterprise. The users in a specified user group may add and thus execute the configured report in the **Configured Reports** functionality or **Configured Reports** page view. Reports are configured in the tool Alfabet Expand.

If user groups are specified but the user is not member of any of the user groups, the user can access the report if the user profile allows access.

Access permissions to a configured report may be limited to only the authorized user and authorized user group by means of the **Restrict to Authorized Access** attribute defined in the **Report** editor accessible in the **Reports Administration** functionality. In order for a user to have access to a configured report, the **Restrict to Authorized Access** attribute should NOT be selected in the **Report**

editor. If a checkmark is set for the **Restrict to Authorized Access** field, a user will NOT be able to access the selected configured report unless he/she is the authorized user or member of an authorized user group. For more information, see the section Defining and Managing User Access to Configured Reports. Users with relevant access permissions can access the configured report in the **Reports** functionality in the **Search** module. For more information, see *Executing Your Configured Reports* in the reference manual *Getting Started with Alfabet*.

User access to configured reports can be specified for individual users, user groups, and user profiles. For an overview of the various access permission concepts for configured reports, see the section Understanding User Access to Configured Reports.

1) In the **Reports Administration** functionality, select the report that you want to define user access for and click the **Navigate**  button.
2) Click **User Groups**. All user groups configured for your enterprise are displayed. The following information is displayed:
   - **User Group**: The name of the user group.
   - **User Decides**: A checkmark indicates that an individual user may decide whether or not to add the configured report to the **Reports** view available via the **Search** functionality when accessed via the selected user profile.
   - **Included**: A checkmark indicates that the configured report is automatically included in the **Configured Reports** view available via the **Search** functionality when accessed via the selected user profile. The configured report cannot be removed from the table by the user.
   - **Excluded**: A checkmark indicates that the configured report is automatically excluded for the **Configured Reports** view available via the **Search** functionality when accessed via the selected user profile. Users with the selected user profile cannot view the configured report.
3) To specify the accessibility of a configured report for the selected user group, select the user group and click one of the following options in the **Edit** menu:
   - To allow an individual user in the selected user group to add the configured report to the **Configured Reports** functionality or **Configured Reports** page view, select **Edit** > **Allow Users to Add Report**. A checkmark is displayed in the **User Decides** column. The user may add the configured report to his/her list of configured reports.
   - To automatically add the configured report for all users in the selected user group to the **Configured Reports** functionality or **Configured Reports** page view, select **Edit** > **Automatically Include Report for All Users in User Group**. A checkmark is displayed in the **Included** column. The user will automatically see the configured report in his/her list of configured reports.
   - To exclude this configured report from all users in the selected user group so that they cannot add the configured report to the **Configured Reports** functionality or **Configured Reports** page view, select **Edit** > **Allow No Users in User Group to Access Report**. A checkmark is displayed in the **Excluded** column. The user cannot access the configured report. If the user previously had access to this configured report, it will be displayed in red and cannot be executed.

Managing the Visibility of the Report for Individual Users

The users with the relevant access permissions for a configured report can execute the configured report in the **Configured Reports** functionality or **Configured Reports** page view. The configured report is either configured to be displayed to the user by default or the user must actively choose to view the report. In any case, the user can actively remove the report from the table in the **Configured Reports** functionality or **Configured Reports** page view.

The **Users** view of a configured report gives an overview of the users that currently view the configured report in their table in the **Configured Reports** functionality or **Configured Reports** page view. It also allows to forcedly remove or add a report to the table of individual users. This functionality may be useful for example in the following situations:

- A configured report shall be retired. Prior to deleting the configured report, you can explicitly remove it from the reports table of all users currently viewing the report.
- A user with access permissions to a report added to her/his **Configured Reports** view. If you change the access permissions and the user does not have access to the report any longer, the report will remain in the table. However, the **Open Report** button will be deactivated if the user selects the report. You can actively remove the report from the user's **Configured Reports** view by removing the user in the **Users** view of the report.
- A report is configured to be optional, and you want to know whether the report is used by any users (for example, to decide whether it is still required). The **Users** view displays the user's who currently have added the report to his/her table.

In the **Users** view, the following information is displayed about the users that currently view the configured report:

- **User Name**: The user's user name.
- **First Name**: The user's first name.
- **Name**: The user's family name.

To add a configured report to the **Configured Reports** functionality or **Configured Reports** page view of a user that currently does not view the report:

1) In the **Reports Administration** functionality, select the report and click the **Navigate**  button.
2) Click **Users**. All users that currently view the report in the **Configured Reports** functionality or **Configured Reports** page view are displayed.
3) In the toolbar, click **New** > **Associate Persons**.
4) In the object selector, search for the user(s) that should be able to view the report in the **Configured Reports** functionality or **Configured Reports** page view.
5) Select the user and click **OK**.

To remove a configured report from the **Configured Reports** functionality or a user's **Configured Reports** page view that includes the report:

1) In the **Reports Administration** functionality, select the report and click the **Navigate**  button.
2) Click **Users**. All users that currently view the report in the **Configured Reports** functionality or **Configured Reports** page view are displayed.

3) In the table, select the user(s) that should no longer view the report.
4) In the toolbar, click the **Remove** button.
5) Click **Yes** to confirm the warning.

Resetting the User Context Settings for a Tabular Configured Report

A tabular configured report can be configured to display a **Configure** button in the toolbar of the configured report. The user can change the order and number of visible columns in the report via the button. The resulting layout is stored in the user context setting of the user and will be used to render the report with the user in the future.

If a solution designer changes the number of columns in the configured report after users have already configured the report, the users will not see the new content. It is not part of the design stored in the user context settings.

The design saved for a selected configured report can be removed selectively from all user context settings via button interaction to make changes visible to all users. User then have to define a new design if they still want to hide or re-order columns.

To remove the information about configuration for a configured report from all user context settings:

1) In the **Report Administration** functionality, select the report in the table.
2) In the toolbar, click **Action** > **Clear User Context Settings**.

Chapter 6: Configuring Monitors to Track Objects in Alfabet

Alfabet provides a number of different monitors that help your enterprise maintain accurate and up-to-date data. Monitoring allows you to keep track of specified objects for activity (the object has been edited), inactivity (the object has not been edited), or an approaching date (such as an end date) for a specific object or all objects in an object class, and consistency of data. When a monitor is activated, all users defined as listeners to the monitor are automatically sent an email notification. The users defined will be able to access the object that the monitor is about via a hyperlink in the email notification by means of a specified user profile.

Alfabet provides a variety of monitors to alert users about changes that have occurred to specific objects which may require further activity, reviews that did not occur as intended, or transactions that were expected to occur. The following monitor types are available in Alfabet.

Monitor Type	Purpose	Example	Details
Activity Monitor	An activity monitor alerts specified users about changes that have been made to objects in an object class. The monitor owner must specify a set of attributes that are to be monitored and the user(s) that are to be alerted if the monitor is triggered. An email will be sent to the monitor owner and all defined users when the monitor is triggered.	Users responsible for applications should be alerted about changes made to the object state of the information flows associated with the applications. In this case, you would define an activity monitor of the type Application with the monitored context Application Information Flow. You would then define the property Object State as the property to track for this context.	Created in the **Monitors** functionality in the **Home** tab or in the **Solution Admin** tab available via an administrative user profile. For more information about defining activity monitors, see the section *Defining Activity Monitors for Specific Objects* in the reference manual *Getting Started with Alfabet*.
Inactivity Monitor	An inactivity monitor alerts specified users about the absence of activity occurring to objects in an object class. The monitor owner must specify a set of attributes that are to be monitored and a set of users that are to be alerted if the monitor is triggered. An email will be sent to the monitor owner and all defined users if a defined attribute for any object in the class is not changed or reviewed within a specified period of time. Users responsible for the object (authorized user, members of authorized user groups, etc.) that review an object but make no modifications can specify the monitor object as reviewed via the **Mark as Reviewed** button in the object profile of the targeted object.	Users responsible for specific applications should be reviewed on a monthly basis. In this case, you would define an inactivity monitor of the type Application with a monthly frequency. The monitored context Application Information Flow.	Created in the **Monitors** functionality in the **Home** tab or in the **Solution Admin** tab available via an administrative user profile. For more information about defining inactivity monitors, see the section *Defining Inactivity Monitors for Specific Objects* in the reference manual *Getting Started with Alfabet*.

Monitor Type	Purpose	Example	Details
Date Monitor	A date monitor allows you to keep track of the approach of a date for specified objects and their related objects (such as an object's start date, end date, target date, etc.). An email will be sent to the monitor owner and all defined users when the monitor is triggered.	Users responsible for specific applications should be alerted about the approaching end date of the business supports provided by the applications.	Created in the **Monitors** functionality in the **Home** tab or in the **Solution Admin** tab available via an administrative user profile. For more information about defining inactivity monitors, see the section *Defining a Date Monitor for Specific Objects* in the reference manual *Getting Started with Alfabet*.
System Date Monitor	A system date monitor is a time-triggered monitor for an object class on a system-wide basis. A system date monitor allows you to keep track of the approach of a date for all objects in a specified object class and their related objects (such as an object's start date, end date, target date, etc.). When a specified date approaches for an object in the object class, an assignment will be generated for the object's authorized user. The authorized user will also receive an email information them of their new assignment.	All users responsible for any application should be alerted about business supports that have an end date later than the applications providing the business support.	Created in the **System Date Monitors** functionality in the **Solution Admin** tab available via an administrative user profile.
Consistency Monitor	A consistency monitor supports the system-wide maintenance of objects in the Alfabet database. The consistency monitor is specified to periodically search for inconsistencies among objects. Each consistency monitor is based on an Alfabet query or native SQL query that defines the object classes targeted by the query as well as the inconsistent attributes to be detected. If an inconsistency is found by the query, an assignment will be generated for the object's authorized user. The authorized user will also receive an email information them of their new assignment. The timely completion of the assignment triggered by a consistency monitor can be tracked by the solution administrator.	Users responsible for applications should be alerted about business supports that have an end date later than the applications providing the business support.	Created in the **Consistency Monitors** functionality in the **Solution Admin** tab available via an administrative user profile.

Monitor Type	Purpose	Example	Details
Notification Monitor	A notification monitor allows email notifications to be automatically triggered based on configured Alfabet queries or native SQL queries. The queries specify the objects and attributes that are targeted as well as the users who shall be notified about the objects found by the queries.	Users who have a specified role for applications should be alerted about any new information flows that are created for the applications that they are associated with.	Created in the **Notification Monitors** functionality in the **Solution Admin** tab available via an administrative user profile.

Activity, inactivity, and date monitors are created in the **Monitors** functionality, which is typically available to the general user community and is therefore described in the chapter *Keeping Track of Objects via Monitors* in the reference manual *Getting Started with Alfabet*. System date monitors, consistency monitors, and notification monitors may only be configured by user logging in with the Admin user profile and are therefore described below.

The following must be configured in order to work with monitors in Alfabet:

- In order for any monitor to be executed and email notifications to be sent to relevant users, a batch process must be configured by your system administrator. For more information about setting up a batch process, see the section *Batch Processing for Monitors and Change Management with AlfaBatchExecutor.exe* in the reference manual *System Administration*.
- All Alfabet functionalities for which the email capability is to be implemented require the setup of a connection to an SMTP server for outgoing email in the tool Alfabet Administrator. For more information, see the section *Activating the Dispatch of Email Notifications in Alfabet* in the reference manual *System Administration*.
- Text templates for email notifications may be customized for all monitors in the configuration tool Alfabet Expand. For more information, see *Configuring Monitors* in the reference manual *Configuring Alfabet with Alfabet Expand*.
- For each monitor created, the monitor owner must define the **User Profile to Access Object in Notification** field in the monitor's editor in the relevant monitor functionality in the **Solution Admin** tab that may be accessed via an administrative user profile. This is described for each monitor type in the chapter Configuring Monitors to Track Objects in Alfabet in the reference manual *User and Solution Administration*.
- Access permissions must be available for the user profile so that relevant users receiving the email notification can access the relevant objects that are targeted by the hyperlink in the email notification. For more information about the configuration of access permissions, see the section *Configuring Access Permissions for Alfabet* in the reference manual *Configuring Alfabet with Alfabet Expand*.
- All object classes that are to be monitored via activity monitors and inactivity monitors have history tracking enabled in the configuration tool Alfabet Expand and therefore, the **Audit** attribute must be set to `True` for these object classes. Please note that this is not necessary for date, system date, consistency, and notification monitors. These monitors will be implemented even if the **Audit** attribute is set to `False` for the relevant object class. For more information, see the section *Specifying History Tracking for an Object Class* in the

chapter *Configuring the Class Model* in the reference manual *Configuring Alfabet with Alfabet Expand*.

For more information about configuring monitors, see:

Defining System Date Monitors

The **System Date Monitors** functionality in the **Solution Administration** module allows you to define time-triggered monitors for an object class on a system-wide basis. When you create a system-wide date monitor, you assign the monitor to a specific object class as well as define the context to be monitored, the property to be monitored (either Start Date, End Date, or Target Date), and the number of days before or after the defined date that you want the notification sent.

When a specified date approaches for an object in the object class, an assignment will be generated for the object's authorized user. The authorized user will also receive an email informing them of their new assignment.

In addition to the general requirements for monitors described in the section Configuring Monitors to Track Objects in Alfabet , the following must be configured to implement system date monitors:

- The assignments generated in the context of system-wide date monitors must be configured by your solution designer in the configuration tool Alfabet Expand. For more information, see the section *Configuring Assignments for System-Wide Date Monitors* in the reference manual *Configuring Alfabet with Alfabet Expand*.
- The following text templates should be configured in the configuration tool Alfabet Expand for the email notifications sent when a system date monitor is triggered. For more information, see the section *Configuring Text Templates for Email Notifications* in the reference manual *Configuring Alfabet with Alfabet Expand*.

- The text template `AssignmentNew` in the `ASMT` folder is used for email notifications about the assignments generated for the monitor in the configuration tool Alfabet Expand.
- The text template `MonitorObjectCountdownReview` in the `MON` folder is used to fill the **Description** property of the assignments that are created.

Please note that assignments are created for system-wide date monitors. Therefore, if the message logging is activated for assignments, then the emails sent in the context of these monitors will be logged. For more information about the **Email Message Log** functionality, see the section Tracking the Email Messages Sent in the Context of Alfabet Functionalities in the reference manual *System Administration*.

To configure a system-wide date monitor for a specific object class:

1) In the **Solution Admin** menu, click **System Date Monitors**.
2) In the toolbar, click **New** > **Create New Monitor**. The object selector opens.
3) Select the object class that should be monitored and click **OK**. The **System Date Monitor** editor opens.
4) Enter information in the fields as needed. Each field is defined below:
   - **Monitor Type**: Displays the object class that the monitor is assigned to track.
   - **Monitored Context**: Select a context to be monitored. The monitored contexts available are preconfigured. For more information about monitored contexts, see the section *Overview of Monitored Contexts* in the reference manual Alfabet Expand.
   - **Monitored Attribute**: Choose the attribute that should be monitored. Your choices will be a start, end, or target date, depending on the object class.
   - **Countdown Period**: Enter the number of days prior to the date of the property selected when you want the date monitor to be triggered. If you want the monitor to be triggered after the date of the selected property, enter a - (minus) sign before the number.
   - **User Profile to Access Object in Notification**: Select the user profile that will be used by email recipients to access the object in Alfabet via the hyperlink in the monitor's notification email.
5) Click **OK** to save the date monitor of **Cancel** to exit without saving.

If you delete a system date monitor, it will be irrevocably deleted from the Alfabet database.

Defining Consistency Monitors

The **Consistency Monitors** functionality in the **Solution Administration** module supports the system-wide maintenance of objects in the Alfabet database. The functionality allows a solution administrator to define monitors that periodically search for inconsistencies among objects. For example, a consistency monitor could be created to alert users responsible for applications about business supports that have an end date later than the applications providing the business support.

Each consistency monitor is based on an Alfabet query or a native SQL query that defines the object classes targeted by the query as well as parameters defining the inconsistent attributes to be detected. If an inconsistency

is found by the query, an assignment will be generated for the object's authorized user requesting them to correct the inconsistency. The authorized user will also receive an email informing them of their new assignment.

The timely completion of the assignment triggered by a consistency monitor can be tracked by the solution administrator. The **Pending Corrections** page view displays all assignments and assigned to the authorized user of each object found via the query and allows the solution administration to monitor and track the completion of all assignments by the specified target dates. An overview of all inconsistent objects found via the query are displayed in the **Inconsistent Objects** page view for the monitor.

In addition to the general requirements for monitors described in the section Configuring Monitors to Track Objects in Alfabet , the following must be configured to implement consistency monitors:

- The Alfabet query or native SQL query assigned to the consistency monitor must be entered in a text field in the **Consistency Monitor** editor. For more information about defining Alfabet queries and the special rules that apply to the use of native SQL queries for Alfabet configurations, see *Defining Queries for Alfabet Configurations* in the reference manual *Configuring Alfabet with Alfabet Expand*.
- The following text templates should be configured in the configuration tool Alfabet Expand for the email notifications sent when an assignment is created if a consistency monitor is triggered. For more information, see the section *Configuring Text Templates for Email Notifications* in the reference manual *Configuring Alfabet with Alfabet Expand*. See the chapter *Overview of Preconfigured Text Templates and Their Variables* in the reference manual *Configuring Alfabet with Alfabet Expand - Appendix* for an overview of the available text templates for notification monitors and their permissible variables.
  - The text template `ConsistencyMonitorDefault` in the `MON` folder is used for the assignments generated in the context of the consistency monitors. The relevant text template must be selected for a consistency monitor in the **Text Template** attribute in the **Consistency Monitor** editor. The text template is used to fill the **Description** attribute of the assignment.
  - The text template `ConsistencyMonitorMail` in the `MON` folder is used when an assignment is generated via batch process for an inconsistent object found by a consistency monitor. If a batch process is configured, the email notification will be sent to the authorized user of an object when a specified consistency is detected. For more information about setting up a batch process, see the section *Batch Processing for Monitors and Change Management with AlfaBatchExecutor.exe* in the reference manual *System Administration*.
  - If a custom text template is selected, the email assignments sent out for the consistency monitor and the assignments generated for the objects found by the consistency monitor will display the same information. The caption specified for the text template is displayed as the subject line of the email and the name of the assignment. The text specified for the text template will be shown as the description of the assignment and the body of the email, which will also include links to each object targeted by each assignment.

Please note that assignments are created for consistency monitors. Therefore, if the message logging is activated for assignments, then the emails sent in the context of these monitors will be logged. For more information about the **Email Message Log** functionality, see the section Tracking the Email Messages Sent in the Context of Alfabet Functionalities in the reference manual *System Administration*.

The following information is available:

Creating a Consistency Monitor

To configure a consistency monitor:

1) In the **Solution Admin** menu, click **Consistency Monitors**.
2) In the toolbar, click **New** > **Create New Monitor**. The **Consistency Monitor** editor opens.
3) Enter information in the fields as needed. Each field is defined below:

   General tab:

   - **ID**: Alfabet assigns a unique identification number to each object in the inventory. This number cannot be edited.
   - **Name**: Enter a unique name for the consistency monitor.
   - **Monitor Frequency**: Enter the frequency that the monitor should be executed. Choices include Daily, Weekly, and Monthly. Weekly monitors are executed on the same day of the week that the monitor is created.

     For example, a monitor created on Thursday, March 27 will be executed on Thursday, April 3, Thursday, April 10, etc.

   - **Start Date**: Enter the start date to determine when the consistency monitor should be executed via the batch process. Enter the date in the appropriate format or select the start date in the calendar. The current date is the default.
   - **End Date**: Enter the end date to determine when the consistency monitor should no longer be executed via the batch process. Enter the date in the appropriate format or select the end date in the calendar.
   - **Correction Period**: Enter the number of days that the assignee of the assignment has to correct the detected inconsistency.
   - **Mandatory Assignment**: Select whether the **Type** attribute should be defined as Mandatory for the assignments that are generated for the inconsistent objects. If the **Mandatory Assignment** attribute is not checked, the assignment created will be of the type Optional.
   - **Default Assignee for Monitor**: Select the Alfabet user that should be assigned responsibility for the assignment if the inconsistent object has no authorized user(s) defined.

     If the inconsistent object has no authorized user and values are not defined for the properties **Default Assignee for Monitor** and **Originator**, then the values configured for the **Assignee** and **Originator** attributes in the XML object **SystemMonitorDefaultDef** will be used. For more information, see the section *Configuring Assignments for System-Wide Date Monitors* in the reference manual *Configuring Alfabet with Alfabet Expand*.

- **Originator**: Select the Alfabet user that should be assigned as the originator of the assignment for the inconsistent object.
- **Text Template**: Select the text template that should be sent when assignments are automatically generated for inconsistent objects. The text template is used to fill the **Description** attribute of the assignment. The name of the assignment is copied from the name of the consistency monitor. The text templates are configured in the configuration tool Alfabet Expand. For more information, see the section *Configuring Text Templates for Email Notifications* in the reference manual *Configuring Alfabet with Alfabet Expand*.
- **User Profile to Access Object in Notification**: Select the user profile that will be used by email recipients to access the object in Alfabet via the hyperlink in the monitor's notification email.

Query tab: Define an Alfabet query or a native SQL query that results in a dataset displaying all inconsistent objects. The tabular output of the query is displayed in the **Inconsistent Objects** page view.

For more information about how to define Alfabet queries and about the rules that apply to the definition of native SQL queries in Alfabet configurations, see *Defining Queries for Alfabet Configurations* in the reference manual *Configuring Alfabet with Alfabet Expand*.

Please note that when defining native SQL queries, the first argument of the `SELECT` statement must be the `REFSTR` of the inconsistent object. The first argument is not displayed in the result set of the **Inconsistent Objects** page view.

4) Click **OK** to save the consistency monitor of **Cancel** to exit without saving.

If you delete a consistency monitor, it will be irrevocably deleted from the Alfabet database.

Activating and Deactivating the Consistency Monitor

You must activate the consistency monitor in order for it to be triggered when the batch process is initiated. To activate a consistency monitor, select the relevant consistency monitor in the **Consistency Monitors** functionality and click **Monitors** > **Activate Monitors**. You can simultaneously select multiple monitors by using the key combination CTRL + click to simultaneously select several objects in the table. A checkmark is displayed in the **Is Active** column for all currently active consistency monitors.

To deactivate a consistency monitor so that it is not triggered when the batch process is initiated, select the relevant consistency monitor in the table and click **Monitors** > **Deactivate Monitors**. The checkmark will be removed from the **Is Active** column.

Viewing the Objects Found by the Consistency Monitor

The **Inconsistent Objects** page view displays all objects that have been found by the Alfabet query or native SQL query configured for the selected consistency monitor.

To access the **Inconsistent Objects** page view, select the relevant consistency monitor in the **Consistency Monitors** functionality and click the **Navigate**  button. In the object profile that opens, click **Inconsistent Objects**. The attributes displayed for the objects will depend on the configuration of `Show` properties in the Alfabet query or `SELECT` arguments of the native SQL query.

An assignment will be generated for the authorized user or default assignee(s) for all objects displayed in the table. The assignments that have been generated for the authorized user or default assignee(s) for all objects listed in the **Inconsistent Objects** page view are displayed in the **Pending Corrections** page view

Viewing the Assignments Generated for Inconsistent Objects

The **Pending Corrections** page view lists all assignments that have been generated for the objects found by the selected consistency monitor. If an assignment has already been generated for an inconsistent object via a previous batch job, additional assignments will not be generated for the object upon subsequent executions of the batch processes. Assignments that are highlighted in red are assignments that have not been completed by the target date.

The assignee of an assignment will see the assignments generated via the consistency monitor in the **My Assignments** functionality. After an assignment's status is changed to a closed status, the assignment will be automatically removed from the views associated with the assignment capability. However, the originator of the assignment can explicitly choose to display any assignments that have been closed after a specific date in the **Sent Assignments** view. For more information about the assignments functionality and how to access closed assignments, see the section *Sending and Receiving Assignments for Alfabet Objects* in the reference manual *Getting Started with Alfabet*.

You can limit the data displayed in the table by defining the filters. In the **Release Statuses** field, select one or more release statuses that you want to display. To view only the assignments that have been closed after a particular day, select the checkbox next to the **Show Closed Assignments with Target Date After** field and enter the specified date in the calendar field. The table displays the following columns:

- **ID**: Alfabet assigns a unique identification number to each object in the inventory. This number cannot be edited.
- **Assignment Type**: Displays whether the assignment is optional or mandatory. Reminder notices may be sent to ensure that an assignment is processed by its target date.
- **Name**: Displays the assignment's title. The name is derived from the consistency monitor.
- **Target Date**: Displays the target date or deadline for the assignment.
- **Status**: Displays the completion status of the assignment.
- **Last Changed Date**: Displays the date when the assignment's status was last changed.
- **Object ID**: Displays the identification number of the object targeted by the consistency monitor.
- **Object Name**: Displays the name of the object targeted by the consistency monitor.
- **Originator Name**: Displays the family name of the user that is defined as the originator of the assignment for the inconsistent object.
- **Originator First Name**: Displays the first name of the user that is defined as the originator of the assignment for the inconsistent object.
- **Assignee Name**: Displays the family name of the user that is assigned responsibility for the assignment for the inconsistent object.
- **Originator First Name**: Displays the first name of the user that is assigned responsibility for the assignment for the inconsistent object.

The following is possible:

- To view the description of the inconsistency, click the object to view the description in the preview pane. The description text is based on the text template assigned to the consistency monitor in the **Text Template** field in the **Consistency Monitor** editor.
- To navigate to the object profile of the object that the assignment is about, select the assignment and click the **Navigate** button.
- To edit the assignment, select the assignment and click the **Edit** button and edit the data, as needed. Each field is defined below:
  - **ID**: Alfabet assigns a unique identification number to each object in the inventory. This number cannot be edited.
  - **Title**: Title that clarifies the purpose of this assignment.
  - **Target Date**: Enter the target date or deadline for the assignment in the format appropriate to your cultures or click the **Calendar** icon to select a date.
  - **Status**: Define or update the completion status of the assignment.

    The set of release statuses available for the class Assignment are configured by your solution designer in the configuration tool Alfabet Expand. For more information, see the section *Defining the Statuses Used for the Assignment Capability* in the reference manual *Configuring Alfabet with Alfabet Expand*.

  - **Assigned To**: Click the **Search** icon to select the user that this assignment is assigned to.
  - **Type**: Define whether completion of the assignment is **Mandatory** or **Optional**. Please note the following consequences if the assignee does not complete the assignment by the specified target date.
    - Mandatory assignments: If the assignee fails to complete a mandatory assignment by the target date, the status of the assignment will automatically change to a re-assigned status when the target date is reached. Consequently, the assignment will be automatically returned to the originator of the assignment and will be removed from the assignee's **My Assignments** view.
    - Optional assignments: If the assignee fails fail to complete an optional assignment by the target date, the status of the assignment will automatically change to a closed status when the target date is reached. Consequently, the assignment will be automatically removed from the assignee's **My Assignments** view. Optional assignments will NOT be returned to the originator of the assignment.
  - **Object**: Displays the object that this assignment is about.
  - **Description**: Enter a meaningful description about the assignment.
  - **Remarks**: Enter any necessary remarks about the assignment.
  - **Documents**: Attach a document stored in the **Internal Document Selector** or define a URL that pertains to the assignment. For more information about attaching documents and defining URLs, see the *Attachments Page View*.

Defining Notification Monitors

The **Notification Monitors** functionality in the **Solution Administration** module allows you to create notification monitors.

A notification monitor is a type of monitor that allows email notifications to be automatically triggered based on configured Alfabet queries or native SQL queries. The queries specify the targeted objects and their object class properties as well as the users who shall be notified about the objects found by the queries.

As part of the monitor configuration, you define when and how often the objects should be monitored. An overview of all objects found via the query are displayed in the **Notification Objects** page view for the monitor.

The general requirements for monitors described in the section Configuring Monitors to Track Objects in Alfabet must be fulfilled. Please note that an Alfabet query can be defined for the notification monitor in the configuration tool Alfabet Expand. For more information, see the section *Defining Queries for Alfabet Configurations* in the reference manual *Configuring Alfabet with Alfabet Expand*.You can check the syntax of the Alfabet query in the configuration tool Alfabet Expand via the **Check ALL AQL Queries** functionality. For more information about checking the query syntax, see the section *Testing Alfabet Queries* in the reference manual *Configuring Alfabet with Alfabet Expand*.

The following information is available:

- Creating a Notification Monitor
- Activating and Deactivating the Notification Monitor
- Deleting a Notification Monitor
- Viewing the Objects Found by the Notification Monitor

Creating a Notification Monitor

To configure a notification monitor:

1) In the **Solution Admin** menu, click **Notification Monitors**.
2) In the toolbar, click **New** > **Create New Monitor**. The **Notification Monitor** editor opens.
3) Enter information in the fields as needed. Each field is defined below:

   General tab:

   - **ID**: Alfabet assigns a unique identification number to each object in the inventory. This number cannot be edited.
   - **Name**: Enter a unique name for the notification monitor.
   - **Monitor Frequency**: Enter the frequency that the monitor should be executed. Choices include Daily, Weekly, and Monthly. Weekly monitors are executed on the same day of the week that the monitor is created. For example, a monitor set to a frequency of weekly and created on Thursday, March 27 will be executed on Thursday, April 3, Thursday, April 10, etc.
   - **Start Date**: Enter the start date to determine when the notification monitor should be executed via the batch process. Enter the date in the appropriate format or select the start date in the calendar. The current date is the default.

- **End Date**: Enter the end date to determine when the notification monitor should no longer be executed via the batch process. Enter the date in the appropriate format or select the end date in the calendar.
- **Text Template**: Select the text template that defines the message the user receives via email when the monitor sends a notification. Alternatively, you can leave this field empty and define an ad-hoc text in the **Text Template** tab. A definition in the **Text Template** tab has precedence over a selection made in the **General** tab, but additional languages versions cannot be created for the ad-hoc text written in the **Text Template** tab.

  If multiple languages are used in your Alfabet solution, it is recommended that you implement a configured text template for which the relevant language versions can be created. Text templates for the notification capability can be created in Alfabet Expand. For more information, see the section *Configuring Text Templates for Email Notifications* in the reference manual *Configuring Alfabet with Alfabet Expand*.

- **User Profile to Access Object in Notification**: Select the user profile that will be used by email recipients to access the object in Alfabet via the hyperlink in the monitor's notification email.

Object Query tab: Define a valid Alfabet query or native SQL query to find the objects for which the monitor shall be executed. When defining an Alfabet query, the object class targeted by the monitor must be defined as the `FIND` class. When defining a native SQL query, the first `SELECT` property must return the values of the `REFSTR` property of the selected object. The query may use the parameter `LAST_EXECUTION` to compare date information of the objects with the last execution date for the monitor. The **Objects** page view displays the tabular result of the query. For information about defining Alfabet queries and about the rules that apply to the definition of native SQL queries in the context of Alfabet, see *Defining Queries* in the reference manual *Configuring Alfabet with Alfabet Expand*.

Recipient Query tab: Define a valid Alfabet query or native SQL query finding the users for which the monitor shall be executed. When defining an Alfabet query, the object class `PERSON` must be defined as the `FIND` class. When defining a native SQL query, the first `SELECT` property must return the values of the `REFSTR` property of objects of the object class `PERSON`. The query may use the parameter `BASE` to specify that the recipient query is executed for each of the objects found by the object query. For information about defining Alfabet queries and about the rules that apply to the definition of native SQL queries in the context of Alfabet, see *Defining Queries* in the reference manual *Configuring Alfabet with Alfabet Expand*.

Text Template tab: Define a text template including the permissible variables for the email notification sent out by the monitor. The subject line of the email notification will be automatically generated and display **Notification Monitor** <NameOfMonitor>, whereby the name of the monitor is taken from the **Name** attribute defined in the **General** tab. Please note that if a text template is specified in the **Text Template** tab, it supersedes an existing text template selected in the **General** tab. Additional languages versions cannot be created for the ad-hoc text written in this tab.

4) If you are defining an ad-hoc text in the **Text Template** tab, you can use any of the following variables. The variables must be written in curly brackets (for example: {`Person:Name`}.

Variable	Content
`Person:Name`	Family name of the user to whom notification about the monitor is to be sent.

Variable	Content
Person:FirstName	First name of the user to whom notification about the monitor is to be sent.
Person:Email	Email address of the user to whom notification about the monitor is to be sent.
Object:<Property>	A scalar object class property defined for the object that the monitor notification is about. Enter the name of the property in the <Property> placeholder.
Object:ObjectType	The object class of the object that the monitor notification is about.
Object:RefImage	The object that is being monitored. The object's configured image properties will be displayed. The image properties are defined via the **Image Properties** attribute in the class setting assigned to the user profile that will be used to access Alfabet.
Link:ObjectView	The link to the object profile of the object that the monitor is about. The class setting for the relevant user profile will determine whether the object view for the object class or object class stereotype will open when the link is clicked.

5) Click **OK** to save the notification monitor or click **Cancel** to exit without saving.

Activating and Deactivating the Notification Monitor

You must activate the notification monitor in order for it to be triggered when the batch process is initiated. To activate a notification monitor, select the relevant notification monitor in the table and click **Monitors** > **Activate Monitors**. You can simultaneously select multiple monitors by using the key combination CTRL + click to simultaneously select several objects in the table. A checkmark is displayed in the **Is Active** column for all currently active notification monitors.

To deactivate a notification monitor so that it is not triggered when the batch process is initiated, select the relevant notification monitor in the table and click **Monitors** > **Deactivate Monitors**. The checkmark will be removed from the **Is Active** column.

Deleting a Notification Monitor

1) In the table, click the notification monitor you want to delete.
2) In the toolbar, click the **Delete** button. The notification monitor is deleted.

Viewing the Objects Found by the Notification Monitor

The **Notification Objects** page view displays all objects that have been found by the Alfabet query associated with the selected notification monitor.

To access the **Notification Objects** page view, select the relevant notification monitor in the **Notification Monitors** functionality and click the **Navigate** button. In the object profile that opens, click **Notification Objects**. The attributes displayed for the objects will depend on the configuration of `Show` properties in the Alfabet query.

Managing the Owners of Activity, Inactivity, and Date Monitors

The **User Monitor Management** functionality allows the ownership of activity, inactivity, and date monitors to be managed and monitors that no longer have an owner in the enterprise to be deleted or reassigned. You can display all activity, inactivity, and date monitors owned by a specific user by selecting the user in the **Monitor Owner** field and clicking the **Update** button.

The monitors owned by the specified user are displayed. The table is structured according to the monitor type. The following columns are displayed:

- **Name**: Displays the name of the monitor
- **Start Date**: Displays the start date of the monitor.
- **End Date**: Displays the end date of the monitor.
- **Frequency**: Displays the frequency that the monitor is executed.
- **Status**: Displays whether the monitor is active or inactive.

You can do the following:

- Change the owner for an individual monitor: Select the monitor in the table and click **Edit** > **Change Owner for Selected Object(s)**.
- Change the owner for an all monitors displayed in the table: Click **Edit** > **Change Owner for All Object(s)**.
- Edit the monitor: Select the monitor in the table and click the **Edit** button and make changes as needed.
- Delete the monitor: Select the monitor in the table and click the **Delete** button.

Chapter 7: Tracking and Managing Workflows

A workflow is a collaborative process made up of workflow steps that are typically carried out by one or more users. A workflow is based on a configured workflow template that determines the sequence of workflow steps that are to be performed on a specific object and its references by specified user(s). Workflow steps may have specific pre- and post-conditions that determine different paths to take in the workflow depending on whether the conditions are or are not met.

Typically, the workflow owner is the user who initiates and is responsible for maintaining the workflow. When a workflow is initiated by the workflow owner and when a workflow advances to the next workflow step, relevant users may be informed via automatically generated emails of their impending responsibility for the workflow step. This functionality ensures that all relevant users are informed and reminded of their responsibilities in the collaborative workflow. The option to refuse, delegate, and pause workflow steps, remind users of an impending target date for a workflow step as well as redirect a workflow that has encountered an error enables workflow owners and workflow administrators to keep track, coordinate, and manage the completion of each workflow step and the workflow as a whole.

The **Workflow Administration** functionality allows a workflow administrator to track the progress of the workflows and resolve any existing conflicts. For example, if a workflow step's target date has expired or if no responsible user is assigned to a workflow step or found via the query specified for the workflow template, the workflow step will escalate. These errors must be dealt with and resolved by either the workflow owner or the workflow administrator. The workflow administrator may also carry out other tasks such as rechannelling the workflow to another workflow step, changing the state of a workflow template or the owner of a workflow.

For an overview of the general use of the workflow capability as well as the tasks relevant to the workflow owner and the user responsible for performing a workflow step, see the chapter *Executing Workflows and Participating in Workflow Steps* in the reference manual *Getting Started with Alfabet*. Workflow templates and their associated workflow steps are configured in the configuration tool Alfabet Expand. For more information about configuring a workflow, see the section *Configuring Workflows* in the reference manual *Configuring Alfabet with Alfabet Expand*.

Workflows in Alfabet are based on a workflow template that is configured in the configuration tool Alfabet Expand by your company's workflow designer.

A workflow template is a customer-defined blueprint for one or more workflows. The template specifies what object class is the point-of-departure in the workflow, which user groups and/or user profiles may initiate and administrate the workflow, which workflow steps comprise the workflow as well as their sequence, any possible per- and post-conditions or update actions associated with a workflow step, and what kinds of workflow notifications should be sent to collaborating users in which contexts. The user who creates the workflow template is the workflow template owner.

A workflow template must have the attribute **Workflow State** attribute set to `Plan` to be configured and validated. Once the workflow template is completed and approved, the attribute **Workflow State** must be switched to `Active` to make it available to the user community. Once the workflow template is available in Alfabet, a permitted user may initiate a workflow based on the workflow template. Multiple workflows may be simultaneously initiated and running for a workflow template.

The **Workflow Administration** functionality displays all workflow templates in the enterprise, regardless of whether they have the state **Active**, **Plan**, or **Retired**. You can trigger workflows for any of these workflow templates. A workflow will be started for every object found by the query configured for the workflow template.

Typically, the user who triggers the workflow is the owner of the workflow. The workflow owner is responsible for maintaining the workflow and ensuring its progress towards completion. The option to refuse, delegate, withdraw, and suspend workflow steps, remind users of an impending target date for a workflow step as well as redirect a workflow that has encountered an error enables workflow owners and workflow administrators to keep track, coordinate, and manage the completion of each workflow step and the workflow as a whole.

When a workflow is initiated by the workflow owner, relevant users are typically informed via automatically generated emails of their impending responsibility in the workflow. Whether emails are automatically sent when a workflow step is entered will depend on the configuration of the workflow.

The table in the **Workflow Administration** functionality is divided into two sections. The **Workflows** section displays all workflows and their current workflow step. Click the  to expand a workflow in order to view it's currently active workflow step. The **Workflow Templates** section of the table displays all workflow templates.

Please note that a maximum of 100 workflows and workflow templates will be displayed in the view. If the combination of workflows and workflow templates exceeds 100, navigate to the next page of results via the floating toolbar in the lower right corner of the view.

It may be that an excessive amount of workflow templates and workflows are displayed. Filters allow you to limit the workflow templates and workflows displayed according to various criteria. For example, to view all workflows that are currently in progress, you could select the value **Running** in the **Workflow State** filter. You can define one or more of the following filters as needed:

- **Workflow Templates**: Click the arrow to select one or more workflow templates to display. The selected workflow templates and the workflows based on the selected workflow templates will be displayed in the table.
- **Workflow State**: Click the arrow to limit the workflows displayed in the table to one or more workflow states that you select.

  The state Initiation is obsolete and should not be selected.

- **Workflow End Date**: Enter a date to display all workflows with an end date before the specified date. The resulting dataset will also include workflows with no end date.
- **Search Pattern**: Define search criteria to find workflows. The criteria you enter will be used to find a match in the caption of the workflow, workflow steps, or ID or name of the object managed in the workflow. If you do not know the full name, use the * symbol as a wildcard character. Enter ABC* to find all objects beginning with the letters ABC, enter *ABC to find objects ending with ABC, or *ABC* to find objects containing the letters ABC.

Click the **Update** button to apply your filter settings to the information displayed. The table displays all workflows and workflow templates configured for your enterprise that match your filter criteria. The following columns are displayed:

- **Workflow**: Displays the caption of the workflow template/workflow.
- **Workflow ID**: Displays the identification number of the workflow that is being processed.
- **Workflow State**: Displays the current state of the workflow template/workflow.
- **Owner**: Displays the name of the current owner of the workflow template/workflow.
- **Workflow Start**: Displays the timestamp showing when the workflow was started.
- **Object ID**: Displays the identification number of the object that is targeted by the current workflow step.
- **Object Name**: Displays the name of the object that is targeted by the current workflow step.
- **Current Workflow Step**: Displays the name of the workflow step that is currently in process in the workflow.

- **Workflow Step Start**: Displays the timestamp showing when the workflow step was started.
- **Locked By**: Displays the current user who is currently working on the workflow step.

The following information is available regarding the initiation of workflows:

- Manually Starting an Automatic Workflow
- Accessing More Information About the Workflow
- Changing the State of a Workflow Template
- Closing Workflows That Are Running
- Changing the Owner of a Workflow Template or Workflow
- Reassessing the Responsibility of Workflow Steps
- Checking a Workflow Step's Deadline
- Checking a Workflow Step's Post-Conditions and Closing the Workflow Step
- Releasing the Lock on a Workflow
- Deleting a Workflow
- Understand the Course of the Workflow
- Suspending, Resuming, or Withdrawing the Workflow
  - Understanding Event Values and Resolving Workflow Errors
- Fixing a Workflow with an Error State
  - Redirecting a Workflow Step to Another Step
  - Adding Responsible Users to a Workflow Step
  - Delegating a Workflow Step to Another User

Manually Starting an Automatic Workflow

Automatically started workflows are typically started via a batch process initiated by your system administrator. However, as the workflow administrator, you can manually trigger the start of a workflow template that has been configured to start automatically. If you trigger a workflow in the **Workflow Administration** functionality via the **Start Automatically** button, then the workflow owner will be based on the owner of the workflow template. If no workflow template owner is defined, then the workflow administrator triggering the workflow via the **Start Automatically** button will be the workflow owner.

The **Start Automatically**  button will only be available for workflow templates for which the **Automatic Start** attribute is set to `True` in Alfabet Expand. The configuration of the workflow template must include a query to find the objects that are the target of the workflows. If such a query has not been configured for the workflow template, new workflows will not be created, and an error message will be displayed.

For detailed information about configuring the workflow template, see the section *Defining the Start of Workflows Based on the Workflow Template* in the reference manual *Configuring Alfabet with Alfabet Expand*. For more information about initiating a batch process for workflows, see the section

Batch Processes for Workflows with AlfaWorkflowCommandPrompt.exe in the reference manual *System Administration* l.

To manually trigger a workflow template configured to start automatically, select the workflow template and click the **Start Automatically** button. The **Start Automatically** button is only enabled for workflow templates configured to start automatically. Confirm the **Info** message. Once the workflows have been initiated, the **Workflow State** column for the initiated workflows will display `Running`. Once the workflow has been initiated, emails are typically generated and the user(s) who are responsible to carry out the subsequent workflow step will be notified. The responsible user(s) will find the workflow step in their **My Workflow Activities** functionality. For more information about this functionality, see *Performing the Workflow Steps That You Are Responsible For* in the reference manual *Getting Started with Alfabet*.

Accessing More Information About the Workflow

There are a number of different means to access more information about the workflow in the **Workflow Administration** functionality:

- To display the workflow template's preview, click-and-hold the workflow template. The following attributes are displayed:
  - **Workflow Name**: The technical name of the workflow template.
  - **Workflow Caption**: The caption of the workflow template.
  - **Workflow Description**: Information describing the purpose of the selected workflow template.
  - **Workflow Base Class**: The base class of the workflow template. This is the object class that the workflow begins with when the initial workflow step is performed.
  - **Workflow State**: The state of the workflow template.
- To open the object profile of the workflow template, select the workflow template and click the **Navigate** button in the toolbar. The following information is displayed:
  - **Caption**: The caption of the workflow template.
  - **Base Class**: The base class of the workflow template. This is the object class that the workflow begins with when the initial workflow step is performed.
  - **Workflow Template State**: The state of the workflow template.
  - **Owner**: The name of the current workflow template owner.
  - **Description**: Information describing the purpose of the workflow template.
- To display the workflow's preview, click-and-hold the workflow. The following attributes are displayed:
  - **ID**: The identification number of the workflow that is being processed.
  - **Workflow Name**: The technical name of the workflow.
  - **Workflow Caption**: The caption of the workflow.
  - **Workflow Base Class**: The base class of the workflow template. This is the object class that the workflow begins with when the initial workflow step is performed.

- **Workflow State**: The state of the workflow.
- **Automatic Start**: Displays True if the workflow was automatically started via a batch process. For more information about configuring and initiating a batch process for workflows, see the section *Batch Processes for Workflows with AlfaWorkflowCommandPrompt.exe* in the reference manual *System Administration*.

- To open the object profile of the workflow, select the workflow and click the **Navigate** button in the toolbar. The following information is displayed:
  - **ID**: The identification number of the workflow that is being processed.
  - **Workflow Name**: The technical name of the workflow.
  - **Workflow Caption**: The caption of the workflow.
  - **Workflow Base Object Class**: The base class of the workflow template. This is the object class that the workflow begins with when the initial workflow step is performed.
  - **Start Base Object**: The name of the object that the workflow was initiated for.
  - **Workflow State**: The state of the workflow.
  - **Owner**: The name of the current workflow owner.
  - **Initiator**: The name of the user who initiated the workflow.
  - **Locked By**: The name of the user who is performing the current workflow step.
  - **Start**: The timestamp showing when the current workflow was started.
  - **End**: The timestamp showing when the current workflow was completed.
  - **Current Step**: The workflow step that is currently being processed.
  - **Workflow Template**: The name of the workflow template that the workflow is based on.
  - **Comment**: Information describing the purpose of the selected workflow.

  The following page views are also available in the workflow object profile:
    - For more information about the **Workflow Event Trace** page view, see the section Suspending, Resuming, or Withdrawing the Workflow.
    - For more information about the **Workflow Diagram** page view, see the section Understand the Course of the Workflow.
    - For more information about the **Active Workflow Steps** page view, see the section Fixing a Workflow with an Error State.

- To view the workflow and all of its potential workflow steps, select the workflow and click the **Show Workflow Diagram** button to open the **Workflow Diagram** page view. Alternatively, you can click-and-hold the workflow and click **Details** to navigate to the **Workflow Diagram** page view. For more information about how to read the diagram, see the section *Understand the Course of the Workflow*.
- To open the workflow step's preview, click-and-hold the workflow step. The following attributes are displayed:

- **ID**: The identification number of the workflow that is being processed.
- **Workflow Step Name**: The technical name of the workflow step.
- **Workflow Step Caption**: The caption of the workflow step.
- **Workflow Step State**: The state of the workflow step.
- **Is Confirmable**: Displays `True` if the workflow step can be processed or `False` if the workflow step either needs to be performed, is paused, or is blocked by an error.
- **Workflow Step Base Object Class**: The base class of the object that the workflow step targets.

- To open the object profile of the workflow step, select the workflow step and click the **Navigate** button in the toolbar. The following information is displayed:
  - **Workflow Step Description**: Displays the description of the workflow step.
  - **ID**: Displays the identification number of the selected workflow step.
  - **Name**: Displays the technical name of the selected workflow step.
  - **Caption**: Displays the caption of the selected workflow step.
  - **Workflow**: Displays the name of the workflow that the workflow step belongs to.
  - **Previous Step**: Displays the name of the workflow step that was performed before the selected workflow step.
  - **Base Object**: Displays the name of the object that the workflow started with.
  - **Workflow Step State**: Displays the state of the workflow step.
  - **Ready**: Displays `True` if the workflow step can be processed. Displays `False` if the workflow step is being pause or is blocked by an error.
  - **Locked By**: Displays the name of the person currently performing the workflow step.
  - **Automatic Closure**: Displays `True` if the workflow step will be automatically closed and advance to the next workflow step via a batch process if all post-conditions have been satisfied. The user responsible for the workflow step does not need to manually confirm the workflow step in this case. Displays `False` if the workflow step must be manually confirmed by the responsible user.
  - **Due Date**: Displays the target date by which the workflow step should be completed.
  - **Responsible Users**: Displays the names of the users that are responsible for performing the workflow step.
  - **Completed Users**. Displays the names of the users that have performed the workflow step.

  The following page views are also available in the workflow object profile:

  - For more information about the **Active Step Activities** page view, see the section *Accessing Details about the Current Workflow Step*.
  - For more information about the **Activities of Preceding Steps** page view, see the section *Accessing Details about the Preceding Workflow Step*.

Changing the State of a Workflow Template

It may be necessary to change the state of a workflow template. For example, if further configuration is required for a workflow template (for example, a new post-condition is required for a workflow step or the query used to find users must be refined, you could change the state from **Active** to **Plan** in order to take the workflow template out of production until the configuration has been revised or if the workflow template is no longer relevant for the enterprise, you could change the state to **Retired**. Please note however that if there are errors in the workflow template configuration, the state cannot be changed. In this case, an error message will be displayed describing the configuration errors that must be fixed before the workflow template's state can be changed.

A workflow template may be assigned any of the following states:

- **Plan**: A workflow template that is in a **Plan** state may have modifications made to the configuration. The workflow template is only visible in the Alfabet interface for administrative purposes. A workflow template in a **Plan** state can only be changed to an **Active** state if there are no errors in the configuration. If you attempt to change a template with the state **Plan** to **Active** and the state does not changed as anticipated, please contact your workflow designer to determine what needs to be corrected in the configuration.
- **Active**: A workflow template that is in an **Active** state is visible in the **Start New Workflow** section of the **My Workflows** functionality for users with a permitted user profile/user group affiliation. New workflows can be created and started based on an active workflow template. If a workflow template must be modified after it has been activated, the state must be returned to **Plan**. However, any modifications made to the workflow template will not impact workflows that have already been started.
- **Retired**: Once a workflow template is set to **Retired**, it is removed from the **Start New Workflow** section of the **My Workflows** functionality and no new workflows may be initiated based on this workflow template. The workflow template cannot be modified. Any workflows that are currently running when the workflow template is retired will continue until they reach the state **Finished** or are deleted by the workflow administrator or workflow owner.

To change the state of the workflow template, select the workflow template in the table and click **Manage** > **Change Workflow State to [WorkflowState]**. The workflow template state will be updated in the **Workflow State** column.

Closing Workflows That Are Running

Running workflows that may be running too long without being completed may be closed by the workflow administrator or workflow owner. To close one or more running workflows, select the running workflows in the **Workflow** section of the table and click **Manage** > **Close Running Workflows**. The workflow state will be changed to **Withdrawn** and the workflow will be removed from the dataset of running workflows. To view the closed workflow, select the option **Withdrawn** in the **Workflow State** filter and click the **Update** button.

Changing the Owner of a Workflow Template or Workflow

The owner of a workflow is determined as follows:

- If a workflow is started manually in the **My Workflows** functionality, the current user starting the workflow is the workflow owner.

- If a workflow is started automatically via the batch tool `AlfaWorkflowCommandPrompt.exe`, then the workflow owner is based on the owner of the workflow template. If no workflow template owner is defined, then the current user executing the batch job is the workflow owner.
- If a workflow is started in the **Workflow Administration** functionality via the **Start Automatically** button, the workflow owner is based on the owner of the workflow template. If no workflow template owner is defined, then the current user triggering the workflow in the **Workflow Administration** functionality is the workflow owner.

If necessary, you can change the owner of the workflow to another user who has access to the **My Workflows** functionality via his/her user profile. The workflow object profile will display the original owner's name in the **Initiator** attribute and the new workflow owner's name in the **Owner** field.

1) Select the workflow in the table and click the **Change Owner** button in the toolbar.
2) The selector opens. Enter search criteria, as needed, and click **Search**.
3) Select the new workflow owner and click **OK**. The new workflow owner is displayed in the **Owner** column.

Reassessing the Responsibility of Workflow Steps

If the responsibilities in your enterprise have changed, it may be necessary to update the users who are responsible for a workflow. By means of the **Reassess the Responsibility of Workflow Steps** functionality, you can reassess the current workflow step of a selected workflow and update the assignment of responsible users to these workflow steps. When one of the **Reassess the Responsibility of Workflow Steps** options is selected, the batch configuration `RescanResponsibility` in the `AlfaWorkflowCommandPrompt.exe` will be triggered.

The instructions associated with the responsibility definition of the workflow step as well as the query defined for the corresponding workflow step action of type `Notification` will be executed to find new responsible users. Any new users found since the last execution of the query will be sent the email notification that is configured for the `OnEnterStep` workflow step action. The workflow step will be displayed in the **My Workflow Activities** functionalities for all users found by the associated query.

For more information about the configuration of batch process for workflows, see the section *Batch Processes for Workflows with AlfaWorkflowCommandPrompt.exe* in the reference manual *System Administration*.

To reassess the ownership of a workflow step, select the workflow step and click **Reassess the Responsibility of Workflow Steps**. Select one of the following options:

- **Reassess Steps and Skip Delegations** to update the responsible users of all workflows steps except the steps that have been reassigned to a different user by means of the **Delegate** function. Workflow steps can be delegated by the user responsible for the step in the or the object profile of the workflow step. For more information about the **Delegate** function, see the section Delegating a Workflow Step to Another User.
- **Reassess Steps and Reset Delegations** to update the responsible users of all workflows steps including the workflow steps that have been reassigned to a different user by means of the **Delegate** function.

- **Reassess Steps and Reapply Delegations** to update the responsible users of all workflows steps and reapply any step delegation decisions users were previously performed by means of the **Delegate** function. In this case, responsible users may be added to or removed from a delegated workflow step.

The **Workflow Event Trace** page view will be automatically update with new responsible users if any new users have been found for the workflow step.

Checking a Workflow Step's Deadline

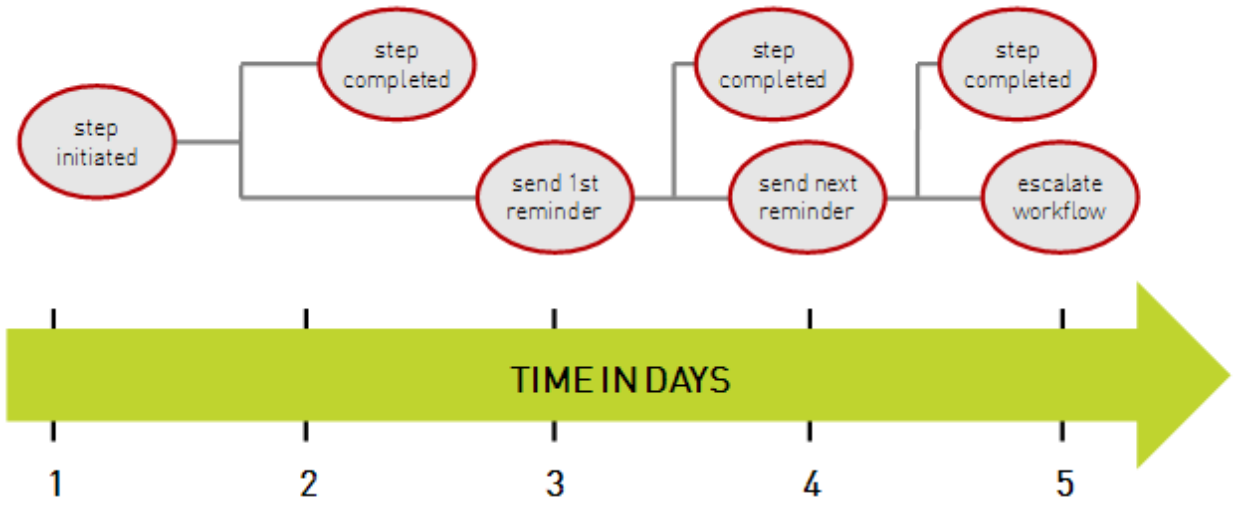


FIGURE: Example of the configuration and execution of a performance duration definition.

Workflow step deadlines are typically checked via a batch process. The **Check Performance Duration** functionality triggers the configured batch process for a selected workflow in order to check the deadline for the current workflow step.

To check the deadlines for a current workflow step in a workflow, select the workflow in the table and click **Manage** > **Check Performance Duration**. A message will be displayed with information about the check. Click **OK** to confirm the message.

If relevant, reminders will be automatically sent notifying responsible users that the deadline is approaching or has passed. If the deadline has passed and an expiration step has been configured for the selected workflow step, then an expired workflow step will automatically advance to the configured step. If this is the case, this series of events will be described in the **Workflow Event Trace** page view for the selected workflow.

If no subsequent step has been configured, the workflow step will be escalated. The status of the workflow step will be changed to `StepExpired` and the workflow step must be manually redirected to a subsequent workflow step via the **Go To** button in the **Workflow Event Trace** page view. For more information, see the section Fixing a Workflow with an Error State.

The **Check Performance Duration** functionality may be disabled for a workflow if the **Performance Duration** attribute has not been configured for the workflow's current workflow step in the configuration tool Alfabet Expand. In other words, a deadline can only be checked for the workflow if a deadline has been configured for the current workflow step. Please be aware of the following optional configuration possibilities for a workflow step:

- The **Performance Duration** attribute specifies the number of days that the workflow step may require to be completed.
- The **Performance Reminder Start** attribute specifies on which day during the duration of the workflow step (day 1 = start of workflow step) the first reminder should be generated and sent to the responsible users.
- The **Performance Reminder Frequency** attribute specifies the intervals (number of days) that an email notification should be generated and sent to the responsible users reminding them that the workflow step must be completed.
- The **Expiration Step** attribute specifies which workflow step should automatically be triggered if the deadline is reached and the current workflow step thus expires.

For more information about configuring the workflow step deadline, see *Defining a Deadline and Reminders for a Workflow Step* in the reference manual *Configuring Alfabet with Alfabet Expand*.

A batch process must be configured in order to execute **Check Performance Duration** functionality. For more information, see the section *Batch Processes for Workflows with AlfaWorkflowCommandPrompt.exe* in the reference manual *System Administration*.

Checking a Workflow Step's Post-Conditions and Closing the Workflow Step

In some cases, a user may have performed a workflow step, but forgotten to confirm or complete the workflow step. In this case, the workflow cannot advance to the next workflow step unless the activity is confirmed and any existing post-conditions completed.

In this case, you can initiate the confirm/complete process, check that any existing post-conditions have been fulfilled, and if they have been fulfilled, close the workflow step so that the workflow can advance to the next step. If post-conditions have not been fulfilled, the workflow step cannot be closed. In this case, the user performing the workflow step must complete the required data input in order to fulfill the post-condition.

To initiate the confirm/complete process, select the workflow with the workflow step that needs to be closed and click **Manage** > **Check Post-Conditions and Close Activity**. A message will be displayed with information about whether the action could be executed.

The **Check Post-Conditions and Close Activity** functionality is only available if the **Allow Automatic Closure** attribute is set to **True** in the configuration of the workflow step's post-condition(s) in Alfabet Expand. Furthermore, **Check Post-Conditions and Close Activity** functionality is only relevant for workflow steps in which no automatic confirmation has been configured. For more information, see *Creating a Workflow Step* in the reference manual *Configuring Alfabet with Alfabet Expand*. For more information about configuring post-conditions, see *Configuring Pre- and Post-Conditions for a Workflow Step* in the reference manual *Configuring Alfabet with Alfabet Expand*.

Releasing the Lock on a Workflow

In some cases, a workflow step may be locked if one of the users responsible for the workflow step is currently working with a workflow step. The lock prevents several users working on the same workflow step concurrently. There may be situations in which the lock remains on the workflow step (for example, a user has started a wizard associated with the workflow step, has become distracted, and the session for the user has timed out). In

such situations, you may wish to release the lock to allow interaction with the workflow step. In this case, the lock on the workflow may be removed by the workflow administrator or a user responsible for the workflow step. Once the lock is removed, the workflow can be delegated to a user and activity resumed on the workflow.

To release the lock on a workflow, select a workflow in the table and click the **Release Lock** button in the toolbar.

Deleting a Workflow

You can delete a workflow that has been initiated or started.

If you delete a running workflow, you also delete any workflow step that is currently being performed by a responsible user. In this case, the workflow will be removed from the **My Workflows** functionality of the user owning the workflow. Any associated workflow steps will be automatically deleted from the **My Workflow Activities** functionality of all users responsible for the workflow step.

To delete a workflow:

1) In the table, select the workflow that you want to delete.
2) In the toolbar, click the **Delete** button.
3) Confirm the warning by clicking **Yes** or click **No** to exit without deleting the selected object(s).

Understand the Course of the Workflow

The **Workflow Diagram** page view allows you to view graphic information about the selected workflow. The diagram displays all workflow steps defined for the workflow template and the possible paths the workflow may take depending on the fulfillment of configured pre-conditions or post-conditions. This view is for informational purposes only and cannot be edited.

If a workflow diagram has been configured for the associated workflow template in the configuration tool Alfabet Expand, that diagram will be displayed in this page view. If no workflow diagram has been designed, a default diagram will be automatically generated. The layout of the default diagram cannot be edited. For more information about designing workflow diagrams, see the section *Configuring and Visualizing a Workflow in a DiagramConfiguring and Visualizing a Workflow in a Diagram* in the reference manual *Configuring Alfabet with Alfabet Expand*.

Navigate to the object profile of the relevant workflow and open the **Workflow Diagram** page view. Please keep the following in mind when viewing the workflow diagram:

- A yellow rectangle is the active step that is currently pending. The step may be pending because it is being performed or because a sub-workflow was triggered during the workflow step. Once the sub-workflow is finished and the workflow step is completed, the rectangle will turn green.
- A green rectangle is a completed step.
- A grey rectangle is a step that has been refused or expired.
- A red rectangle is a workflow where an error has occurred.

- A white rectangle is a step that has not yet been performed or will not be performed (for example, for workflow steps on a path that the workflow does not take because a pre-condition has not been satisfied).
- A blue arrow points to the next workflow step in the configured sequence of the workflow.
- A red arrow points to the next workflow step specified in the configuration should a workflow step be refused.
- A green arrow points to the next workflow step specified in the configuration should a workflow step expire.
- Select a workflow step that has been entered (yellow, green, grey, or red rectangles), and click-and-hold to open the preview window to view basic information about the workflow step. It is not possible to open the preview for a workflow step that has not been instantiated.
- Select a workflow step that has been entered (yellow, green, grey, or red rectangles), and click the **Navigate**  button in the toolbar to open the object profile of the workflow step.

Suspending, Resuming, or Withdrawing the Workflow

The **Workflow Event Trace** page view allows workflow owners and workflow administrators to suspend, resume, or withdraw workflows from operation. Furthermore, it provides valuable information about the course of events that have occurred in a workflow. The view displays all events for the selected workflow. You can limit the content in the view by defining filters. In the **Event Type** field, select the type of events that you want to display. In the **Workflow Step** field, select the workflow step(s) that you want to display. Click **Update** to apply the filter criteria to the view. The following information is displayed:

- **Event**: Displays the actions that have occurred to the workflow. Any workflow that is in the state `Error` is in immediate need of attention and must be dealt with by either the workflow owner or workflow administrator. The workflow owner and workflow administrator will typically be informed of the escalated state of the workflow step via email. For detailed description of all potential values displayed in the **Events** column as well as their meaning, see *Understanding Event Values and Resolving Workflow Errors*.
- **Step**: Displays the workflow step number and caption.
- **Step ID**: Displays the identification number of the workflow step.
- **Initiator of Action**: Displays the name of the user triggering the event.
- **Affected Users**: Displays the users that are impacted by the event.
- **Confirmation Type**: Displays who is responsible to confirm the relevant workflow step. The following values may be displayed:
  - `AllUsers`: Indicates that all responsible users must perform/confirm the workflow step in order to complete the workflow step.
  - `SingleUser`: Indicates that only one of the responsible users must perform/confirm the workflow step in order to complete the workflow step.
  - `SingleUserOfEachUserGroup`: Indicates that one responsible user from each responsible user group must perform/confirm the workflow step in order to complete the workflow step.

- **Message**: Displays either system messages or messages entered by responsible users. If the workflow step is escalated because one or more pre-conditions for the successor step have not been met, you will see the message text configured for the pre-condition(s), which should indicate what is required to fulfill the pre-condition.
- **Time Stamp**: Displays the timestamp indicating when the event occurred.

To suspend and resume the workflow:

- To suspend the workflow, select the current workflow event and click the **Suspend**  button in the toolbar. Depending on the configuration of the workflow step, email notifications may be sent to all users responsible for the selected workflow step informing them that the workflow has been paused. The current workflow step will no longer be displayed in the **My Workflow Activities** view of the responsible users. The workflow will have the state Suspended. To continue the workflow step, you must select the **Resume**  button.
- To continue the workflow, select the current workflow event and click the **Resume**  button in the toolbar. Depending on the configuration of the workflow step, email notifications may be sent to all users responsible for the selected workflow step informing them that the workflow has been resumed. The current workflow step will be displayed in the **My Workflow Activities** view of the responsible users. The workflow step will have the state Running.
- To withdraw the workflow, select the current workflow event and click the **Withdraw**  button in the toolbar. The workflow has been withdrawn from the user community and any current workflow steps will no longer be displayed in the **My Workflow Activities** view of the responsible users.

Understanding Event Values and Resolving Workflow Errors

The following values may appear in the **Event** column:

Event Column Value	Meaning and Possible Resolution
`Error`	The workflow step encountered an error during its execution. The value `Error` indicates that an error has occurred in the execution of the workflow. This could be caused by the following situations: • No responsible users have been found for the current workflow step. In this case, the workflow owner can add responsible users to the workflow step. This is carried out in the **Active Step Activities** page view. For more information, see the section *Fixing a Workflow with an Error State*. • A responsible user has refused a workflow step and no next step has been configured for this situation. This error must be corrected in the configuration of the workflow template. Please contact your workflow designer. • A workflow step has expired and no next step has been configured for this situation. This error must be corrected in the configuration of the workflow template. Please contact your workflow designer.

Event Column Value	Meaning and Possible Resolution
	• Multiple workflow steps qualify as a next step based on the pre-conditions defined for the potential next steps. This error must be corrected in the configuration of the workflow template. Please contact your workflow designer. • A workflow step cannot be performed because a required wizard is absent or a required Alfabet query cannot be executed. This error must be corrected in the configuration of the workflow template. Please contact your workflow designer. **NOTE**: For more information, contact your workflow designer or see the chapter *Configuring Workflows* in the reference manual *Configuring Alfabet with Alfabet Expand*.
`Resolved_StepResponsiblesNotFound`	If the workflow step is in an `Error` state because no responsible users were found for the current workflow step and the workflow owner has added responsible users to the workflow step via the **Active Step Activities** page view, this event type will be displayed displaying the user who resolved the error and the users assigned responsibility for the workflow step.
`StepCancelled`	The workflow step has been cancelled. This could be the result of redirecting the workflow to another workflow step via the **Go To** button.
`StepConfirmed`	The workflow step has been confirmed by all responsible users required to confirm the workflow step.
`StepDelegated`	The workflow step has been delegated by one responsible user to another responsible user. Text in the **Comments** column provides an explanation about the delegation.
`StepEntered`	The workflow step has been started.
`StepExited`	The workflow step has been exited. All responsible users required to confirm have confirmed the workflow step.
`StepExpired`	The configured amount of time to complete the workflow step has been surpassed and the workflow step is now expired. Depending on the configuration, the workflow step may be automatically redirected to another workflow step. If this is not the case, the workflow owner can redirect the workflow to another workflow step. This is carried out in the **Active Step Activities** page view. For more information, see the section *Fixing a Workflow with an Error State*. **NOTE**: For more information about configuring the amount of time to complete a workflow step, see the section *Defining a Deadline and Reminders for a Workflow Step* in the reference manual *Configuring Alfabet with Alfabet Expand*.

Event Column Value	Meaning and Possible Resolution
	NOTE: Depending on the configuration of the workflow step, the step may automatically advance to another workflow step if expired. For more information about configuring a workflow step to follow an expired workflow step, see the section *Defining the Sequence of the Workflow Steps* in the reference manual *Configuring Alfabet with Alfabet Expand*.
`StepPerformed`	The workflow step is being performed by a responsible user and is therefore locked and cannot be accessed.
`StepRefused`	The workflow step has been refused by a responsible user. All other responsible users are no longer required to confirm the workflow step. Depending on the configuration, the workflow step may be redirected to another workflow step. **NOTE**: Depending on the configuration of the workflow step, the step may automatically advance to another workflow step if refused. In this case, the `StepEntered` value will be displayed in the **Event** column and will indicate the workflow step that has been configured to follow in case of refusal. For more information about configuring the refusal of a workflow step, see the section *Defining the Sequence of the Workflow Steps* in the reference manual *Configuring Alfabet with Alfabet Expand*.
`WorkflowFinished`	The workflow has completed the last workflow step. There are no next workflow steps.
`WorkflowStarted`	The workflow has started and is in process.
`WorkflowSuspended`	The workflow administrator or workflow owner has paused the execution of the workflow.
`WorkflowResumed`	The workflow administrator or workflow owner has resumed the execution of the workflow.
`WorkflowTrigger`	One or more sub-workflows have been triggered by the current workflow step. The superordinate workflow can proceed to the next workflow step and will not wait for the completion of the sub-workflows. The sub-workflow(s) are triggered and executed independent of the parent workflow triggering them.
`WorkflowTriggeredAndWait`	One or more sub-workflows have been triggered by the current workflow step. The superordinate workflow will resume activity as soon as the first of the sub-workflows has finished. The completion of the first sub-workflow will be used to determine how to proceed in the superordinate workflow.
`WorkflowUpdated`	The workflow template has been updated in the context of a migration to a new workflow template. In the case of a running workflow, the current workflow step based on the old workflow template will be cancelled and automatically updated

Event Column Value	Meaning and Possible Resolution
	according to the new workflow template. The state defined for the workflow prior to the migration will be maintained. This is particularly relevant for workflows with the state `WorkflowSuspended`. **NOTE**: For more information about the configuration of a workflow template migration, see the section *Creating a Migration Definition to Update Running Workflows* in the reference manual *Configuring Alfabet with Alfabet Expand*.
`WorkflowWithdrawn`	The workflow step has been withdrawn and is no longer a current workflow step. Typically, the workflow will be configured so that the workflow will advance to another configured workflow step.

Fixing a Workflow with an Error State

The **Active Workflow Steps** page view allows workflow owners and workflow administrators to delegate a workflow step to another user, add new responsible users to the workflow steps, or redirect the workflow to a different workflow step.

Any workflow that is in the state `Error` is in immediate need of attention and must be dealt with by either the workflow owner or workflow administrator. The workflow owner and workflow administrator will typically be informed of the escalated state of the workflow step via email. In this case, the workflow owner or workflow administrator should first review the error displayed in the **Event Type** column in the **Workflow Event Trace** page view to understand the type of error that has occurred.

The event type `Error` could be caused by a number of situations. Typically, errors result due to problems in the configuration of the workflow template. For example, an error could occur because a responsible user has refused a workflow step and no next step has been configured for this situation, multiple workflow steps qualify as a next step based on the pre-conditions defined for the potential next steps, or a workflow step cannot be performed because a required wizard is absent, or a required query cannot be executed. These are errors that typically must be resolved in the configuration of the workflow. In this case, you must contact your workflow designer so that the configuration of the workflow template is corrected.

In some cases, the event type `Error` could be caused because no responsible users have been found for the current workflow step. The event type `StepExpired` indicates that the time configured to complete the workflow step has been surpassed. In this case, the workflow owner can redirect the workflow to another workflow step. Both errors can be corrected as described below by the workflow owner or workflow administrator in the **Active Step Activities** page view.

For more information about working with the **Workflow Event Trace** page view and understanding the event type values, see the sections Suspending, Resuming, or Withdrawing the Workflow and Understanding Event Values and Resolving Workflow Errors.

The following information is available:

- Redirecting a Workflow Step to Another Step
- Adding Responsible Users to a Workflow Step
- Delegating a Workflow Step to Another User

Redirecting a Workflow Step to Another Step

Any workflow that displays a state with the value `Error` is in immediate need of attention. This indicates that an error has occurred in the execution of the workflow and the workflow must either be rechanneled or, depending on the cause of the error, reconfigured.

To rechannel the workflow to a different workflow step:

1) Navigate to the object profile of the relevant workflow and open the **Active Workflow Steps** page view.
2) Select the current workflow event and click the **Go To** button in the toolbar.
3) An editor will open that requires you to select the new workflow step and provide an explanation about why you are redirecting the workflow step. In the **Go to Another Workflow Step** editor that opens, define the following:
   - **Select the workflow step...**: Select the workflow step that the workflow should be redirected to.
   - **Execute the post-conditions...**: Set a checkmark if post-conditions configured for the workflow step selected in the **Active Workflow Steps** page view should be checked before the new workflow step is triggered.
   - **Execute the pre-conditions...**: Set a checkmark if pre-conditions configured for the workflow step selected in the **Select the workflow step...** field should be checked.
   - **Comment**: Provide an explanation about why the workflow is being redirected to the selected workflow step. The explanation will be available to the user responsible for the subsequent step in the object profile of the workflow step.
4) Click **OK** to save your changes. The responsible user(s) will typically be notified via email that the they are responsible for the new workflow step.

Adding Responsible Users to a Workflow Step

If the workflow is in a state of `Error` because no users have been identified as responsible for the workflow step, then you must manually assign responsible users to the workflow step to resolve the workflow error. The **Add Responsible Users** button will be enabled if no responsible users have been found via the workflow step configuration. The new responsible user(s) will typically be notified via email that the workflow step has been delegated to him/her.

To add responsible users to a workflow step:

1) Navigate to the object profile of the relevant workflow and open the **Active Workflow Steps** page view.
2) Select the current workflow event and click the **Add Responsible Users** button in the toolbar.
3) The person selector will open. Select the relevant users to add to the workflow step.
4) Click **OK** to save the definition or click **Cancel** to exit the editor without saving your changes. The selected user(s) will typically be notified via email that they have been assigned a workflow step. The event type `Resolved_StepResponsibilesNotFound` will be displayed in the **Event** column in the **Workflow Event Trace** page view.

Delegating a Workflow Step to Another User

A workflow may need to be delegated to another responsible user. You can specify whether the responsibility of the workflow will be removed from only the selected responsible user or all users responsible for the workflow step. The delegated user will typically be notified via email that the workflow step has been delegated to him/her.

To delegate the workflow to another responsible user:

1) Navigate to the object profile of the relevant workflow and open the **Active Workflow Steps** page view.
2) Select the current workflow event and click the **Delegate**  button in the toolbar.
3) An editor will open that requires you to provide an explanation about why you are delegating the workflow step to the selected user. In the **Delegate Workflow Step** editor that opens, define the following:
   - **Substitute All Current Workflow Step Owners**: Select this checkbox if all users currently responsible for the selected workflow step should be replaced by the user(s) that the workflow step is being delegated to.
   - **Revoke Responsibility from Current User**: Select this checkbox if you want to remove yourself as a user responsible for the workflow step that is being delegated. Clear this checkbox if you want to include yourself as a user responsible for the workflow step that is being delegated.
   - **Comment**: Provide an explanation about why the workflow is being delegated to another user. The explanation will be displayed in the **Message** column of the **Workflow Step Event Trace** page view.
   - **Delegates**: Click the **Add Person** button to select one or more persons to delegate the workflow step to.
4) Click **OK** to save the delegation definition or click **Cancel** to exit the editor without saving your changes. The selected user(s) will typically be notified via email that the workflow step has been delegated to him/her.

Chapter 8: Defining Discussion Groups for Collaborative Discussions

The **Discussion Groups** functionality in the **Solution Administration** module allows you to configure and administrate discussion groups for the discussion capability.

A discussion in Alfabet will typically proceed as follows:

- A discussion is initiated in the **Discussion** page view for a selected object by an authorized user, member of an authorized user group, or deputy who is responsible for the object. The **Discussion** page view is only available in the object profile for object classes for which the discussion capability has been configured. The configuration will specify the release status that an object must have in order for a discussion to be initiated for that object as well as which discussion groups may participate in the discussion. Typically, the user initiating the discussion will make the first contribution to the discussion explaining why discussion was initiated. When initiating the discussion, the user must select a discussion group to be the managing discussion group for the discussion.
- Any member of the managing discussion group may choose to invite other discussion groups to participate in the discussion in the **My Discussions** functionality. The managing discussion group may only invite discussion groups that have been configured to be invited to the discussion. Any member of the managing discussion group can include or exclude an invited discussion group at any time. Depending on the configuration of the discussion capability, email notification may be sent to all members of the managing discussion group and invited discussion groups when a discussion is initiated about an object.
- All users with authorized access to the object as well as members of the managing discussion group and invited discussion groups may contribute to and track the ongoing discussion about the object in the **My Discussions** functionality. Discussion participants may add comments, define the status of the discussion, and trigger email notification for other discussion participants information them about the contribution. Depending on the configuration, the members of a participating discussion group may or may not be able to edit the object targeted by the discussion.
- Once it has been determined that the discussion has been completed, an authorized user of the object or a member of the managing discussion group can deactivate and close the discussion. No further contributions can be made to an inactive discussion.

For a detailed description about initiating and participating in a discussion, see the section *Initiating, Contributing to, and Managing a Discussion About Objects* in the reference manual *Getting Started with Alfabet*.

The **Discussion Groups** functionality allows you to create discussion groups, assign users to a discussion group, specify the object classes that will be the focus of each discussion group, define the access permissions of discussion group members to the objects under discussion, configure whether automatic email notifications should be sent in the context of a discussion, and identify other discussion groups that may potentially be invited to participate in a discussion that is managed by a selected discussion group.

- Click the **Discussion Group Manager** icon  at the top of the explorer to see all existing discussion groups in the **Root Discussion Groups** view. Each column in the table is defined below:
  - **ID**: Displays the identification number of the discussion group.
  - **Name**: The discussion group's name.
  - **Discussion Group Type**: Displays the discussion group type that has been assigned to the discussion group.

- In the **Discussion Group Manager** explorer:
  - Click a discussion group at the top level of the discussion group hierarchy to define the members, its subordinate class-based discussion groups, and manageable groups for the selected discussion group.
  - Click the symbol next to discussion group to see its subordinate class-based discussion groups.

The following information is available:

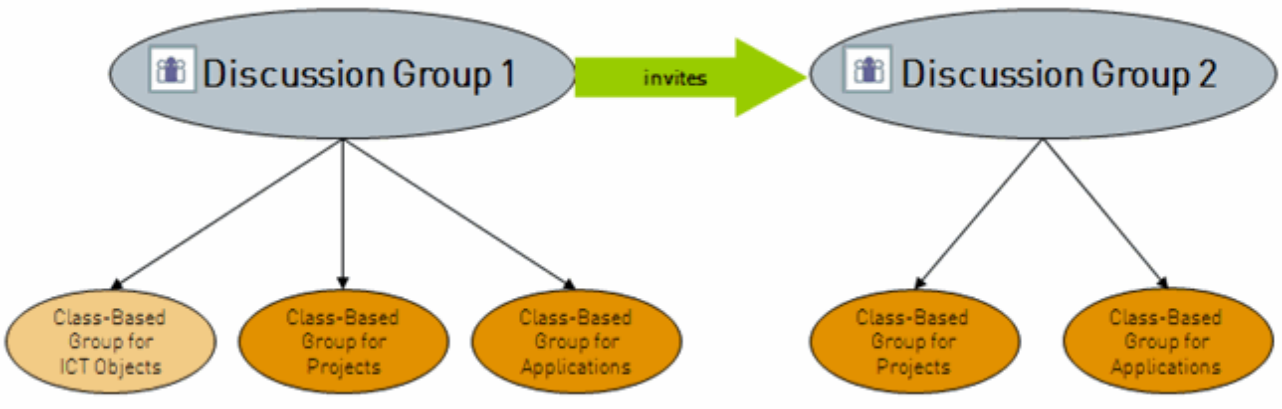
- Configuring Discussion Groups and the Discussion Functionality
- Creating a Parent Discussion Group
- Defining the Members of a Discussion Group
- Specifying an Object Class for the Discussion Group
- Defining Discussion Groups to Invite to a Discussion

Configuring Discussion Groups and the Discussion Functionality

A discussion group is a preconfigured group of users that have been defined permission to discuss objects in specified object classes.

A discussion group must have at least one subordinate class-based discussion group defined, which specifies the object class that is the focus of discussion, the discussion statuses to be implemented for the discussion, whether members of the discussion group have been granted Read/Write or ReadOnly access permission to the objects being discussed, and whether automatic email notifications should be sent in the context of discussion activity.

Other discussion groups may be invited to join a discussion that has been initiated by the discussion group managing the discussion. In this case, the discussion group that is to be invited must be included in the discussion group's set of manageable discussion groups and must have a relevant class-based discussion group defined for it. For example, if the managing discussion group is discussing an application and invites another discussion group to provide input to the discussion, the invited discussion group must also have a class-based discussion group defined for the object class Application.



In the figure above, Discussion Group 1 may engage in discussions about ICT objects, projects, and applications. Discussion Group 2 may engage in discussions about projects and applications. If Discussion Group 2 has been

defined as a manageable group for Discussion Group 1, then Discussion Group 1 may invite Discussion Group 2 to participate in their discussions about applications and projects. Discussion Group 2, however, will not be able to participate in any discussions about ICT objects.

The following steps must be carried out in the **Discussion Groups** functionality in the **Solution Administration** module in order to implement the discussion functionality in Alfabet:

- Create a discussion group. When you create a discussion group, you define activity states (typically Active and Inactive) which control whether a discussion group may participate in the discussion. You must also specify whether emails should be automatically sent to the discussion group members when discussion is initiated for an object, and whether the members of the discussion group will have Read/Write or ReadOnly access permissions to the objects targeted by the discussion. For more information, see the section Creating a Parent Discussion Group.
- Specify the users that constitute the parent discussion group. For more information, see the section Defining the Members of a Discussion Group.
- Create one or more subordinate class-based discussion groups for the parent discussion group. A parent discussion group must have at least one subordinate class-based discussion group defined for it. The class-based discussion group determines the object class that is the target of discussion. A class-based discussion group should be defined for each object class that the discussion group may discuss. The settings specified for the parent discussion group serve as a template and can be reused or modified for its subordinate class-based discussion groups. For more information, see the section Specifying an Object Class for the Discussion Group.
- Create other discussion groups as described above that may be invited to a discussion. You must ensure that the relevant subordinate class-based discussion groups are also defined for the discussion groups that may be invited to the discussion. For more information, see Defining Discussion Groups to Invite to a Discussion.
- Ensure that the required configuration has been carried out if emails are to be automatically sent to members of a discussion group when a discussion is initiated. You must ensure that the following configuration is complete in order for the emails to be sent:
  - Each user assigned to the relevant discussion group(s) must have an email defined by the user administrator. For more information, see the chapter Defining and Managing Users.
  - The **Send Activation Mail** checkbox must be selected in the **Discussion Group** editor for the parent discussion group as well as each class-based discussion group.
  - The text template used for the email must be configured in the configuration tool Alfabet Expand. For more information, see the section *Configuring Text Templates for Email Notifications* in the reference manual *Configuring Alfabet with Alfabet Expand*. The configured text template must be selected in the **Activation Template** field in the **Discussion Group** editor. For more information, see Creating a Parent Discussion Group.
  - Specify the user profile used to open Alfabet via the email notification in the **User Profile to Access Object in Notification** field in the **Discussion Group** editor. For more information, see Creating a Parent Discussion Group..
  - The email capability requires the setup of a connection to an SMTP server for outgoing emails. For more information, see the section *Activating the Dispatch of Email Notifications in Alfabet* in the reference manual *System Administration*.
  - An arbitrary member of the discussion group will be entered in the email's **To:** field and all other discussion members will be entered in the email's **CC:** field. The sender of the email will typically be the initiator of the discussion group unless otherwise configured in the server alias

configuration. For more information, see the section *Specifying Sender Email Addresses* in the reference manual *System Administration*.

- Ensure that the discussion group types (for example, Steering Committee, Round Table, etc.) that are relevant to your enterprise are configured in the configuration tool Alfabet Expand. The discussion group types specified in the `DiscussionGroupType` enumeration are displayed in the **Discussion Group Type** field of the **Discussion Group** editor. For more information, see the section *Defining Protected and Custom Enumerations* in the reference manual *Configuring Alfabet with Alfabet Expand*.

Creating a Parent Discussion Group

An unlimited number of parent discussion groups can be created for your enterprise. For each parent discussion group you define, you must define a subordinate class-based discussion group for each object class that may be the focus of discussion by the parent discussion group. The attributes that you specify in the **Settings** tab in the **Discussion Group** editor will be inherited by the subordinate class-based discussion groups but can later be modified for each class-based discussion group, as needed.

It is possible to invite other discussion groups to a discussion. The discussion groups you plan to invite must be defined as described here. A class-based discussion group should be defined for each object class that the discussion group may discuss. The discussion groups that you want to potentially invite to a discussion must be assigned as a manageable discussion group to the discussion group that is doing the inviting. For more information, see the sections Specifying an Object Class for the Discussion Group and Defining Discussion Groups to Invite to a Discussion.

1) Click the **Discussion Group Manager** icon  at the top of the explorer tree.
2) In the toolbar, click **New** > **Create Discussion Group**. The **Discussion Group** editor opens.
3) Enter information in the fields, as needed.

   Properties tab:

   - **ID**: Alfabet assigns a unique identification number to each object in the inventory. This number cannot be edited.
   - **Name**: Enter a unique name for the discussion group.
   - **Discussion Group Type**: Select a discussion group type in the drop-down list. Values (such as Round-Table, Approval, etc) available for the **Type** attribute are configured by your company in the configuration tool Alfabet Expand. For more information, see the section *Defining Protected and Custom Enumerations* in the reference manual *Configuring Alfabet with Alfabet Expand*.
   - **Description**: Enter a meaningful description that will clarify the purpose of the discussion group.

   Settings tab:

   - **Discussion Statuses**: Enter the statuses that members of the discussion group may select to describe the stage of the discussion at a given time (for example, Under Review, Approved, Rejected).
   - **Grant Object Rights**: Select the checkbox if the members of this discussion group should have permission to edit the object that is the focus of the discussion. Clear the checkbox if the members of the selected discussion group should not be able to edit the object. The access permissions granted to users via the discussion group are independent of the accessibility granted via an authorized user, authorized user group, or deputy definition. However, access permissions specified

via mandates are applied. If your enterprise implements a federated architecture, you must ensure that the members of the discussion group will be able to view the objects targeted by the discussions. For detailed information about access permissions in Alfabet, see the section *Understanding Access Permissions in Alfabet* in the reference manual *Getting Started with Alfabet*.

- **Send Activation Mail**: Select the checkbox if email notifications should be sent to the members of the discussion group when discussion for a specific object is initiated.
- **Activation Template**: Select the text template that should be used for the email notifications that are generated when discussion is initiated for an object. If you do not select a text template, email notifications will not be generated even if a checkmark has been selected for the **Send Activation Mail** field. An arbitrary member of the discussion group will be entered in the email's **To:** field and all other discussion members will be entered in the email's **CC:** field. The sender of the email will typically be the initiator of the discussion group unless otherwise configured in the server alias configuration.

  Text templates can be configured by your solution designer in the configuration tool Alfabet Expand. For more information, see the section *Configuring Text Templates for Email Notifications* in the reference manual *Configuring Alfabet with Alfabet Expand*.

- **Discussion Activity States**: Optional. Discussion activity states control whether a discussion group may participate in the discussion. They are displayed in the **Discussion Items** editor that allows a permissible user to manage the discussion and invite a discussion group to join the discussion. If no values are entered here, the default values **Active** and **Inactive** will be automatically displayed in the **Discussion Items** editor. If necessary, enter all alternative discussion activity states that users can select for discussions. The value you enter in the **Inactive States** field must be included in the **Discussion Activity States** field.
- **Inactive States**: Optional. If no values are entered here, the default value **Inactive** will automatically be used in the **Discussion Items** editor to exclude a discussion group from a discussion and close a discussion. IF necessary, enter all alternative states that qualify as inactive states for the discussion. A contribution cannot be made by a discussion group that has an inactive activity state.

4) Click **OK** to save your changes or click **Cancel** to exit without saving. The discussion group is displayed at the top level of the explorer hierarchy.

Defining the Members of a Discussion Group

The **Members** page view allows you to define the users that belong to the selected discussion group. All users you assign to the discussion group will be able to access and participate in discussions about objects in the object classes specified in all subordinate class-based groups defined for the selected discussion group. The access permissions granted to users via the discussion group are independent of the accessibility granted via an authorized user, authorized user group, or deputy definition. However, accessibility specified via mandates are applied. If your enterprise implements a federated architecture, you must ensure that the members of the discussion group will be able to view the objects targeted by the discussions.

The table displays existing users that have already been assigned to the selected discussion group. Each column in the table is defined below:

- **Name**: The family name of the user.
- **First Name**: The first name of the user.
- **Email**: The email address of the user. Emails may be automatically generated and sent to the members of a discussion group when discussion is initiated about an object. The automatic generation of emails

for members of a discussion group is defined in the **Class-Based Discussion Groups** page view. For more information, see the section Specifying an Object Class for the Discussion Group.

To define the members of the selected discussion group:

1) In the **Discussion Group Manager** explorer tree, click the parent discussion group and click **Members** to open the view.
2) In the toolbar, click **New** > **Add Existing Persons**.
3) The object selector opens. Enter search criteria, as needed, and click **Search**.
4) Select the user(s) that you want to add to the discussion group and click **OK** to save your changes or click **Cancel** to exit without saving. All members assigned to the parent discussion group will be able to participate in all subordinate class-based discussion groups.

The access permissions for the objects targeted by the discussion group can be defined for the members of a discussion group on the level of the parent discussion group or the class-based discussion group.

Specifying an Object Class for the Discussion Group

The **Class-Based Discussion Groups** page view allows you to define which object classes may be the focus of discussion by the selected discussion group. A discussion group must have at least one class-based discussion group defined for it. The original settings defined for the ascendant discussion group are copied to the class-based discussion group and can be modified for each class-based discussion group, as needed.

For more information about the object classes that can be configured for class-based discussion groups, see *Overview of Configurable Features for Object Classes* in the reference manual *Configuring Alfabet with Alfabet Expand - Appendix*.

Once a class-based discussion group has been defined, the **Discussion** page view will be displayed in the object profiles for the object classes specified in the class-based discussion groups. A discussion can be initiated for an object in the **Discussion** page view by a user with relevant access permissions, such as authorized user, authorized user group, or deputy.

If you are defining a discussion group that may be invited to join a discussion by another discussion group, you must ensure that the invited discussion group also has a relevant class-based discussion group defined. For example, if a discussion group is discussing an application, the invited discussion group must also have a class-based discussion group defined for the class Application. For more information, see the section Defining Discussion Groups to Invite to a Discussion.

To define an object class that may be the focus of the selected discussion group:

1) In the **Discussion Group Manager** explorer tree, click the parent discussion group and click **Class-Based Discussion Groups** to open the view.
2) In the toolbar, click **New** > **Define Object Class for Discussion**. The **Discussion Group** editor opens. Some of the settings inherited from the parent discussion group are greyed out and cannot be edited.
3) In the **Class Name** field, select the object class that should be the focus of discussion by this discussion group. If you are defining a discussion for objects based on an object class stereotype, be sure to select the correct stereotype.

4) Depending on the object class you select, a set of release statuses may be displayed in the **Permissible Object Statuses for Discussion Activation** field and **Permissible Object Statuses for Discussion Contribution** field. If release statuses are displayed:

   - In the **Permissible Object Statuses for Discussion Activation** field, select one or more release statuses that the object must have in order for a discussion to be initiated for that object.

     If release statuses are displayed in the **Permissible Object Statuses for Discussion Activation** field and you do not select a status, then NO discussions can be initiated for any objects in the selected object class.

   - In the **Permissible Object Statuses for Discussion Contribution** field, select one or more release statuses that the object must have in order for a contribution to be made by discussion participants.

5) In the **User Profile to Access Object in Notification**, select the user profile that shall be used by email recipients to access the discussion via the hyperlink in the notification email.

6) Click the **Settings** tab and review the definitions that have been copied from the selected discussion group. Modify any of the settings, as needed. In particular, you should ensure that the **Grant Object Rights** field is correctly defined for the object class. Please note that if email notifications are to be automatically sent in the context of this class-based discussion group, the **Send Activation Mail** checkbox must be selected for both the parent discussion group as well as the class-based discussion group.

7) Click **OK** to save your changes or click **Cancel** to exit without saving.

Defining Discussion Groups to Invite to a Discussion

The **Manageable Groups** page view allows you to define which other discussion groups may potentially be invited to participate in a discussion that is managed by the selected discussion group. All existing discussion groups that may potentially be invited must be added to the set of manageable groups. The discussion groups that are added to the set of manageable groups must also have a class-based discussion group defined for the relevant object class that is the focus of the discussion. For example, if the selected discussion group is discussing an application, the manageable group must also have a class-based discussion group defined for the class Application. For more information about defining a class-based discussion group for a discussion group, see the section Specifying an Object Class for the Discussion Group.

Once the discussion is initiated, all discussion groups that may participate in the discussion are displayed in the **Discussion** page view in the relevant object profile and the **My Discussions** functionality. The discussion group managing the discussion can then invite a discussion group to join the discussion in the **My Discussions** functionality. Members of the invited discussion group will be able to access the discussion in the **My Discussions** functionality.

To add an existing discussion group to the set of manageable groups for the selected discussion group:

1) In the **Discussion Group Manager** explorer tree, click the parent discussion group  and click **Manageable Groups** to open the view.

2) In the toolbar, click **New** > **Add Existing Discussion Group**.

3) The object selector opens. Enter search criteria, as needed, and click **Search**.

4) Select the relevant discussion group that you want to add and click **OK** to save your changes or click **Cancel** to exit without saving.

Chapter 9: Uploading Documents and Managing User Permissions to Document Folders in the Internal Document Selector

The **Internal Documents** functionality allows you to administrate access permissions to the document folders available in the **Internal Document Selector**. The **Internal Document Selector** is a central repository in Alfabet that allows documents to be uploaded for the following purpose:

- Files that users attach to objects in the *Attachments Page View*. A document attached to an object can also be downloaded in the **Attachments** page view by users with permission to the object.
- CSS files that will be referenced as stylesheets in the HTML configurations for workflows, wizard steps, must be stored in a document folder that is subordinate to the root folder of the **IDOC** explorer. Please note that the stylesheet may not be located in the root folder of the **IDOC** explorer in the **Internal Document Selector**.

Please note the following:

- The `SYSTEM` folder is reserved for system documents related to specific functionalities in Alfabet. You should not make any manual changes to the `SYSTEM` folder unless explicitly advised.
- Style sheets referenced in the context of object cockpits, wizards steps, HTML templates, etc. may not be stored in the root folder of the **IDOC** explorer. Stylesheets stored in the **Internal Document Selector** must be located in a document folder that is subordinate to the root folder of the **IDOC** explorer.
- For security reasons, a blacklist and whitelist concept has been introduced to restrict the uploading and downloading of files with permissible file extensions in Alfabet. An error message will be displayed f a user attempts to upload or download an impermissible file extension type. For more information about restricting file extensions, see the section *Specifying the Permissible File Extensions for Uploading/Downloading Files* in the reference manual *Configuring Alfabet with Alfabet Expand*.
- Please note that the following file types cannot be opened directly in Alfabet and will be automatically saved and downloaded as a ZIP file even if not defined in the blacklist or additionally defined in the whitelist:.com,.bat,.exe,.ad,.adprototype,.asax,.ascx,.ashx,.asmx,.asp,.aspx,.axd,.browser,.cd,.compiled,.config,.cs,.csproj,.dd,.exclude,.java,.jsl,.ldb,.ldd,.lddprototype,.ldf,.licx,.master,.mdb,.mdf,.msgx,.refresh,.rem,.resources,.resx,.sd,.sdm,.sdmDocument,.sitemap,.skin,.soap,.svc,.vb,.vbproj,.vjsproj,.vsdisco, and.webinfo. A GUID extension will be added to the file name of the ZIP file.

The documents can be organized in document folders. When you open the **Internal Documents** functionality, you will see the document folders that have been created in the **Internal Documents** functionality or in the **Internal Document Selector** which can be accessed via the *Attachments Page View*. Click the **IDOC** node to display all document folders in the table. Click a document folder in the explorer to view the sub-folders and documents stored in it. The following information is displayed:

- **Type**: Displays whether the object is a document folder or document.
- **Name**: The document folder's name.
- **Size**: Displays the size in KB of the document folder/document.
- **Creation Date**: Displays the date that the document folder was created or the document was uploaded.

- **Change Date**: Displays the date that the document folder/document was changed.
- **Expiration Date**: Displays an expiration date for publications generated via the Alfabet Publication Framework (APF). For more information, see the chapter *Publishing Data In Microsoft® Word Format* in the reference manual *Configuring Alfabet with Alfabet Expand*.
- **Category**: The category that the document is assigned to. Document categories are configured in the configuration tool Alfabet Expand. For more information, see the section *Defining Protected and Custom Enumerations* in the reference manual *Configuring Alfabet with Alfabet Expand*.

The following information is available:

- Creating a New Document Folder and Specifying Access Permissions
- Uploading Documents to the Internal Document Selector
- Downloading Documents to a Local Drive

Creating a New Document Folder and Specifying Access Permissions

New document folders and sub-folders can be created in the **Internal Documents** functionality in order to structure and organize documents. A folder can have multiple levels of sub-folders.

Typically, authorized users and user groups will be specified to have permission to modify existing document folders.

If the document folder is specified as protected, then only authorized users and user groups with the specified permissions will have authorization to rename and delete the document folder, add sub-folders to the document folder, and upload or delete files to/from the document folder. If the document folder is not specified as protected, all users may potentially have access permissions to the document folder.

Please note that the access permissions defined are only relevant to restrict access to documents and document folders from within the **Internal Document Selector**. Documents attached to an object via the *Attachments Page View* are NOT controlled by these permission settings. Hence, documents in the *Attachments Page View* can be accessed and viewed by anyone with access permissions to the relevant object view, regardless of the permissibility setting for the document folder that they are stored in.

Authorized users and user groups as well as the protection of document folders are defined in the **Internal Documents** functionality available via an administrative user profile. For more information, see the section Uploading Documents and Managing User Permissions to Document Folders in the Internal Document Selector in the reference manual *User and Solution Administration*.

To create a new document folder:

1) Open the **Internal Documents** functionality.
2) To create a new document folder at the root level of the document folder hierarchy, click the root node of the **IDOC** explorer. To create a new document folder that is subordinate to an existing document folder, click the document folder in the **IDOC** explorer.
3) In the toolbar, click **New** > **Create New Folder**. The **Document Folder** editor opens.
4) Define the following fields, as needed.
   - **Name**: Displays the name of the document folder.

- **Protected**: Select the checkbox to restrict access to the document folder to users logged in with an administrative user profile. If you do not select the checkbox, all users may potentially access the document folder.

  Please note that if the **Protected** attribute is selected, it may be overridden if any of the attributes listed in the **Default Access Permissions** pane are selected. If default access permissions are defined to allow access to the folders, all users will be able to access/delete/edit folders even if the **Protected** checkbox is selected.

- **Default Access Permissions**: If a user is logged in to the **Internal Document Selector** with a user profile for which the **Is Administrative User Profile** attribute is set to `True`, the access permissions defined for the document files and folders will be ignored.
  - **Visible**: Select the checkbox if the document folder should be visible per default to potentially all users.
  - **List Items**: Select the checkbox if all files stored in the document folder should be visible per default to potentially all users.
  - **Modify**: Select the checkbox if the document folder's properties may be edited per default by potentially all users.
  - **Manage Items**: Select the checkbox if files may be added to or deleted from the document folder per default to potentially all users.
  - **Open Items**: Select the checkbox if the files in the document folder may be opened and read per default to potentially all users.
  - **Edit Items Content**: Select the checkbox if the content of the files in the document folder may be edited per default to potentially all users

5) Enter a name for the document folder and click **OK**. The document folder is available in the **Internal Documents** functionality.

Uploading Documents to the Internal Document Selector

Documents can be uploaded to the **Internal Document Selector** so that users can attach the documents to objects in the *Attachments Page View*. A document attached to an object can also be downloaded in the **Attachments** page view by users with permission to the object.

Please note the following:

- The `SYSTEM` folder is reserved for system documents related to specific functionalities in Alfabet. You should not make any manual changes to the `SYSTEM` folder unless explicitly advised.
- Style sheets referenced in the context of object cockpits, wizards steps, HTML templates, etc. may not be stored in the root folder of the **IDOC** explorer. Stylesheets stored in the **Internal Document Selector** must be located in a document folder that is subordinate to the root folder of the **IDOC** explorer.
- For security reasons, a blacklist and whitelist concept has been introduced to restrict the uploading and downloading of files with permissible file extensions in Alfabet. An error message will be displayed f a user attempts to upload or download an impermissible file extension type.

For more information about restricting file extensions, see the section *Specifying the Permissible File Extensions for Uploading/Downloading Files* in the reference manual *Configuring Alfabet with Alfabet Expand*.

- Please note that the following file types cannot be opened directly in Alfabet and will be automatically saved and downloaded as a ZIP file even if not defined in the blacklist or additionally defined in the whitelist:.com,.bat,.exe,.ad,.adprototype,.asax,.ascx,.ashx,.asmx,.asp,.aspx,.axd,.browser,.cd,.compiled,.config,.cs,.csproj,.dd,.exclude,.java,.jsl,.ldb,.ldd,.lddprototype,.ldf,.licx,.master,.mdb,.mdf,.msgx,.refresh,.rem,.resources,.resx,.sd,.sdm,.sdmDocument,.sitemap,.skin,.soap,.svc,.vb,.vbproj,.vjsproj,.vsdisco, and.webinfo. A GUID extension will be added to the file name of the ZIP file.

Depending on the configuration of your enterprise's solution, documents may be available to the user community via an external file system or via the **Internal Document Selector**. For more information about configuring the availability of documents, see the section *Making Documents and Files Available to the Alfabet User Community* in the reference manual *System Administration*.

To upload a file to the **Internal Document Selector**:

1) Open the **Internal Documents** functionality.
2) In the **IDOC** explorer, select the document folder that you want to store the documents in.
3) In the toolbar, click **New** > **Upload Documents**. The **Upload files into IDOC** window opens.

If you upload a file that has the same file name as a file that already exists in the **Internal Document Selector** functionality, the new version of the file will replace the original version in the **Internal Document Selector** functionality and the **Attachments** page view.

4) Click the **Choose Files** button to select the files you want to add and click **Upload**. The document is displayed in the **Internal Documents** functionality.
5) To assign the document to a document category, select the document in the table and click the **Edit**  button.
6) In the dialog, select a document category in the **Category** field and click **OK** to close the editor. Document categories are configured in the configuration tool Alfabet Expand. For more information, see the section *Defining Protected and Custom Enumerations* in the reference manual *Configuring Alfabet with Alfabet Expand*.

Downloading Documents to a Local Drive

Any file that has been uploaded to the **Internal Document Selector** can be downloaded to the local hard disk and opened manually by means of the relevant locally-installed program. To download a file, select the file in the table and click the **Download Document**  button. In the dialog that opens, click the **Download** button and save the file to the local hard disk. If the **Internal Document Selector** contains only a single folder and a file is downloaded, a GUID is automatically added to the file name to prevent files of the same name being generated.

Chapter 10: Deleting and Archiving Alfabet Objects

The **Archive Manager** functionality allows you to delete Alfabet objects from the Alfabet database and create an archive object that can be downloaded and stored locally.

An archive object is a snapshot of a deleted Alfabet object. When an Alfabet object is archived, a ZIP file is created containing HTML files that display the object profile for the archived object as well as the object profiles of its dependent objects. Each archived object profile displays a preconfigured set of page views, whereby the visibility of these views will depend on the class setting configured for the object class. If a page view displays dependent objects, a user can click the dependent object in the HTML view in order to open another HTML file showing the archived object profile of the selected dependent object.

Alfabet objects are typically archived by a solution administrator. The ZIP file may be downloaded to a local disk and extracted. The relevant HTML file can then be viewed in a browser window. An archive object will be created for each culture setting supported by your enterprise.

This functionality is especially useful to remove outdated objects from the Alfabet database and thus enhance the performance of Alfabet. The archived data can be saved and displayed when needed in a Web browser.

The following information is available:

- Understanding the Consequences of Deletion
- Deleting an Alfabet Object and Creating an Archive Object
- Downloading and Storing an Archived Object
- Accessing the Archived Information for an Archive Object

Understanding the Consequences of Deletion

Any Alfabet object you archive will be deleted from the Alfabet database regardless of its current object state! All relevant dependent objects will simultaneously be deleted! There is no Undo action. Deleted objects cannot be restored!

Please note that it is possible to delete and archive active objects as well as retired objects. Therefore, you must ensure that the object that you plan to archive is also relevant for deletion from the Alfabet database. The object class property **Integrity Info** of object classes specifies which object classes are dependent object classes. For general information about the structure of integrity reference definitions and information about the individual configuration for each relevant object class, see the reference manual *Alfabet Meta-Model*.

Any Alfabet object may be deleted by users with relevant access permissions except for objects that have a non-editable release status. Whenever an object is deleted, a dialog box will be displayed to the user asking him/her to confirm that the object and its dependent objects will be deleted.

The consequences of deletion are the same for an object that is deleted by its authorized user and an object that is deleted and archived by an administrator in the **Administration** application. Any object that is deleted will be irrevocably deleted from the Alfabet database. If the object has dependent objects, they will also be deleted from the Alfabet database. For example:

- If you delete an application, it will be irrevocably deleted from the Alfabet database.

- If the deleted application has application variants, the application variants will become application versions. In this case, the value `True` will be removed from the **Is Variant** attribute.
- If any of the following dependent objects or data is defined for the deleted application, these will also be deleted: Enterprise Release Item, Business Service, Business Data Usage, Information Flow, Local Component, Stack, Migration Rule, Operational Business Support, Tactical Business Support, Solution Business Support, Solution Information Flow, Application Diagram, Platform Diagram, Assignment, Attachment, Deputy, Evaluation, Lifecycle, Role.

Depending on the object that is being deleted, there may also be consequences to objects that are subordinate to the deleted object. The context-sensitive online Help provides detailed information about the consequences of the **Delete** action for a specific object in question. For example:

- If you delete an ICT object category, it will be irrevocably deleted from the Alfabet database.
- If you delete an ICT object category that has ICT objects assigned to it, those ICT objects will remain in the database but will no longer be assigned to an ICT object category. To view all ICT objects that are not assigned to a category and reassign them to another category, see the *Unassigned ICT Objects Page View*.
- Only an ICT object category with no subordinate ICT categories can be deleted. Therefore, you may need to first delete all sub-categories from the selected ICT object category or reassign them to another ICT object category. To reassign an ICT object category to another ICT object category, see the *Sub-Categories Page View* of the category that is to be the new ascendant (parent) ICT object category.
- If any of the following dependent objects or data is defined for the deleted object, these will also be deleted: Assignment, Attachment, Deputy, Role.

Deleting an Alfabet Object and Creating an Archive Object

Any Alfabet object you archive will be deleted from the Alfabet database regardless of its current object state! All relevant dependent objects will simultaneously be deleted! There is no Undo action. Deleted objects cannot be restored!

Please note that it is possible to delete and archive active objects as well as retired objects. Therefore, you must ensure that the object that you plan to archive is also relevant for deletion from the Alfabet database. The object class property **Integrity Info** of object classes specifies which object classes are dependent object classes. For general information about the structure of integrity reference definitions and information about the individual configuration for each relevant object class, see the reference manual *Alfabet Meta-Model*.

To delete an Alfabet object and create an archive object, you must first go to the **Simple Search** functionality available via the `Admin` user profile. Only users with access to the **Admin** application may archive an Alfabet object. Thus, an administrator will typically receive an assignment from an object's authorized user requesting the administrator to archive the object. In the **Simple Search** functionality, you must first search for and select the object that you plan to delete and archive and then click the **Archive Object**  button. The content of the archived object profile will depend on two factors:

- The class settings defined for the object class. Class settings are configured by your solution designer in the configuration tool Alfabet Expand. For more information, see the section *Configuring Class Settings*

for Object Classes and Object Class Stereotypes in the reference manual *Configuring Alfabet with Alfabet Expand*.

- The visibility of page views in an archived object profile is preconfigured by Software AG. In order to know which pages views are included in the archived object profile, Alfabet users should access the online Help for a relevant object profile that shows all page views potentially available in Alfabet for the object class. The **Archive Object** symbol denotes those page views that have been preconfigured to be included in the archived object profile.

You can delete and archive multiple objects simultaneously. Once an archive object has been created, it can be downloaded to a local disk via the **Archive Manager**. To delete an Alfabet object and create an archive object:

1) Open the **Simple Search** functionality available via the `Admin` user profile and enter relevant search criteria to find the object(s) that you want to archive. The **Simple Search** functionality is described in the section *Searching for Your Objects* reference manual *Getting Started with Alfabet*.
2) Select the object in the table that you want to delete and archive and click the **Archive Object** button in the toolbar.
3) Confirm the warning by clicking **Yes**. The Alfabet object will be deleted from the Alfabet database and an archive object will be created. Once the hourglass icon has disappeared, the archiving process is finished, and the archive object has been created. You can now access the archive object in the **Archive Manager** functionality and download it to a local disk and access the archived information. For more information, see Downloading and Storing an Archived Object.

Downloading and Storing an Archived Object

The **Archive Manager** functionality displays all objects that have been deleted and archived in the **Simple Search** functionality available via the `Admin` user profile. These objects have been deleted from the Alfabet database. The archive objects can now be downloaded to a local disk and stored.

To download an archive object:

1) Open the **Archive Manager** functionality.
2) If necessary, define filter settings to limit the objects in the table. In the **Archive Date** field, enter a date to show all objects archived after the date and in the **Search Pattern**, enter search criteria. Click Update.
3) In the table, select the object that you want to download and click **Download Archive**.
4) In the **Save As** window, define where you want the ZIP file to be saved and click **Save**. The ZIP file can be renamed, as needed.

Once the archive object has been downloaded, you can delete it from the **Archive Manager**. To delete the archive object, select it and click the **Delete** button. You can now store the ZIP file for the archive object according to the procedures defined by your enterprise. To access and view the archived information, see Accessing the Archived Information for an Archive Object.

Accessing the Archived Information for an Archive Object

Once an archive object has been downloaded, you can extract the ZIP file and access the archived information.

The ZIP file that is created via the **Archive Manager** functionality consists of a folder for each culture setting defined for your enterprise. (For example, the folder 1033 will contain HTML files for the English-language Alfabet interface, and 1031 will contain HTML files for the German-language interface.)

Each culture setting folder contains a file labelled `_Archive.html` as well as other HTML files and graphic files needed to view the object profiles of the selected archive object and its dependent objects.

To extract the ZIP file and access the archived information:

1) Navigate to the saved ZIP file and extract the ZIP file via the appropriate software utility. You will see a folder containing HTML files for each culture setting that your enterprise supports.
2) In the relevant culture setting folder, click the `_Achive.html` to open the object profile of the archive object. A browser window will open and you will see a snapshot of the object profile displaying a preconfigured set of page views.

   You may need to deactivate the pop-up blocker mechanism in your browser in order to open the object profiles of the archive object and its dependent objects.

3) In the archived object profile, navigate to a page view and double-click the dependent object in order to open its archived object profile.

Chapter 11: Defining Broadcast Messages for the User Community

A broadcast message is a message that can be made available to the entire user community or to specific users, user profiles, or user groups and will appear when a user logs in to the Alfabet Server. For example, broadcast messages could be defined to inform users when the system will be shut down for maintenance. General broadcast messages can be specified to be displayed for the period that they are defined as active and scheduled broadcast messages can be specified to be displayed during a defined set of start and end dates. Multiple broadcast messages can be specified.

All broadcast messages will be displayed in a pop-up window in the lower-right corner of the Alfabet interface. If the text is excessively long, a vertical scrollbar will be added to the window. If multiple broadcast images have been configured, the number of available broadcast messages to view will be displayed in the pop-up window. The user must click the X in the upper-right corner of the pop-up window to close the current message for the current user session and view the next one. The message can be closed for the current and future user sessions by clicking the **Do not show this message again** button.

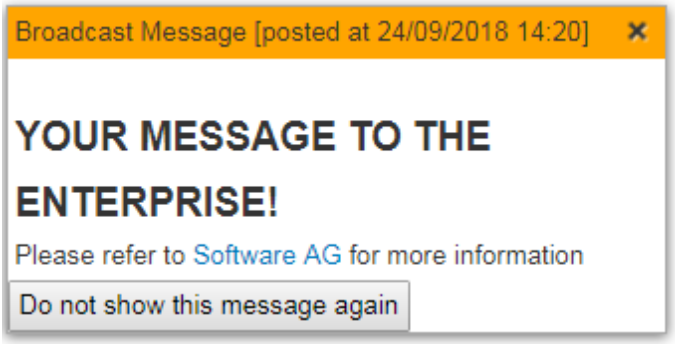


FIGURE: Example of a broadcast message

The broadcast message can be written in plain text or in HTML, which allows hyperlinks to be included. The HTML must be compliant with XHTML, XML-conform HTML, compliant with HTML 5, and use standard HTML tags. The HTML header implements standard HTML elements. Updates of broadcast messages on an Alfabet Web Client are only performed when the user is currently working with Alfabet. During periods of inactivity, no broadcast message updates will be performed.

The system administrator must set the **Broadcast Messages** attribute in the server alias settings to `True` to enable the sending of broadcast messages to the relevant users logged in to the Alfabet interface. For more information about configuring the server alias settings, see the section *Configuration Attributes for the Alfabet Components* in the reference manual *System Administration*.

Each column in the table is defined below:

- **Message**: The message that will be displayed in the status bar at the bottom of the Alfabet interface.
- **Active**: Displays an X if the message is activated. All activated messages will be displayed for users that log in to the Alfabet Server.
- **Creation Date**: The date that the broadcast message was originally created.
- **Is HTML**: Displays an X if the message is written in HTML format.
- **Start Date**: Displays the start date of a scheduled broadcast message.
- **End Date**: Displays the end date of a scheduled broadcast message.

- **Is Timed Message**: Displays an X if the broadcast message has a defined start and end date. If not X is displayed, the broadcast message is a general message that will be displayed for the period that it is active.

The following information is available:

- Creating and Activating a General Broadcast Message
- Creating a Scheduled Broadcast Message

Creating and Activating a General Broadcast Message

1) In the **Broadcast Messages** functionality, click **New** > **Create General Broadcast Message**. The **Broadcast Message** editor opens. Define the following as needed:
   - **Message**: Enter the text that is to be displayed. Please note the following if your specify HTML:
     - The HTML must be compliant with XHTML, XML-conform HTML, compliant with HTML 5, and use standard HTML tags. The HTML header implements standard HTML elements.
     - The code must start with an `<xhtml>` tag and end with `</xhtml>`. Standard HTML elements must be written in lower-case letters in order to be correctly parsed: `<xhtml>`, `<html>`, `<head>`, and `<body>`.
     - You must select the **Is HTML** checkbox.

     For example:

```
<xhtml>
    <html>
        <body>
            <h1>Your Message to the Enterprise! </h1>
            <p>Please refer to <a
            href="https:www.softwareag.com"
            target=_blank">Software AG </a> for more
            information</p>
        </body>
    </html>
</xhtml>
```

   - **Active**: Select the checkbox to activate the broadcast message. To deactivate the broadcast message and prevent it from displaying, remove the checkmark from the **Active** checkbox.
2) Define the users for which the broadcast message shall be available:
   - **Users** tab: Click the **Add Users** button and select the users that shall see the broadcast message.
   - **User Profiles** tab: Click the **Add User Profiles** button and select the user profiles that shall see the broadcast message.

- **User Groups** tab: click the **Add User Groups** button and select the user groups that shall see the broadcast message. Please note that inheritance is not applied to the user groups. You must explicitly select each user group that shall view the broadcast message.

Please note:

- If a user, user profile, and user group are specified to have access to the broadcase message, the `OR` condition will apply, and the broadcast message will be displayed to a user with a matching user name, user profile, or user group affiliation.
- If no users, user profiles, or user groups are specified, the broadcast message will be available to all users logged in to Alfabet

3) Click **OK** to save your changes or click **Cancel** to exit without saving.

Creating a Scheduled Broadcast Message

1) In the **Broadcast Messages** functionality, click **New** > **Create Scheduled Broadcast Message**. The **Broadcast Message** editor opens. Define the following as needed:

- **Message**: Enter the text that is to be displayed. Please note the following if your specify HTML:
  - The HTML must be compliant with XHTML, XML-conform HTML, compliant with HTML 5, and use standard HTML tags. The HTML header implements standard HTML elements.
  - The code must start with an `<xhtml>` tag and end with `</xhtml>`. Standard HTML elements must be written in lower-case letters to be correctly parsed: `<xhtml>`, `<html>`, `<head>`, and `<body>`.
  - You must select the **Is HTML** checkbox.

For example:

```
<xhtml>
    <html>
        <body>
            <h1>Your Message to the Enterprise! </h1>
            <p>Please refer to <a
            href="https:www.softwareag.com"
            target=_blank">Software AG </a> for more
            information</p>
        </body>
    </html>
</xhtml>
```

- Define the period of time when the message shall be displayed in the **Start Date** and **End Date** fields.

2) Define the users for which the broadcast message shall be available:

- **Users** tab: Click the **Add Users** button and select the users that shall see the broadcast message.

If no users, user profiles, or user groups are specified, the broadcast message will be available to all users logged in to Alfabet.

- **User Profiles** tab: Click the **Add User Profiles** button and select the user profiles that shall see the broadcast message.
- **User Groups** tab: click the **Add User Groups** button and select the user groups that shall see the broadcast message. Please note that inheritance is not applied to the user groups. You must explicitly select each user group that shall view the broadcast message.

3) Click **OK** to save your changes or click **Cancel** to exit without saving.

Chapter 12: Configuring the Risk Management Capability

Alfabet provides a risk assessment and risk evaluation capability that allows objects such as applications, business, data, projects, etc. in the enterprise to be evaluated for risk as well as for mitigations to be defined and planned to prevent or reduce the risks.

The risk assessment in Alfabet consists of two phases. A first phase is the **Risk Evaluation** phase and allows the enterprise to determine the base risk exposure of objects in the enterprise. This phase allows objects to be evaluated in order to determine whether they should be included in the risk assessment. The second phase is the **Risk Assessment** phase and allows the enterprise to document and assess the risk to the objects targeted by the risk evaluation. This includes defining the risk, its potential cost of damage to the object and that probability of damage to the object as well as the means to mitigate the risk. This approach allows the enterprise to streamline the risk assessment process and specifically target the objects that are considered most at risk.

To implement the **Risk Management** capability in Alfabet, you should configure the following:

- Risk management templates must be configured to specify the object classes that are the target of the risk evaluation, the questions to use for the **Risk Evaluation** phase, and the indicator types to define the risk damage and probability of risk damage to the risk object in the **Risk Assessment** phase. Risk management templates are defined in the Configuring Risk Management Templates for the Risk Management Functionality.
- Risk templates may be configured in order to group a standard set of risks and, if relevant, suggested mitigations to the risk object. A risk can be defined explicitly for a specific risk object or it can be added to the risk object by means of a configured risk template. A risk template is defined via the Risk Templates Page View available for a class-based risk management template.
- Risk mitigation templates may be configured in order to capture predefined risk mitigations for a specific threats in order to articulate how to avoid, reduce or contain the risk derived from a potential threat. The risk mitigation template includes the name of the risk mitigation, the target date when the risk mitigation should be implemented, and a number to prioritize the risk mitigation. The risk mitigation can then be defined for a risk object in the context of a risk assessment. Risk mitigation templates are configured via the Risk Mitigation Templates Explorer.

For an overview of the methodology of risk assessment and risk evaluation in Alfabet, see the reference manual *IT Governance, Risk and Compliance*.

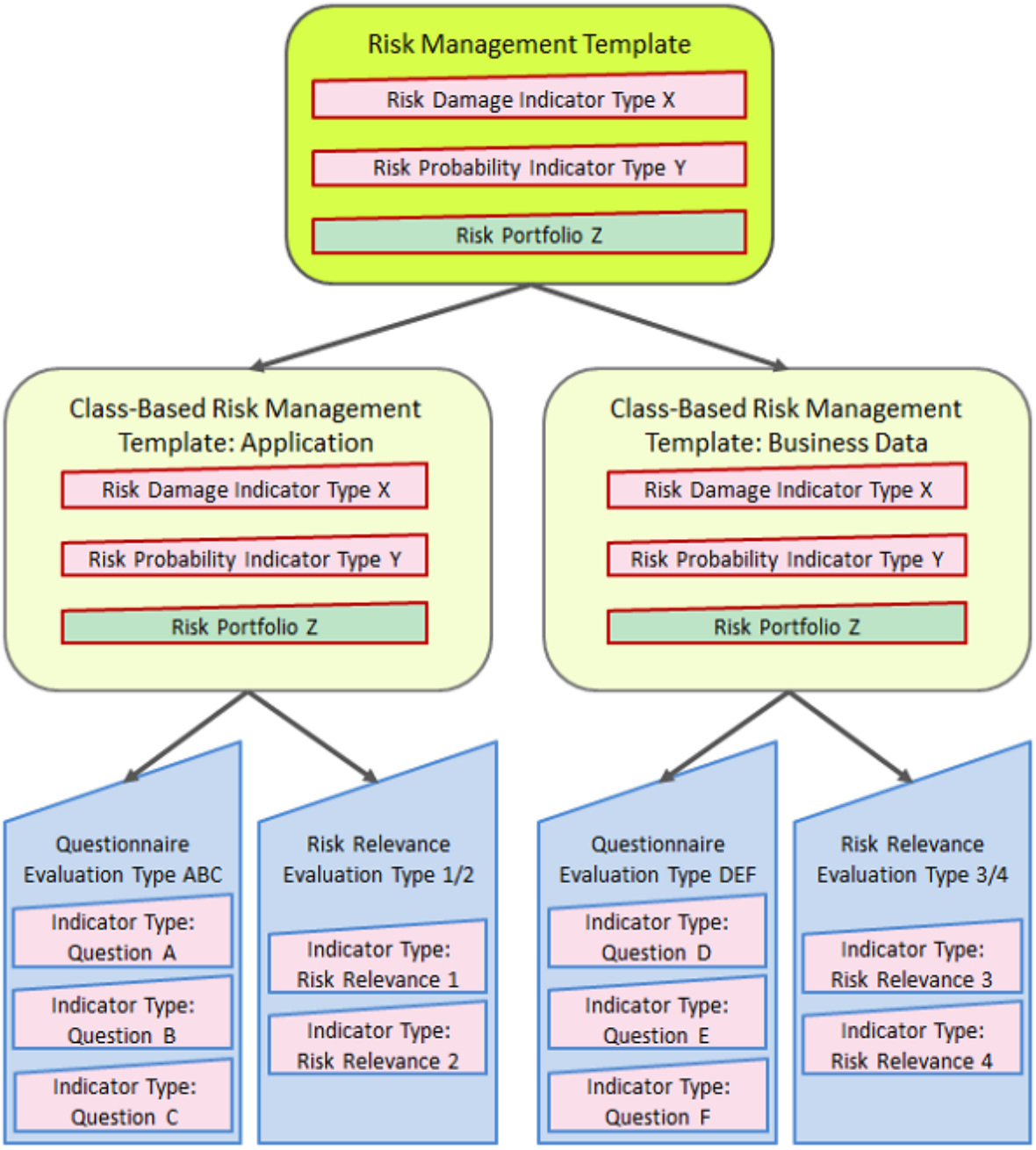
The following information is available:

Configuring Risk Management Templates for the Risk Management Functionality

In order to make the **Risk Management** capability available for the user community, a risk management template must be configured that serves as a blueprint used to evaluate and assess the risk to objects in the enterprise. For example, a risk assessment could address the risk of data in the context of General Data Protection Regulations (GDPR), the assessment of risks to a project during project execution, or the potential risk posed to applications and the impact on the enterprise's IT operations. For each risk management template, you must specify the object classes that are the target of the risk evaluation, the questions to use for the **Risk Evaluation** phase in order to determine which objects should be evaluated for risk, and the indicator types to define the risk damage and risk probably to the risk object in the **Risk Assessment** phase.

For an overview of the methodological approach to the **Risk Management** capability in Alfabet, see the chapter *Application Risk Management* in the reference manual *IT Governance, Risk and Compliance*. This chapter describes the implementation of the **Risk Management** capability for an application risk assessment, but the methodology described applies to any object class that is the target of a risk assessment.

Additionally, risk templates may be configured to bundle risks to ease the definition of risks to a risk object. Risk mitigation templates may be configured to ease the definition of mitigation of the risks to risk objects.



The following is required to configure a risk management template.

- A risk management template must be configured. The risk management template includes the definition of an indicator type that is used to assess the potential damage to risk objects to be assessed and an indicator type that allows the potential probability of damage to risk objects to be captured.
- One or more class-based risk management templates must be defined to capture the object classes that are targeted by the risk assessment. The class-based risk management templates inherit the risk damage and potential risk damage indicator types that are assigned to the risk management template that the class-based risk management templates are subordinate to. You can specify a class-based risk management template for each object class that the risk damage and potential risk damage indicator types are relevant for.
- An indicator lookup table must be defined for each class-based risk management template. The indicator lookup table consists of two sets of evaluation types that are used to evaluate the base risk

exposure of the objects. These evaluation types allow for an initial evaluation to determine which objects should actually be more closely assessed for risk. This allows the risk assessment to focus on specific objects that are most at risk rather than evaluating all objects for an object class. The following evaluation types are required:

- Evaluation types must be configured in order to specify the questions to be answered in order to determine whether an object should be evaluated for risk. In this case, each question is captured via an indicator type defined for the evaluation type. Each indicator type must have valid values defined via the **Range** attribute of the indicator type. The values defined for the **Range** attribute constitute the potential answer to the question.

  It is highly recommended that the user designing the risk project pursues a pragmatic qualitative approach that is directed at the relevant stakeholders. The evaluation questions should be a compact set of questions with simple answers and the answers should be mapped to numeric values for easy analysis.

- Indicator types must be configured that represent an issue of relevance for the risk evaluation (such as **Data Protection**, **Data Sensitivity,** and **Risk of Leakage**). The answer to each question will be mapped as a numerical value to each indicator type. The mapped values for all questions for all indicator types will be added together to produce a value that represents the risk relevance score. The risk relevance score indicates how high the base risk exposure is for the risk object and whether the object should/should not be channeled to the next stage of risk assessment.

- Once the evaluation types have been defined for the indicator lookup table, you must map the possible answers for each question to a numerical value for each risk relevance score. In other words, you must define a numerical value for each value defined in the **Range** attribute of each indicator type of the questionnaire evaluation types to an integer for each indicator type specified for the risk relevance score evaluation types.

- One or more risk templates may be configured that group a standard set of risks and mitigations that might be relevant for the risk objects targeted by the risk evaluation. The risk templates can be assigned to the class-based risk management template.

- One or more risk mitigation templates may be configured in order to capture predefined risk mitigations for a specific threat in order to articulate how to avoid, reduce or contain the risk derived from a potential threat.

The **Risk Management Templates** functionality allows you to configure one or more risk management templates. When working with the **Risk Management Templates** functionality, you can create new risk management templates, class-based risk management templates, and indicator lookup tables. The explorer displays the object classes **Risk Management Template**  , **Class-Based Risk Management Template**  , and **Indicator Lookup Table**  . Click any object in the explorer to see its object profile.

The following information is available:

- Configuring Evaluation Types and Indicator Types for the Risk Management Capability
- Creating a Risk Management Template
- Creating a Risk Management Template as a Copy of an Existing Risk Template
- Defining the Object Classes to Evaluate in the Risk Assessment
- Creating an Indicator Lookup Table for a Class-Based Risk Management Template

- Creating a New Indicator Lookup Table
- Adding an Existing Indicator Lookup Table
- Replacing an Indicator Lookup Table
- Detaching the Indicator Lookup Table from the Class-Based Risk Management Template

- Assigning Evaluation Types to the Indicator Lookup Table
- Mapping the Values in the Indicator Lookup Table

Configuring Evaluation Types and Indicator Types for the Risk Management Capability

In order to implement the **Risk Management** capability in Alfabet, several indicator types must be configured. The following is required:

- Two indicator types are required to assess potential damage to risk objects:
  - One indicator type must be configured to capture the potential damage to risk objects as well as the damage after suggested mitigation actions have been implemented. The damage potential is captured in the context of risk assessment executed for the risk management template. Please note the following:
    - The **Name** attribute defined for the indicator type will be displayed as the field caption in the **Risk** editor. The **Description** attribute should provide an explanation of the indicator type for the users evaluating their objects for risk. The options that can be selected for the indicator type must be defined in the **Range** attribute of the indicator type.
    - The indicator type is assigned to the risk management template in the **Risk Damage Indicator Type** field in the **Risk Management Template** editor.
    - The risk is captured in the **Risk** editor in the *Risk Assessment Page View* of a risk object. The name of the indicator will be displayed as the caption of the field where the risk damage is defined.
  - One indicator type must be configured to capture the potential probability of the damage to risk objects as well as the probability of the damage after suggested mitigation actions have been implemented. The damage potential is captured in the context of risk assessment executed for the risk management template. Please note the following:
    - The **Name** attribute defined for the indicator type will be displayed as the field caption in the **Risk** editor. The **Description** attribute should provide an explanation of the indicator type for the users evaluating their objects for risk. The options that can be selected for the indicator type must be defined in the **Range** attribute of the indicator type.
    - The indicator type is assigned to the risk management template in the **Risk Probability Indicator Type** field in the **Risk Management Template** editor.
    - The risk is captured in the **Risk** editor in the *Risk Assessment Page View* of a risk object. The name of the indicator will be displayed as the caption of the field where the probability of the risk damage is defined.
- Two sets of evaluation types are required to evaluate the base risk exposure of objects.
  - One or more evaluation types must be configured that have a set of indicator types that represents the questions to be answered in order to determine whether an object should be assessed for risk.

The potential answers to the questions are captured via the valid values defined for the **Range** attribute of the indicator type.

For a GDPR Risk Assessment, for example, the following questions (indicator types) and answers (value range) may be relevant to identify whether an object should be included in the risk assessment.

Indicator Type	Values of Range Attribute
Data Anonymized?	• 1 - Yes • 2 - No
Data Encrypted?	• 0 - Fully Encrypted • 1 - Partially Encrypted • 2 - Not Encrypted
Retention Period of Processed Data	• 1 - < 3 years • 2 - Between 3 and 30 years • 3 - > 30 years

- The **Name** attribute defined for the indicator type will be displayed as the question that the user evaluating the object must answer. The answers that can be defined for the indicator type must be defined in the **Range** attribute of the indicator type.
- The evaluation type is assigned to the indicator lookup table defined for a class-based risk management template via the **Add Questionnaire Evaluation Types** option in the Indicator Lookup Table Structure Page View available for the relevant class-based release management template.
- The questions represented by the indicator types defined for the evaluation type are defined for a risk object in the *Risk Relevance Questionnaire Page View* of a risk object.

- One or more evaluation types must be configured that have a set of indicator types that represents an issue of relevance for the risk evaluation. The answer to each question will be mapped as a numerical value to the indicator type. The numerical values for all indicator types for all questions will be added together to generate a risk relevance score for the risk object. The risk relevance helps to determine the base risk exposure of the risk object, in which areas the risk exists, and whether the object should/should not be channeled to the next stage of risk assessment.

For a GDPR Risk Assessment, for example, the risk relevance scores could be **Data Protection**, **Data Sensitivity,** and **Risk of Leakage**.

- The **Name** attribute defined for the indicator type will be displayed as the name of issue that the score is being defined for. The definition of the **Range** attribute is not required for this indicator type.

- The evaluation type is assigned to the indicator lookup table defined for a class-based risk management template via the **Add Risk Relevance Score Evaluation Types** option in the Indicator Lookup Table Structure Page View available for the relevant class-based release management template.
- The questions represented by the indicator types defined for the evaluation type are defined for a risk object in the *Risk Relevance Questionnaire Page View* of a risk object.

- The indicator types representing the questions and their answers must be mapped to the indicator types representing the risk relevance scores in the Indicator Lookup Value Mapping Page View. When an object is being evaluated for base risk exposure, the risk relevance scores will be added together for that object in order to determine whether the base risk exposure is above a threshold specified in the *Assigned Objects Page View*. All objects with a score above a defined threshold will be considered risk objects and will require additional detailed assessment of their potential risk damage and probable risk damage.

Indicator Type (Question)	Range Attribute (Answer)	Indicator Type (Risk Relevance): Data Protection	Indicator Type (Risk Relevance): Data Sensitivity	Indicator Type (Risk Relevance): Risk of Leakage
Data Anonymized?				
	1 - Yes	0.00	0.00	0.00
	2 - No	0.00	0.00	10.0
Data Encrypted ?				
	0 - Fully Encrypted	1.0	0.00	1.0
	1 - Partially Encrypted	5.0	0.00	10.0
	2 - Not Encrypted	10.0	0.00	10.0
Retention Period of Processed Data				
	1 - < 3 years	0.00	0.00	5.0

Indicator Type (Question)	Range Attribute (Answer)	Indicator Type (Risk Relevance): Data Protection	Indicator Type (Risk Relevance): Data Sensitivity	Indicator Type (Risk Relevance): Risk of Leakage
	2 - Between 3 and 30 years	0.00	0.00	10.0
	3 - > 30 years	0.00	0.00	15.0

For more information about defining evaluation types and indicator types, see the section *Configuring Evaluations, Prioritization Schemes, and Portfolios* in the reference manual *Configuring Evaluation and Reference Data in Alfabet*.

Creating a Risk Management Template

To create a new risk management template:

1) Click the **Risk Management Templates** icon at the top of the explorer.
2) In the toolbar, click **New** > **Create Risk Management Template**. You will see the **Risk Management Template** editor.
3) Enter information into each field, as required.

   Basic Data tab:

- **Name**: Enter a unique name for the risk management template.
- **Description**: Enter a meaningful description that will clarify the purpose of the risk management template.
- **Risk Damage Indicator Type**: Select a configured indicator type to use to assess potential damage to risk objects. The indicator type will be used to assess the potential damage if a risk occurs as well as if the risk is reduced/avoided by means of a mitigation.
- **Risk Probability Indicator Type**: Select a configured indicator type to use to assess potential probability of damage to risk objects. The indicator type will be used to assess the potential probability of a risk occurring as well as if the probability of the risk occurring if a mitigation is implemented.

Risk damage and risk probability are captured in the **Risk** editor in the *Risk Assessment Page View* of a risk object. The name of the indicator types will be displayed as captions of the fields where the risk damage/risk probability is defined.

For details about the configuration of the indicator types for the risk management capability, see the section Configuring Evaluation Types and Indicator Types for the Risk Management Capability. For general information about defining indicator types, see the section *Configuring Evaluations, Prioritization Schemes, and Portfolios* in the reference manual *Configuring Evaluation and Reference Data in Alfabet*.

- **Risk Portfolio**: Select a configured portfolio to implement to report and analyze the risk evaluation. This portfolio will be available in the *Risks Portfolio Page View* displayed for a selected risk object as well as the *Risk Objects Portfolio Page View* displayed for a selected risk management group.

4) Click **OK** to save the risk management template or **Cancel** if you do not want to save it.

- If you delete a risk management template, it will be irrevocably deleted from the Alfabet database.
- If you delete a risk management template that has subordinate class-based risk management templates, all class-based risk management templates, the lookup tables defined for the class-based templates as well as any previously assigned risk objects will be irrevocably deleted from the Alfabet database.
- If any of the following dependent objects or data is defined for the deleted object, these will also be deleted: Assignment, Attachment, Deputy, Evaluation, Role.

Creating a Risk Management Template as a Copy of an Existing Risk Template

You can create a new risk mangement template based on an existing risk management copy. The class-based risk management template as well as indicator lookup table will be copied to the new risk management template but may be modified. This allows you to make changes to the structure of the indicator lookup table. Changes can only be made to the indicator lookup table if no existing risk objects are associated with the indicator lookup table.

1) Click the **Risk Management Templates** icon at the top of the explorer.
2) In the toolbar, click **New** > **Create Risk Management Template as Copy**.
3) In the selector, select the risk management template that you want to copy to the new risk management template and click **OK**.
4) In the **Risk Management Template** editor, enter information into each field, as required.

   Basic Data tab:

- **Name**: Enter a unique name for the risk management template.
- **Description**: Enter a meaningful description that will clarify the purpose of the risk management template.
- **Risk Damage Indicator Type**: Select a configured indicator type to use to assess potential damage to risk objects. The indicator type will be used to assess the potential damage if a risk occurs as well as if the risk is reduced/avoided by means of a mitigation.
- **Risk Probability Indicator Type**: Select a configured indicator type to use to assess potential probability of damage to risk objects. The indicator type will be used to assess the potential probability of a risk occurring as well as if the probability of the risk occurring if a mitigation is implemented.

Risk damage and risk probability are captured in the **Risk** editor in the *Risk Assessment Page View* of a risk object. The name of the indicator types will be displayed as captions of the fields where the risk damage/risk probability is defined.

For details about the configuration of the indicator types for the risk management capability, see the section Configuring Evaluation Types and Indicator Types for the Risk Management Capability. For general information about defining indicator types, see the section *Configuring Evaluations, Prioritization Schemes, and Portfolios* in the reference manual *Configuring Evaluation and Reference Data in Alfabet*.

- **Risk Portfolio**: Select a configured portfolio to implement to report and analyze the risk evaluation. This portfolio will be available in the *Risks Portfolio Page View* displayed for a selected risk object as well as the *Risk Objects Portfolio Page View* displayed for a selected risk management group.

5) Click **OK** to save the risk management template or **Cancel** if you do not want to save it. Continue to modify the risk management template as needed. For more information, see the following:

- Defining the Object Classes to Evaluate in the Risk Assessment
- Creating an Indicator Lookup Table for a Class-Based Risk Management Template
- Assigning Evaluation Types to the Indicator Lookup Table
- Mapping the Values in the Indicator Lookup Table

Defining the Object Classes to Evaluate in the Risk Assessment

The **Class-Based Risk Management Template** page view allows you to create one or more class-based risk management templates for the selected risk management template. The class-based risk management template allows you to specify the object class that will be the focus of the risk evaluation and risk assessment.

Multiple class-based risk management templates can be created for a risk management template. The class-based risk management template will inherit the risk damage and risk probability indicator types and the risk portfolio defined for the ascendant risk management template. You should only create one class-based risk management template per object class per risk management template.

Once the class-based risk management template has been created, you can specify the evaluation types that capture the questions to be asked to assess whether an object should be evaluated for risk. Each class based-risk management template may require different evaluation types. The evaluation types are captured for the class-based risk management template in an indicator lookup table which is created in the Indicator Lookup Table Page View.

To create a new class-based risk management template:

1) Click the risk management template in the **Risk Management Template** s explorer and then click **Class-Based Risk Management Template**.
2) In the toolbar, click **New** > **Create Class-Based Risk Management Template**. You will see the **Class-Based Risk Management Template** editor.
3) In the **Class Name** field, select the object class that should be the focus of the risk assessment. The classes will be listed alphabetically using the syntax: Object Class:Caption or Object Class Stereotype:Caption.
4) In the **Description** field, enter a meaningful description that will clarify the purpose of the class-based risk management template.
5) Click **OK** to save the class-based risk management template or **Cancel** if you do not want to save it.

If you delete a class-based risk management template, it will be irrevocably deleted from the Alfabet database. The indicator lookup table assigned to the class-based risk management template will also be deleted.

Creating an Indicator Lookup Table for a Class-Based Risk Management Template

The **Indicator Lookup Table** page view allows you to create one indicator lookup table for a selected class-based risk management template. An indicator lookup table may be reused and assigned to multiple class-based risk management templates.

The indicator lookup table is a means to evaluate the base risk exposure of a specified set of objects and determine whether the objects should be assessed for risk. The indicator lookup table captures two different evaluation types used to evaluate the base risk exposure as well as the mapping of the two evaluation types to create a risk relevance score. One evaluation type groups a set of indicator types that represent the questions to be answered in order to determine whether an object should be assessed for risk. The other evaluation type groups a set of indicator types that allow the answer to each question to be translated to a risk relevance score - a score that determines how high the risk is to the object and whether the object should/should not be channeled to the next stage of risk assessment.

The indicator lookup table thus also specifies the mapping between the answers to the questions and the risk relevance score. Each potential value defined for a question will have a corresponding risk relevance score value. When an object is being evaluated for its base risk exposure, the risk relevance scores will be added for that object in order to determine a risk relevance score. All objects with a score above the threshold will be considered risk objects and will require additional detailed assessment of their potential risk damage and risk probability.

Once the indicator lookup table has been created in the **Indicator Lookup Table** page view, you can do the following:

- Assign the evaluation types to the indicator lookup table in the Indicator Lookup Table Structure Page View available for the relevant class-based release management template.
- Map the indicator types representing the questions and their answers to the indicator types representing the risk relevance scores in the Indicator Lookup Value Mapping Page View.

The following information is available:

- Creating a New Indicator Lookup Table
- Adding an Existing Indicator Lookup Table
- Replacing an Indicator Lookup Table
- Detaching the Indicator Lookup Table from the Class-Based Risk Management Template

Creating a New Indicator Lookup Table

Only one indicator lookup table may be assigned to a class-based risk management template.

1) In the **Class-Based Risk Management Template** page view, select the relevant class-based risk management template and click the **Navigate**  button.

2) Click **Indicator Lookup Table**.
3) In the toolbar, click **New** > **Create Indicator Lookup Table**. The **Indicator Lookup Table** editor opens.
4) In the **Name** field, enter the name of the indicator lookup table and in the **Description** field, enter a meaningful description that will clarify the purpose of the indicator lookup table. For example, you could provide a description explaining the sets of indicator types that are being mapped for the risk assessment.
5) Click **OK** to save the indicator lookup table or **Cancel** if you do not want to save it.
6) Double-click the indicator lookup table in order to navigate to the Indicator Lookup Table Structure Page View and assign the evaluation types for questionnaire items and risk relevance score evaluations to the indicator lookup table.

Adding an Existing Indicator Lookup Table

You can add an existing indicator lookup table to the class-based risk management template. Only one indicator lookup table may be assigned to a class-based risk management template. Therefore, if an indicator lookup table is already assigned to the class-based risk management template, you will not be able to add an existing indicator lookup table. If this is the case, you can create a new risk management template based on an existing risk management copy. The class-based risk management template as well as indicator lookup table will be copied to the new risk management template but may be modified. For more information, see the section Creating a Risk Management Template as a Copy of an Existing Risk Template.

1) In the **Class-Based Risk Management Template** page view, select the relevant class-based risk management template and click the **Navigate**  button.
2) Click **Indicator Lookup Table**.
3) In the toolbar, click **New** > **Create Indicator Lookup Table**.
4) In the selector that opens, select the indicator lookup table that you want to add to the selected class-based risk management template and click **OK**.
5) The **Indicator Lookup Table** editor opens. In the **Name** field, enter the name of the indicator lookup table and in the **Description** field, enter a meaningful description that will clarify the purpose of the indicator lookup table. For example, you could provide a description explaining the sets of indicator types that are being mapped for the risk assessment.
6) Click **OK** to save the indicator lookup table or **Cancel** if you do not want to save it.
7) Double-click the indicator lookup table in order to navigate to the Indicator Lookup Table Structure Page View and assign the evaluation types for questionnaire items and risk relevance score evaluations to the indicator lookup table.

Replacing an Indicator Lookup Table

You can replace an indicator lookup table with a new or existing indicator lookup table if no existing risk objects are associated with the indicator lookup table. Changes can only be made to the indicator lookup table if no existing risk objects are associated with the indicator lookup table.

If risk objects already exist, the **Replace with New Indicator Lookup Table** and **Replace with Existing Indicator Lookup Table** options will be disabled. In this case, you can create a new risk management template based on an existing risk management copy. The class-based risk management template as well as indicator lookup table will be copied to the new risk management template but may be modified. This allows you to make changes to the structure of the indicator lookup table. For more information about creating a new risk management template based on the existing risk management template, see the section Creating a Risk Management Template as a Copy of an Existing Risk Template.

1) In the **Class-Based Risk Management Template** page view, select the relevant class-based risk management template and click the **Navigate** button.
2) Click **Indicator Lookup Table**.
3) In the toolbar, click one of the following: **New** > **Create Indicator Lookup Table**. The **Indicator Lookup Table** editor opens.
   - Click **New** > **Replace with New Indicator Lookup Table**. The **Indicator Lookup Table** editor opens.
   - Click **New** > **Replace with Existing Indicator Lookup Table**. In the selector that opens, select the indicator lookup table that you want to replace the current indicator lookup table with and click **OK**. Click the **Edit** button to open the editor and modify the **Name** or **Description** fields.
4) In the **Name** field, enter the name of the indicator lookup table and in the **Description** field, enter a meaningful description that will clarify the purpose of the indicator lookup table. For example, you could provide a description explaining the sets of indicator types that are being mapped for the risk assessment.
5) Click **OK** to save the indicator lookup table or **Cancel** if you do not want to save it.
6) Double-click the indicator lookup table in order to navigate to the Indicator Lookup Table Structure Page View and assign the evaluation types for questionnaire items and risk relevance score evaluations to the indicator lookup table.

Detaching the Indicator Lookup Table from the Class-Based Risk Management Template

You can remove an indicator lookup table from the selected class-based risk management template if no existing risk objects are associated with the indicator lookup table. If the indicator lookup table is detached from the selected class-based risk management template, the evaluation type and its indicator types will be removed from the risk management template.

If risk objects already exist, the **Detach** button will be disabled. In this case, you can create a new risk management template based on an existing risk management copy. The class-based risk management template as well as indicator lookup table will be copied to the new risk management template but may be modified. This allows you to make changes to the structure of the indicator lookup table or remove it from the class-based risk management template. For more information about creating a new risk management template based on the existing risk management template, see the section Creating a Risk Management Template as a Copy of an Existing Risk Template.

1) In the **Class-Based Risk Management Template** page view, select the relevant class-based risk management template and click the **Navigate** button.
2) Click **Indicator Lookup Table**.

3) In the toolbar, click the Detach button.

4) Click **OK** to save the indicator lookup table or **Cancel** if you do not want to save it.

5) Double-click the indicator lookup table to navigate to the Indicator Lookup Table Structure Page View and assign the evaluation types for questionnaire items and risk relevance score evaluations to the indicator lookup table.

Assigning Evaluation Types to the Indicator Lookup Table

The **Indicator Lookup Table Structure** page view allows you to assign the evaluation types relevant to assess the base risk exposure of the objects targeted by the risk evaluation to the indicator lookup table. Two different types of evaluation types are required to evaluate the base risk exposure of objects.

- One or more evaluation types must be configured, where each evaluation type groups a set of indicator types that represents the questions to be answered to determine whether an object should be assessed for risk. The potential answers to the questions are captured via the valid values defined for the **Range** attribute of the indicator type.

  For a GDPR Risk Assessment, for example, the following questions (indicator types) and answers (value range) may be relevant to identify whether an object should be included in the risk assessment.

Indicator Type	Values of Range Attribute
Data Anonymized?	• 1 - Yes • 2 - No
Data Encrypted?	• 0 - Fully Encrypted • 1 - Partially Encrypted • 2 - Not Encrypted
Retention Period of Processed Data	• 1 - < 3 years • 2 - Between 3 and 30 years • 3 - > 30 years

- The **Name** attribute defined for the indicator type will be displayed as the question that the user evaluating the object must answer. The answers that can be defined for the indicator type must be defined in the **Range** attribute of the indicator type.
- The evaluation type is assigned to the indicator lookup table defined for a class-based risk management template via the **Add Questionnaire Evaluation Types** option in the Indicator Lookup Table Structure Page View available for the relevant class-based release management template.
- The questions represented by the indicator types defined for the evaluation type are defined for a risk object in the *Risk Relevance Questionnaire Page View* of a risk object.

- One or more evaluation types must be configured, where each evaluation type groups a set of indicator types that allow the answer to each question to be captured as a score for an aspect relevant for the risk evaluation. The score helps to determine how high the risk might be to the object and whether the object should/should not be channeled to the next stage of risk assessment.

  For a GDPR Risk Assessment, for example, the risk relevance scores could be **Data Protection**, **Data Sensitivity,** and **Risk of Leakage**.

  - The **Name** attribute defined for the indicator type will be displayed as the name of issue that the score is being defined for. The definition of the **Range** attribute is not required for this indicator type.
  - The evaluation type is assigned to the indicator lookup table defined for a class-based risk management template via the **Add Risk Relevance Score Evaluation Types** option in the Indicator Lookup Table Structure Page View available for the relevant class-based release management template.
  - The questions represented by the indicator types defined for the evaluation type are defined for a risk object in the *Risk Relevance Questionnaire Page View* of a risk object.

- The indicator types representing the questions and their answers must be mapped to the risk relevance scores in the Indicator Lookup Value Mapping Page View. When an object is being evaluated for base risk exposure, the risk relevance scores will be added together for that object in order to determine whether the base risk exposure is above a threshold specified in the *Assigned Objects Page View*. All objects with a score above a defined threshold will be considered risk objects and will require additional detailed assessment of their potential risk damage and probable risk damage.

Indicator Type (Question)	Range Attribute (Answer)	Risk Relevance Score: Data Protection	Risk Relevance Score: Data Sensitivity	Risk Relevance Score: Risk of Leakage
Data Anonymized?				
	1 - Yes	0.00	0.00	0.00
	2 - No	0.00	0.00	10.0
Data Encrypted ?				
	0 - Fully Encrypted	1.0	0.00	1.0
	1 - Partially Encrypted	5.0	0.00	10.0

Indicator Type (Question)	Range Attribute (Answer)	Risk Relevance Score: Data Protection	Risk Relevance Score: Data Sensitivity	Risk Relevance Score: Risk of Leakage
	2 - Not Encrypted	10.0	0.00	10.0
Retention Period of Processed Data				
	1 - < 3 years	0.00	0.00	5.0
	2 - Between 3 and 30 years	0.00	0.00	10.0
	3 - > 30 years	0.00	0.00	15.0

Once the evaluation types has been assigned to the indicator lookup table, you can map the potential values of the questionnaire items to the risk relevance score evaluations in the Indicator Lookup Value Mapping Page View.

If an incorrect reference is defined to either a questionnaire item or risk relevance score evaluation, you should delete the indicator lookup table in the **Class-Based Risk Management Template** page view and create a new one.

The table displays the indicator types associated with the evaluation types that have been assigned to the selected indicator lookup table. The table section **Questionnaire Items** displays the evaluation types followed by their indicator types. The indicator types represent the questions that are asked about the objects in the relevant object class. The table section **Risk Relevance Score Evaluation** displays the evaluation types followed by their indicator types that have been added as the aspect of risk relevance that should be evaluated for each question.

To assign the evaluation types to the indicator lookup table:

1) In the **Indicator Lookup Table** page view, select the relevant indicator lookup table and click the **Navigate** button.
2) Click **Indicator Lookup Table Structure**.
3) In the toolbar, select **New** > **Add Questionnaire Evaluation Types**.
4) The object selector opens. Enter search criteria, as needed, and click **Search**.
5) Select the relevant evaluation type and click **OK**. The indicator types are displayed below the **Questionnaire Item** heading in the table.
6) To assign an evaluation type to the indicator lookup table, select **New** > **Add Risk Relevance Score Evaluation Types**.
7) The object selector opens. Enter search criteria, as needed, and click **Search**.

8) Select the relevant evaluation type and click **OK**. The indicator types are displayed below the **Risk Relevance Score Evaluation** heading in the table.

Mapping the Values in the Indicator Lookup Table

The **Indicator Lookup Table Value Mapping** page view allows you to map the possible answers for each indicator type (the questions) to a numerical value for each risk relevance score evaluation. The numerical values will then be automatically added to generate a risk relevance score for an object that has been evaluated.

The indicator types representing the questions and their answers must be mapped to the risk relevance scores. When an object is being evaluated for its base risk exposure, the risk relevance scores will be added together to produce a total score for that object to determine whether the base risk exposure is above a threshold specified in the *Assigned Objects Page View*. All objects with a score above the defined threshold will be considered risk objects and will require additional detailed assessment to identify the risks as well as potential risk damage and risk probability.

The mapping for a GDPR risk assessment could be defined as follows:

Indicator Type (Question)	Range Attribute (Answer)	Risk Relevance Score: Data Protection	Risk Relevance Score: Data Sensitivity	Risk Relevance Score: Risk of Leakage
Data Anonymized?				
	1 - Yes	0.00	0.00	0.00
	2 - No	0.00	0.00	10.0
Data Encrypted ?		0.00	0.00	0.00
	0 - Fully Encrypted	1.0	0.00	1.0
	1 - Partially Encrypted	5.0	0.00	10.0
	2 - Not Encrypted	10.0	0.00	10.0
Retention Period of Processed Data				

	1 - < 3 years	0.00	0.00	5.0
	2 - Between 3 and 30 years	0.00	0.00	10.0
	3 - > 30 years	0.00	0.00	15.0

The **Indicator Lookup Table Value Mapping** page view displays the following:

- The first column displays the indicator types that represent the questions. The answers available for a question are displayed below the indicator type. These are the values defined in the **Range** attribute of each indicator type.
- The subsequent columns display the indicator types assigned as risk relevance score evaluations.
- The entry **Undefined** has also been added below each indicator type in order to specify the values if a user does not provide define one of the indicator types representing a risk relevance evaluation score.
- The values in the matrix cells represent the numerical value mapping the indicator type representing the question and the indicator type representing the risk relevance score.

To map the indicator types for a selected indicator lookup table:

1) In the **Indicator Lookup Table** page view, select the relevant indicator lookup table and click the **Navigate** button.
2) Click **Indicator Lookup Table Value Mapping**.
3) To define a value for each possible indicator that may be selected for a questionnaire item and map it to a risk relevance score evaluation, click the **Edit** button and select **Edit Values** in the drop-down menu. The **Values Editor** opens.
   - In the rows displaying indicator types representing questions, enter a numerical value in each matrix cell corresponding to an indicator type representing a risk relevance score.
   - In the cells for the row **Undefined**, enter a numerical value in each matrix cell if a user does not provide a value for the corresponding indicator type representing a risk relevance score.
4) Once all values have been mapped, click **OK** to save the mapping.

To clear the values from the table, click the **Edit** button and select **Clear Values** in the drop-down menu.

Once the values have been mapped in the indicator lookup table, you can create and define a risk template in the Risk Template Page View.

Configuring Risk Templates for the Risk Management Functionality

A risk template bundles a set of risks and, if necessary, their suggested mitigations. Multiple risk templates can be assigned to a class-based risk management template.

When a risk object is assessed, the risk template can be selected in order to add a standard set of risks and, if relevant, their mitigations to the risk object. Irrelevant risks can be removed from the risk object and additional risks that may not have been captured in the risk template can be defined for the risk object.

The following information is available:

- Defining a Risk Template for the Class-Based Risk Management Template
- Adding Risks to the Risk Template

Defining a Risk Template for the Class-Based Risk Management Template

The **Risk Templates** page view allows you to create a risk template. You can add risks to the risk template via the Risks Page View. Users will then be able to select a risk template and automatically add the relevant risks to a risk object.

Risks that are not relevant to the risk object can then be deleted or new risks that are not included in the risk template can be defined for the risk object.

Once you have created a risk template, you can add risks to it. For more information, see the section Adding Risks to the Risk Template.

To create a new risk template for the selected class-based risk management template:

1) In the **Class-Based Risk Management Template** page view, select the relevant class-based risk management template and click the **Navigate**  button.
2) Click **Risk Templates**.
3) In the toolbar, click **New** > **Create Risk Template**. You will see the **Risk Template** editor.
4) Define the **Name** and **Description** fields.
5) Click **OK** to save the risk template or **Cancel** if you do not want to save it.

- If you delete a risk template, it will be irrevocably deleted from the Alfabet database.
- If risks have been assigned to the risk template, they will also be irrevocably deleted from the Alfabet database.
- If a risk template and its risks have been assigned to a risk object, the risks will not be deleted from the risk object if the risk template is deleted.

Adding Risks to the Risk Template

The **Risks** page view allows you to define risks and, if necessary, any suggested mitigations for the selected risk template. The risk will automatically be added to a risk object when the risk template that the risk is assigned to

is specified for the risk object. If the risk is not relevant, it can be deleted from the risk object. An unlimited number of risks can be assigned to a risk template.

The page view displays all risks assigned to the selected risk template. The following columns are displayed:

- **Name**: The name of the risk.
- **Description**: A description of the risk.
- **Suggested Mitigation Actions**: Describes the mitigation actions that could potentially mitigate the risk.

To create a new risk:

1) In the **Risk Templates** page view, select the relevant risk template and click the **Navigate**  button.
2) Click **Risks**.
3) In the toolbar, click **New** > **Create Risk**. You will see the **Risk** editor.
   - **Name**: Enter a name for the risk.
   - **Description**: Enter a meaningful description that will clarify the purpose of the risk.
   - **Suggested Mitigation Actions**: Enter a description of a standard mitigation for this risk.

     The definition in the **Suggested Mitigation Actions** attribute is informational only. To create a risk mitigation for a risk that can be tracked and implemented, the user assessing the risk object must select the risk in the *Risk Assessment Page View* of the risk object and navigate to it's object profile. The risk mitigation can be created for the risk in the *Risk Mitigations Page View*. The user can later define the architecture objects in the IT landscape that are impacted by the risk mitigation and specify a demand based on the risk mitigation.

4) Click **OK** to save the risk template or **Cancel** if you do not want to save it.

Configuring Risk Mitigation Templates for the Risk Management Functionality

The **Risk Mitigation Templates** functionality allows risk mitigations to be planned and implemented in order to address potential risks as well as avoid, reduce or contain risks to the IT landscape. Risk mitigations may be standardized by means of risk mitigation templates that allow consistent data to be captured for a set of risk mitigations.

The **Risk Mitigation Templates** functionality allows risk mitigation template categories and risk mitigation templates be defined. You can assign the risk mitigation template to a risk mitigation template category via the **Risk Mitigation Templates** functionality or to a threat or threat group via the *Threat Management Functionality*.

To create a risk mitigation that can be tracked and implemented for a risk, the user assessing the risk object must select the risk in the *Risk Assessment Page View* of the risk object and navigate to it's object profile. The risk mitigation can be created for the risk in the *Risk Mitigations Page View* based on the risk mitigation template. The user can later define the architecture objects in the IT landscape

that are impacted by the risk mitigation and specify a demand for the risk mitigation. A project can then be created for the demand in order to realize the risk mitigation.

The **Risk Mitigation Templates** explorer displays risk mitigation template categories  and risk mitigation templates  . Click a risk mitigation template category or a risk mitigation template in the explorer to see its object profile or object cockpit.

The following information is available:

- Creating a Risk Mitigation Template Category at the Top Level of the Hierarchy
- Finding Risk Mitigation Templates Not Assigned to a Risk Mitigation Template Category
- Creating a Risk Mitigation Template for the Risk Mitigation Template Category

Creating a Risk Mitigation Template Category at the Top Level of the Hierarchy

The **Root Risk Mitigation Template Categories** view allows you to create one or more risk mitigation template categories at the top level of the risk mitigation template category hierarchy.

To create a new risk mitigation template category:

1) Click the **Risk Mitigation Templates** icon  at the top of the explorer and click **Root Risk Mitigation Template Categories**.
2) In the toolbar, click **New** > **Create Risk Mitigation Template Category**. You will see the **Risk Mitigation Template Category** editor.
3) Enter information into each field, as required.
   - **ID**: Displays the identification number of the risk mitigation template category.
   - **Name**: Enter the name of the risk mitigation template category.
   - **Description**: Provide a meaningful description about the risk mitigation template category.
4) Click **OK** to save the risk mitigation template category or **Cancel** if you do not want to save it.

If you delete a risk mitigation template category, the risk mitigation template category and all subordinate risk mitigation categories will be irrevocably deleted from the Alfabet database. Please note that the risk mitigation templates assigned to the deleted risk mitigation template category can be reassigned to a new risk mitigation template category via the **Risk Mitigation Templates** functionality.

To define subordinate risk mitigation template categories, navigate to the object profile of the new risk mitigation template and define further sub-categories in the *Risk Mitigation Template Sub-Categories Page View*.

To add risk mitigation templates to the selected risk mitigation template category, navigate to the object profile of the new risk mitigation template and define further sub-categories in the *Risk Mitigation Templates Page View*.

Finding Risk Mitigation Templates Not Assigned to a Risk Mitigation Template Category

In Alfabet, each risk mitigation template should be assigned to a risk mitigation template category. However, if a risk mitigation template category containing risk mitigation templates is deleted, the risk mitigation templates will remain in the Alfabet database but will no longer be assigned to a risk mitigation template category and not displayed in an explorer. The **Unassigned Risk Management Templates** page view allows you to view all risk mitigation templates that are not assigned to a risk mitigation template category and reassign them to an existing risk mitigation template category.

To assign a risk mitigation template to a risk mitigation template category:

1) Click the **Risk Mitigation Templates** icon  at the top of the explorer and click the **Unassigned Risk Mitigation Templates**.
2) In the table, select the risk mitigation template that you want to assign to a risk mitigation template category and click the **Assign** button.
3) The object selector opens. Enter search criteria, as needed, and click **Search**.
4) Select a risk mitigation template category and click **OK**. The risk mitigation template is assigned to the selected risk mitigation template category.

Creating a Risk Mitigation Template for the Risk Mitigation Template Category

A risk mitigation template captures a preconfigured definition of a risk mitigation including the name of the risk mitigation, the target date when the risk mitigation should be implemented, and a number used to prioritize the risk mitigation. A risk mitigation template can be created for a specific threat, threat group, or risk mitigation template category. Each risk mitigation template can be assigned to one risk mitigation template category.

To navigate to the **Risk Mitigation Template** page view:

1) Click the **Risk Mitigation Templates** icon  at the top of the explorer and click **Root Risk Mitigation Template Categories**.
2) Select the risk mitigation template category and navigate to the object profile to add a risk mitigation template to the risk mitigation template category.
3) Open the **Risk Mitigation Template** page view.

To create a new risk mitigation template:

1) In the toolbar, click **New** > **Create New Risk Mitigation Template**. The **Risk Mitigation Template** editor opens.
2) Enter information into each field, as required.
   - **ID**: Displays the identification number of the risk mitigation template.
   - **Name**: Enter a name for the risk mitigation template.
   - **Short Name**: Enter a short name for the risk mitigation template.
   - **Description**: Provide a meaningful description about the risk mitigation template.
3) Click **OK** to save the risk mitigation template or **Cancel** if you do not want to save it.

Chapter 13: Configuring Technical Environment Definitions

The **Technical Environments** functionality allows you to configure technical environment definitions and their technical environment definition items.

A technical environment describes a specific area of the technology required to support the development, maintenance, operation, or testing of an object. Technical environments are typically documented for testing and operations management purposes and may include, for example, Development Tools or Testing Tools. Technical environments could also be implemented to document the applications or components that are developed in-house. In this case, a number of design, development or testing tools (components) may be required that must be evaluated in the context of the application/component they are used for.

A technical environment definition can be configured for any of the following object classes: Application, Component, Deployment, Device, Standard Platform, and Vendor Product. If the object class supports object class stereotypes, any configured stereotypes may be configured to support the definition of technical environments. The technical environment definition also determines the user profiles that have Read/Write permission to specify the technical architecture elements relevant to the selected object.

Each technical environment definition can have multiple technical environment definition items and an unlimited number of levels in the technical environment definition item hierarchy. An Alfabet query is typically defined for a technical environment item in order to find the set of architecture elements that could be specified for an aspect of the technical environment.

For example, the technical environment Testing Tools could include the subordinate technical environment items Functional Testing, Performance & Load Test, etc. Each technical environment item would include an Alfabet query that would find the relevant set of testing tools that a user can choose from in order to define the object's testing environment. For example, an Alfabet query associated with the technical environment definition item Functional Testing might be configured to find the applications relevant to the functional testing of devices.

The **Technical Environment Definition** explorer displays the object classes Technical Environment Definition and Technical Environment Definition Item. Click any technical environment definition or technical environment definition item in the explorer tree to see its configured object cockpit or object profile.

The following information is available:

- Configuration Requirements for the Technical Environments Capability
- Creating a Technical Environment Definition
- Creating Technical Environment Definition Items
- Creating Subordinate Technical Environment Definition Items

Configuration Requirements for the Technical Environments Capability

In order for users to be able to work with the **Technical Environments** capability, the following configuration must be carried out:

1) One or more technical environment definitions must be created in the **Technical Environment Definition** explorer. The technical environment definition determines the object classes that the technical environment is applicable to as well as the user profiles that have permission to define the technical environment in the **Technical Environments** page views available in the relevant object profiles. For more information about defining the valid user profiles and object classes for a technical environment definition, see the section Creating a Technical Environment Definition.

2) For each technical environment definition, one or more technical environment definition items must be created in the Technical Environment Definition Items Page View. A technical environment item may also have subordinate technical environment items which are created in the **Subordinate Technical Environment Definition Items** page view.

An Alfabet query must first be created in the Alfabet Query Builder, available in the configuration tool Alfabet Expand, and then copied and pasted into the **Technical Environment Item** editor. For more information about how to access the Alfabet Query Builder and configure an Alfabet query, see the section *Defining Queries for Alfabet Configurations* in the reference manual *Configuring Alfabet with Alfabet Expand*.

3) Once the configuration is complete, users with the relevant user profiles can define the technical environments in the Technical Environment Page View. The technical environment definitions relevant to the object class can be selected for definition in a **Technical Environment** filter in the view. The results of the Alfabet query can be viewed for each technical environment definition item for which an Alfabet query has been designed. Only one architecture element in the Alfabet query results can be selected per technical environment item.

For example, the technical environment Development Tools includes compilers. The technical environment item Compilers thus might have two subordinate technical environment items, each configured to find a different set of compilers. One subordinate technical environment Microsoft Compilers might have an Alfabet query that searches the set of results for all components with Microsoft® in the name and another subordinate technical environment Borland Compilers has an Alfabet query that searches for all components with Borland® in the name.

Creating a Technical Environment Definition

To create a new technical environment definition:

1) Click the **Technical Environment Definition**  icon at the top of the explorer.
2) In the toolbar, click **New** > **Create New Technical Environment Definition**.
3) You will see the **Technical Environment Definition** editor. Enter information into each field, as required.

   Basic Data tab:

- **Name**: Enter a unique name for the technical environment definition.
- **Apply to Classes**: Select one or more object classes that the technical environment definition is valid for. The technical environment definition items that are configured for this technical environment definition will be available in the Technical Environment Page View for objects in the selected object classes.

Please note the following regarding the configuration of the technical environment definition for an object class stereotype:

1) The object class stereotype must be added to the enumeration items for the `EnvironmentClassesDef` enumeration.
2) Create a technical environment definition for the object class stereotype in the **Technical Environment Definition** explorer.

3) Define one or more technical environment definition items each providing a query to find objects in the database. For example:

```
ALFABET_QUERY_500 FIND Component

    InnerJoin Domain AS dom1 ON dom1.REFSTR =
    Component.Domain

    InnerJoin Domain AS dom2 ON dom2.REFSTR =
    dom1.BelongsTo

WHERE dom2.Name = 'Communication'

SHOW Component.Name Component.Version

SORT Component.Name Component.Version
```

4) Define the technical environments in the **Technical Environments** page view. Please note that if the **Technical Environments** page view should not be displayed for a specific object class stereotype, then the page view should be hidden in the relevant view scheme definition in the configuration tool Alfabet Expand. For more information about hiding page views, see the section *Defining the Visibility of Page Views/Configured Reports Available at the Root Node of an Explorer* in the reference manual *Configuring Alfabet with Alfabet Expand*.

To remove object classes from the list of object classes in the **Apply to Classes** field, the solution designer must modify the `EnvironmentClassesDef` enumeration. For more information, see the section *Defining Protected and Custom Enumerations* in the reference manual *Configuring Alfabet with Alfabet Expand*.

- **Description**: Enter a meaningful description that will clarify the purpose of the technical environment definition.

User Profiles tab:

- **Activate for User Profiles**: Select the user profiles that permit access to this technical environment definition. A user logged on with the selected user profile will have Read/Write permission to the technical environment for an object in the object classes selected in the **Apply to Classes** field. If no user profile is selected, the technical environment will be available to all users with permission to access the object.

5) Click **OK** to save the technical environment or **Cancel** if you do not want to save it.

If you delete a technical environment definition, it will be irrevocably deleted from the Alfabet database. All technical environments associated with architecture artifacts (for example, applications or components) will also be deleted. If you delete a technical environment definition that has subordinate technical environment definition items, all technical environment definition items will also be deleted.

Creating Technical Environment Definition Items

This page view allows you to define the technical environment definition items for the selected technical environment definition. The technical environment definition item contains an Alfabet query that searches for a set of objects that are permissible to use to define the technical environment for the specific architecture elements (for example, applications or components).

Any technical environment definition item may have subordinate technical environment definition items, each with a different Alfabet query that allows you to further refine the set of values available to define an artifact's environment.

For example, the technical environment Development Tools includes compilers. The technical environment item Compliers thus might have two subordinate technical environment items, each configured to find a different set of compilers. One subordinate technical environment Microsoft Compilers might have an Alfabet query that searches the set of results for all components with Microsoft® in the name and another subordinate technical environment Borland Compilers has an Alfabet query that searches for all components with Borland® in the name.

Please keep the following in mind when defining a technical environment definition item:

- The technical environment item is only accessible to users logged in with the user profile(s) defined in the **User Profiles** tab in the **Technical Environment Definition** editor of the ascendant technical environment definition. The Alfabet query configured for the technical environment definition item finds the available objects from that a user may select.
- An Alfabet query must first be created in the Alfabet Query Builder, available in the configuration tool Alfabet Expand, and then copied and pasted into the **Technical Environment Definition Item** editor. For more information about how to access the Alfabet Query Builder and configure an Alfabet query, see the section *Defining Queries for Alfabet Configurations* in the reference manual *Configuring Alfabet with Alfabet Expand*.
- The results of the Alfabet query configured for a technical environment definition item will be available in the *Technical Environment Page View*s for objects of the relevant object classes. For more information about defining the technical environment for a selected object, see the *Technical Environment Page View*.

The table displays the technical environment definition items assigned to the selected technical environment definition. Each column in the table is defined below:

To create a new technical environment definition item:

1) In the toolbar, click **New** > **Create New Technical Environment Definition Item**. Alternatively, you can select an existing technical environment definition item that you want to base the new technical environment definition item on and click **Create New Technical Environment Definition Item as Copy**. The **Technical Environment Definition Item** editor opens.
2) Enter information into each field, as required:

   Basic Data tab:

   - **Name**: Enter a unique name for the technical environment definition item.
   - **Description**: Enter a meaningful description that will clarify the purpose of the technical environment definition item.

   Query tab:

   - **Query**: Paste a query in the field. The query searches for a set of objects that may be defined as elements in the technical environment definition.

     For more information about how to configure a query for Alfabet, see the section *Defining Queries for Alfabet Configurations* in the reference manual *Configuring Alfabet with Alfabet Expand*.

3) Click **OK** to save the technical environment definition item.

If you delete a technical environment definition item, it will be irrevocably deleted from the Alfabet database. Any subordinate technical environment definition items defined for the selected technical environment definition item will also be deleted.

Creating Subordinate Technical Environment Definition Items

The **Subordinate Technical Environment Definition Items** page view allows you to define the technical environment definition items for the selected technical environment definition item. The technical environment definition item contains an Alfabet query that searches for a set of objects that users require in order to define the technical environment definition for the relevant object class.

To create a new subordinate technical environment item, select the parent technical environment item in the Technical Environment Definition Items Page View and click the **Navigate** button. Click **Subordinate Technical Environment Definition Items** and proceed as described in the Technical Environment Definition Items Page View.

Chapter 14: Creating an Index for the Full-Text Search

The **Full-Text Search** functionality allows usser to execute a full-text search on the basis of a search index for predefined search group. A search group defines the scope of the full-text search. Search groups limit search to objects of defined object classes and to values of defined object class properties of these object classes. The full-text search can be conducted in the primary language as well as in all secondary languages for which a culture has been configured by your enterprise. To conduct a full-text search, the user first selects a search group, then enters a search term and clicks **Search** to view the search results. The information displayed in the search results for each object matching the search condition depends on the configuration of the search group.

The full-text search functionality must be preconfigured by a solution designer. The solution designer defines the search groups and the basic parameters for the generation of full-text search indexes. For more information, see the section *Configuring the Full-Text Search Capability* in the reference manual *Configuring Alfabet with Alfabet Expand*.

Full-text search in a search group can only be conducted if a search index is available for the search group in the language the user is currently using to render the user interface. There are different methods for generation of the full-text search indexes:

- An administrative user can create and update the full-text search indexes for globally defined search groups in the administrative **Full-Text Search** functionality on the Alfabet user interface. The functionality allows creation and update of indexes for all supported languages to be performed. The manual index creation and update procedure is described in the following.
- Via the **Job Schedule** functionality on the Alfabet user interface, administrative users can create job schedules for the automatic creation and update of search indexes for globally defined full-text search groups in defined time intervals. A job schedule triggers the creation and update of the search index of a single full-text search group. Please note that the job schedule functionality currently supports only definition of full-text search indexes for the language English (United States). For information about how to schedule full-text search index update via a job schedule, see Creating a Job Schedule for the Generation of a Full-Text Search Index.
- Via the executable `FullTextSearchUtil.exe`, system administrators can configure a batch process to regularly create and update search indexes for globally defined full-text search groups. The command line tool can create and update indexes for all supported languages. It can either update an index for a defined search group or for all globally defined full-text search groups. For more information about the configuration of the batch process, see the section *Updating Indexes with the FullTextSearchUtil.exe* in the reference manual *System Administration*.

Next to the global **Full-Text Search** functionality, an object-centric full-text search is available to search within objects related to a base object. This search is conducted on the **Full-Text Search** page view in the object profile of the base object. The object-centric full-text search requires one index for each combination of base object, search group and supported language. The index is therefore not centrally created and updated by an administrator but created directly by the user conducting the search prior to entering the search term. This limits the search index creation to the indexes for the base objects for that a search is actually conducted. Object-centric search groups are neither available in the global **Full-Text Search** functionality nor in the administrative **Full-Text Search** functionality.

The user interface must be set to display the relevant primary or secondary language when the search index is created.

To create or update the index, select the configured search group for which you want to generate an index in the **Search Group** field and click the **Create Index** button. The index for the combination of the selected search group and the current language is updated and displays the date and time of the most recent update.

Once an index is created, it should be updated regularly to reflect changes made to the Alfabet inventory.

Chapter 15: Managing Alfabet Objects in the Context of Changes in the Enterprise

The **Administration Desktops** functionality allows you to manage objects in the context of structural change. The following information is available:

Managing Alfabet Objects in the Context of Organizational Changes

The **Organization Admin** functionality allows you to address modifications in the ownership of objects that result from changes to the organizational structure.

Relevant object classes or references that can be reassigned include:

- Sub-Organizations
- Business Processes
- Market Products
- ICT Objects
- Owned Demands
- Affected Demands
- Affecting Projects
- Owned Value Nodes
- Affecting Value Nodes
- Skill Requests
- Skill Offers
- Roles
- Operational Business Supports
- Strategic Business Supports
- Tactical Business Supports
- Solution Business Supports

Custom properties of the type Reference that reference the object class `OrgaUnit` can also be reassigned to an organization in the **Organization Admin** functionality. Custom properties of the type `Reference` as well as the functionality required to implement such properties in the **Organization Admin** functionality must be configured by Software AG Support. For more information, please contact Software AG Support.

1) To add the relevant organizations that are affected by the organizational changes, Select **New** > **Add Columns to Matrix**.
2) The object selector opens. Enter search parameters, as needed, and click **Search**.
3) Select the organizations that you want to remove objects from as well as the organizations that you want to move the objects to. Click **OK** to add the selected organizations to the desktop.
4) Select the type of reference that you want to reassign in the **Reference Types** field and click **Update**. All relevant objects assigned to the organizations displayed in the **Organization Admin** functionality are shown.

5) To move objects from one organization to another organization, select the organization from which you want to remove the objects and click **Action** > **Move All Items in Selected Column**. An editor will open showing the organizations displayed in the **Organization Admin** functionality that you can move the objects to. Select the relevant organization and click **OK**. The objects will be displayed in the column of the organization that you have moved them to.

6) Continue this process for all relevant references displayed in the **Reference Types** field.

To remove an object that has been erroneously reassigned to an organization, select the object in the desktop and click the **Detach**  button.

Managing Alfabet Objects in the Context of Changes to the Application Landscape

The **Application Group Admin** functionality allows you to address modifications in the assignment of objects that result from changes to the application group structure.

Relevant object classes or references that can be reassigned include Sub-Groups and Applications.

Custom properties of the type Reference that reference the object class Application Group can also be reassigned to an application group in the **Application Group Admin** functionality. Custom properties of the type `Reference` as well as the functionality required to implement such properties in the **Application Group Admin** functionality must be configured by Software AG Support. For more information, please contact Software AG Support.

1) To add the relevant application groups that are affected by the application group changes, Select **New** > **Add Columns to Matrix**.

2) The object selector opens. Enter search parameters, as needed, and click **Search**.

3) Select the application groups that you want to remove objects from as well as the application groups that you want to move the objects to. Click **OK** to add the selected application groups to the desktop.

4) Select the type of reference that you want to reassign in the **Reference Types** field and click **Update**. All relevant objects assigned to the application groups displayed in the **Application Group Admin** functionality are shown.

5) To move objects from one application group to another application group, select the application group from which you want to remove the objects and click **Action** > **Move All Items in Selected Column**. An editor will open showing the application groups displayed in the **Application Group Admin** functionality that you can move the objects to. Select the relevant application group and click **OK**. The objects will be displayed in the column of the application group that you have moved them to.

6) Continue this process for all relevant references displayed in the **Reference Types** field.

To remove an object that has been erroneously reassigned to an application group, select the object in the desktop and click the **Detach**  button.

Managing Alfabet Objects in the Context of Changes to Component Groups

The **Component Group Admin** functionality allows you to address modifications in the assignment of objects that result from changes to the component group structure.

Relevant object classes or references that can be reassigned include Sub-Groups and Components.

Custom properties of the type Reference that reference the object class Component Group can also be reassigned to an component group in the **Component Group Admin** functionality. Custom properties of the type `Reference` as well as the functionality required to implement such properties in the **Component Group Admin** functionality must be configured by Software AG Support. For more information, please contact Software AG Support.

1) To add the relevant component groups that are affected by the component group changes, Select **New** > **Add Columns to Matrix**.
2) The object selector opens. Enter search parameters, as needed, and click **Search**.
3) Select the component groups that you want to remove objects from as well as the component groups that you want to move the objects to. Click **OK** to add the selected component groups to the desktop.
4) Select the type of reference that you want to reassign in the **Reference Types** field and click **Update**. All relevant objects assigned to the component groups displayed in the **Component Group Admin** functionality are shown.
5) To move objects from one component group to another component group, select the component group from which you want to remove the objects and click **Action** > **Move All Items in Selected Column**. An editor will open showing the component groups displayed in the **Component Group Admin** functionality that you can move the objects to. Select the relevant component group and click **OK**. The objects will be displayed in the column of the component group that you have moved them to.
6) Continue this process for all relevant references displayed in the **Reference Types** field.

To remove an object that has been erroneously reassigned to an component group, select the object in the desktop and click the **Detach** button.

Managing Alfabet Objects in the Context of Domain Changes

The **Domain Admin** functionality allows you to address modifications in the ownership of objects that result from changes to the domain structure.

Relevant object classes or references that can be reassigned include:

- Functional Module
- Business Process
- Business Function
- Business Object
- ICT Object
- Application
- Component
- Standard Platform
- Vendor Product
- Domain Structure

Custom properties of the type Reference that reference the object class DOMAIN can also be reassigned to an domain in the **Domain Admin** functionality. Custom properties of the type `Reference` as well as the functionality required to implement such properties in the **Domain Admin** functionality must be configured by Software AG Support. For more information, please contact Software AG Support.

1) To add the relevant domains that are affected by the domain changes, Select **New** > **Add Columns to Matrix**.
2) The object selector opens. Enter search parameters, as needed, and click **Search**.
3) Select the domains that you want to remove objects from as well as the domains that you want to move the objects to. Click **OK** to add the selected domains to the desktop.
4) Select the type of reference that you want to reassign in the **Reference Types** field and click **Update**. All relevant objects assigned to the domains displayed in the **Domain Admin** functionality are shown.
5) To move objects from one domain to another domain, select the domain from which you want to remove the objects and click **Action** > **Move All Items in Selected Column**. An editor will open showing the domains displayed in the **Domain Admin** functionality that you can move the objects to. Select the relevant domain and click **OK**. The objects will be displayed in the column of the domain that you have moved them to.
6) Continue this process for all relevant references displayed in the **Reference Types** field.

To remove an object that has been erroneously reassigned to an domain, select the object in the desktop and click the **Detach** button.

Managing Alfabet Objects in the Context of Changes to ICT Object Categories

The **ICT Object Category Admin** functionality allows you to address modifications in the assignment of objects that result from changes to the ICT object category structure.

Relevant object classes or references that can be reassigned include Sub-Categories and ICT Objects.

Custom properties of the type Reference that reference the object class ICT Object Category can also be reassigned to an ICT object category in the **ICT Object Category Admin** functionality. Custom properties of the type `Reference` as well as the functionality required to implement such properties in the **ICT Object Category Admin** functionality must be configured by Software AG Support. For more information, please contact Software AG Support.

1) To add the relevant ICT object category that are affected by the ICT object category changes, Select **New** > **Add Columns to Matrix**.
2) The object selector opens. Enter search parameters, as needed, and click **Search**.
3) Select the ICT object categories that you want to remove objects from as well as the ICT object categories that you want to move the objects to. Click **OK** to add the selected ICT object categories to the desktop.

4) Select the type of reference that you want to reassign in the **Reference Types** field and click **Update**. All relevant objects assigned to the ICT object categories displayed in the **ICT Object Category Admin** functionality are shown.

5) To move objects from one ICT object category to another ICT object category, select the ICT object category from which you want to remove the objects and click **Action** > **Move All Items in Selected Column**. An editor will open showing the ICT object categories displayed in the **ICT Object Category Admin** functionality that you can move the objects to. Select the relevant ICT object category and click **OK**. The objects will be displayed in the column of the ICT object category that you have moved them to.

6) Continue this process for all relevant references displayed in the **Reference Types** field.

To remove an object that has been erroneously reassigned to an ICT object category, select the object in the desktop and click the **Detach**  button.

Chapter 16: Capturing Data with Data Capture Templates

The **Extended Data Capture Templates** functionality provides a sophisticated and comprehensive means to collect large sets of data for object classes, reference information for properties of the type `ReferenceArray`, or costs in the context of XLSX files. Data may be captured in any of the languages supported by your enterprise.

Three different types of data capture templates are available in the **Extended Data Capture Templates** functionality.

- Class-based data capture templates allow data about object classes to be captured including which object class properties, indicator types, role types, and lifecycle phases to capture. One or more data capture templates can be configured per permissible object class in order to address multiple data capture approaches for different regional units, customer segments, products, etc. The objects exported to the XLSX file may be limited by means of a stereotype definition or configured report. The class-based data capture templates can be specified to allow users to potentially create new objects, modify data for existing objects, delete objects, or allow no changes to be made to objects. A row will be added to the XLSX file for each object exported to the data capture template. A column will be added to the XLSX file for each property, indicator type, role type, and lifecycle phase to capture for the objects.

- Reference-based data capture templates allow reference information for object class properties of the type `ReferenceArray` to be captured. The reference-based data capture template allows the set of objects available for the base class as well as the set of objects available for the object that the base object references to be specified. The objects available to define the references may be found by either a stereotype definition or configured report. In the XLSX file, users may potentially create or update the objects for the base class as well as referenced class or delete the reference. A row will be added to the XLSX file for each existing instance of the `ReferenceArray` property specified in the reference-based data capture template. A column will be added to the XLSX file for the base class and the reference class of the `ReferenceArray` property.

  Please note that object class properties of the type `ReferenceArray` may also be captured via a class-based data capture template. In the class-based data capture template, however, the base object of the reference cannot be modified. This is possible in a reference-based data capture template.

- Cost-based data capture templates allow cost information to be captured for the classes **Application**, **Deployment**, or **ICT Object**. The cost-based data capture template allows request, current, or budget cost types to be captured in one or more currencies for relevant yearly cost buckets for the specified architecture class. Data capture templates can be specified to allow users to potentially create new cost objects, modify data for existing cost objects, or specify that no changes shall be made to a cost object. Furthermore, data capture templates can specify that users may change the owning object or cost type of a record. A row will be added to the XLSX file for each cost object found via the data capture template.

Users capturing the data must have a spreadsheet program installed on their computer to process the XLSX files. Each exported object/reference will constitute a row in the XLSX file. The first column will constitute an **Operations** column where the user must specify which operation to execute for the object (for example, `Create`, `Update`, `Delete`, and `No Change`). The subsequent columns represent the data to capture (such as properties, indicator types, role types, etc. specified for the data capture template). Depending on the property type to capture, users can enter a value in a cell or select a value in a drop-down list of a cell.

Upon import, all valid data captured in the XLSX file will overwrite existing data in the database. A validation process will be executed, and the total number of records attempted to be imported as well as the number that have been successfully imported and the number of invalid records that were not imported will be displayed in the **Extended Data Capture Templates** functionality. A status report in the form of an XLSX file can be

opened to understand which records were not imported and an explanation describing the error. The user can directly correct the invalid data in the status report and subsequently import the corrected data to Alfabet.

The **Extended Data Capture Templates** functionality displays all data capture templates that have been configured in your enterprise. Expand the dataset to view all data in the view. The following information is displayed.

- **Class**: Displays the object class of a class-based data capture template or the base class of a reference-based data capture template. Data capture templates specified to capture cost information will be listed under the class `BudgetValue`.
- **Data Capture Template Stereotype**: Displays **Class** if the data capture template is a class-based data capture template, **Reference** if the data capture template is a reference-based data capture template, or **Cost** if the data capture template is a cost-based data capture template.
- **Data Capture Template Name**: Displays the name of the data capture template.
- **Asynchronous**: Display a checkmark if the import/export of the data capture template has been specified to be asynchronously executed.
- **Timestamp of Upload**: Displays the timestamp of the most recent import for the data capture template.
- **Total Records**: Displays the total number of records in the imported XLSX file.
- **Processed Records**: Displays the number of records that were successfully imported to the Alfabet database.
- **Discarded Records**: Displays the number of records that were invalid and therefore not imported to the Alfabet database.

Data can only be captured for object classes and their public and protected properties if the **Enable for Data Capture Templates** attribute is set to `True` for the object class and relevant object class property in the configuration tool Alfabet Expand. The ID property is an internal object class property which cannot be manipulated in data capture templates. For details about the configuration required to capture data via the **Extended Data Capture Templates** functionality as well as the configuration requirements to export objects via configured reports or implement asynchronous data import/export, see the chapter *Configuring the Extended Data Capture Functionality* in the reference manual *Configuring Alfabet with Alfabet Expand*.

The following information is available:

- Configuring Class-Based Data Capture Templates
- Configuring a Reference-Based Data Capture Template
- Configuring a Cost-Based Data Capture Template for Applications, Deployments, and ICT Objects
- Configuring a Cost-Based Data Capture Template for Projects
- Creating a Data Capture Template Based on an Existing Data Capture Template
- Exporting and Capturing Data in the XLSX File
- Validating the XLSX File Before Import
- Importing Data from the XLSX File and Correcting Invalid Data

Configuring Class-Based Data Capture Templates

The **Extended Data Capture Templates** functionality allows class-based data capture templates to be defined to collect data about object classes including which object class properties, indicator types, role types, and lifecycle phases to capture. Object hierarchies such as an application group hierarchy or domain hierarchy can be created via the `BelongsTo` property available for such an object class.

Each data capture template describes the data to capture for one class. The class-based data capture templates can be specified to allow users to potentially create new objects, modify data for existing objects, delete objects, or specify that no changes shall be made to objects. A row will be added for each Alfabet object that is exported to the XLSX file.Information about whether a definition is mandatory or not as well as hints specified to help users to define the properties, roles, indicators, and lifecycle phases for objects will be displayed in an additional **Help** tab in the XLSX file.

The **Enable for Data Capture Templates** attribute must be set to `True` in Alfabet Expand for each object class for which a data capture template is to be configured. Data capture is only supported for artifact classes.

Additionally, the **Enable for Data Capture Templates** attribute must also be set to `True` for each object class property that shall be captured in a data capture template. If the object class property is an inherited property, than the **Enable for Data Capture Templates** attribute will be disabled. In this case, you must specify a local setting for the property that will apply to the property only in the context of this object class. To do so, expand the **Local Settings** section of the attribute window and set the **Enable for Data Capture Templates** attribute to `True`.

For more information about configuring object classes and object class properties, see the chapter *Configuring the Class Model* in the reference manual *Configuring Alfabet with Alfabet Expand*.

The objects exported to the XLSX file may be limited by means of a stereotype definition or configured report. If a configured report is implemented to find the objects for the relevant object class to export to the XLSX file, the configured report must already be available when the class-based data capture template is created. If properties of type `Reference`or `ReferenceArray` are to be captured in the class-based data capture template, then configured reports may also be implemented to limit the set of the objects that the base class objects may reference.

Please note that object class properties of the type `ReferenceArray` may be captured via a class-based data capture template, although the base object of the reference cannot be modified by the use. If users should be able to explicitly specify the base object for properties of the type `ReferenceArray`, you should create a reference-based data capture template. For more information about capturing properties of type `ReferenceArray`, see the section Configuring a Reference-Based Data Capture Template.

Configured reports will only be available in relevant editor fields in data capture template editors if the **Category** attribute of the configured report has been set to the relevant category configured for the use case `DataCaptureTemplates` and the semantic analysis of the configured report reveals that the configured report returns `REFSTR` values of the relevant object class or object classes. For more information about specifying the **Category** attribute of a configured report, see the section *Assigning a Category for Specific Functional Use to a Configured Report*in the reference manual *Configuring Alfabet with Alfabet Expand*.

If properties, role types, indicator types, etc. have been configured for an existing data capture template and have then been removed from the class model, an error message will be displayed when the **Data Capture Template - Class** editor is opened explaining that the relevant role type,

indicator type, etc. has been removed from the data capture template due to the change made to the class model.

A row will be added to the XLSX file for each object exported to the data capture template. A column will be added to the XLSX file for each property, indicator type, role type, and lifecycle phase to capture for the objects. Please note that the ID property is an internal object class property which cannot be manipulated in data capture templates.

To create a class-based data capture template:

1) In the **Extended Data Capture Templates** functionality, click **New** > **Create New Data Capture Template**.
2) In the **Stereotype Selector**, select **Class-Based Data Capture Template**. The **Data Capture Template - Class** editor opens.
3) In the **Basic Data** tab, specify basic information about the class-based data capture template. Define the following fields as needed:

- **Name**: Mandatory: Provide a meaningful name for the data capture template.
- **File Name Base**: Provide a string that will be used as the first part of the file name of the file that is generated based on the data capture template. The string will be appended with a GUID to assure that the file name is unique.
- **Release Status**: Select a predefined release status for the data capture template.

  Release status definitions must be configured in the XML **ReleaseStatusDefs** for the classes `ALFA_DATACAPTURETEMPLATE:Class`. For more information about specifying release statuses, see the section *Configuring Release Status Definitions for Object Classes* in the reference manual *Configuring Alfabet with Alfabet Expand*.

- **Class**: Mandatory: Select the object class for which the data shall be collected in the data capture template. Only object classes for which the **Enable for Data Capture Templates** attribute is set to `True` will be available in the drop-down list.
- **Applicable Stereotype**: Select the object class stereotype configured for the object class specified in the **Class** field for which the data shall be collected in the data capture template. Only objects of the defined stereotype may be exported to the XLSX file. The ID prefix configured for the object class stereotype will automatically be used as the ID prefix for new objects. If a configured report is specified in the field, the report query will search for objects in the set of objects found via the **Applicable Stereotype** definition. If new objects are created in the XLSX file, the new objects will automatically be assigned the specified stereotype. If no stereotype is defined, then all relevant objects based on all stereotypes of the object class may be exported.

  Please note that if the **Enable for Data Capture Templates** attribute is set to `True` for the object class property `Stereotype`, the `Stereotype` property will not be added to the properties displayed in the **Class Properties** tab if the **Applicable Stereotype** attribute is defined. The permissible hierarchy of stereotypes for domains, features, ICT objects, organizations, service products, projects, and value nodes specified in the relevant XML objects (**DomainManager**, **FeatureManager**, **ICTObjectManager**, **OrganizationManager**, **ServiceProductManager**, **ProjectManager**, and **ValueManager**) will be validated when the data capture template is imported. If the XML object is empty or invalid, the default stereotype will be used for the exported objects.

- **Dataset Provider**: Specify the method to export the object data to the XLSX file. The objects will be displayed as records in the **Export** tab in the XLSX file. You can do one of the following:

- Select **All Records** to populate the XLSX file will all relevant objects of the class and, if defined, applicable stereotype definition. As an alternative to specifying a configured report, the set of reference properties of type `Reference` or `ReferenceArray` to be captured for an object class in the class-based data capture template can be restricted to a meaningful subset of object classes via the **Properties Details** column in the **Class Properties** tab. If multiple object classes are referenced by the property of type `Reference` or `ReferenceArray`, a multi-select combo box will be displayed in the column and the referenced classes that are relevant for the data capture can be selected. If the **Dataset Provider** attribute is set to **All Records** and the **Property Details** tab is specified, only the relevant records will be included in the generated XLSX file.

- Select a configured report to populate the XLSX file with objects found by the report query. Configured reports may be applied to find a specific set of objects that may be relevant for a regional unit, customer segment, etc.

  Report parameters may be specified for a configured report selected in the **Dataset Provider** field and applied to the data capture template when the XLSX is generated. The parameters allow the data capture template to be reused for different contexts such as different organizations or locations of data capture activity. A **Dataset Provider Parameters**  button will be displayed next to the **Dataset Provider** field if parameters have been configured for the report. Click the button to open the **Extended Data Capture - Report Parameters** editor and specify a value that shall be entered for the parameter in the text field next to the parameter name. At the time of creation of the XLSX file, the parameter value will be injected in the report and the data capture template will filter the export result based on the provided filter.

  The configured report should be of the type `Query` or `NativeSQL` and should return the set of objects to be reviewed, updated, or deleted as well as the sort order of the returned objects. For more information about how to create a configured report, see the chapter *Configuring Reports* in the reference manual *Configuring Alfabet with Alfabet Expand*.

- Select **No Records** to generate an XLSX that has no instance data exported. The XLSX file will contain columns as defined in the data capture template but no records that have been exported from the Alfabet database. Exporting a file without instance data might be used, for example, if only new objects shall be created via the XLSX file. In this case, you must ensure that **Create** is selected in the **Permitted Operations** field.

- **Description**: Enter a meaningful description that will clarify the purpose of the data capture template.

- **Sample Record Provider**: Sample data may be exported to the XLSX file to provide examples of existing data for users to understand how to create new data in the XLSX file. The sample data is exported to a **Sample Data** tab in the XLSX file where users can experiment with the data. The sample data will not be reimported to Alfabet and therefore the Alfabet database will not be impacted by changes made to the sample data. Configured reports for which the **Category** attribute has been set to ADIF will be listed in the **Sample Record Provider** field. Select a configured report to populate a set of sample data in the XLSX file.

- **Permitted Operations**: Mandatory: Specify the operations that shall be permissible in the XLSX file. Each row in the **Operations** column will have a drop-down list that allows the user to specify which operation shall be applied to the record. The follow operations are permissible:

  - **Create**: Allows a new object to be created. Users must define the new object in an empty row in the XLSX file and define the relevant columns as needed.

- **Update**: Allows one or more properties, role types, indicator types, or lifecycle phases to be modified for the object.
- **Delete**: Allows the entire object to be deleted.
- **No Change**: Allows no modification to be made to the object. Please note that if **No Change** is not selected, users must modify the record in some way.

- **Primary Language**: Mandatory: Select the primary language to be used to generate the information in the XLSX file. If a primary language other than English is specified for the data capture template, the data in the XLSX file will be displayed in the specified language if a translation is available for the string. Date property values will be generated in the Excel file based on the format configured for the specified primary language. A validation of the formatting of the data loaded with the XLSX file will be executed for the primary language specified in the data capture template.
- **Other Supported Languages**: Select the additional languages in which the data shall be captured. A column will be added to the XLSX file to capture the relevant data for each language selected in the **Other Supported Languages** field. The column header will display < **Property Name** > <(**Language Culture Name**)>

  Please consider the following regarding the language settings for the data capture template:

  - A language column will only be created in the XSLX file for custom and protected properties for which the **Enable Data Translation** attribute has been set to `Manual` or `ManualAndAutomated` for the object class property in Alfabet. With the exception of **Name** and **Description** properties, the **Enable Data Translation** attribute is predefined as `None` for most protected properties. For more information, see the section *Configuring the Translation of Object Data* in the reference manual *Configuring Alfabet with Alfabet Expand*.
  - If both the **Primary Language** and **Other Supported Languages** fields are specified, than the language of the primary culture configured for the solution interface must be selected in one of the fields. If no value is imported for the language of the primary culture, an error will occur. For more information about configuring the primary culture, see the section *Specifying the Cultures Relevant to Your Enterprise* in the reference manual *Configuring Alfabet with Alfabet Expand*.

- **Capture Lifecycle**: Select the checkbox if lifecycle information shall be captured for the objects targeted by the data capture. If the **Capture Lifecycle** checkbox is selected, the lifecycle phases specified for the relevant object will be automatically displayed in the **Lifecycle** tab for the object class/object class stereotype specified in the data capture template. If the **Capture Lifecycle** checkbox is not selected, no lifecycle phases will be displayed in the **Lifecycle** tab.
- **Max. Number of Rows**: Enter the maximum number of records that may be imported via the XLSX file. The number of valid records starting with the first valid record in the XLSX file will be imported.
- **User Loading the Data is the Responsible User**: Select the checkbox if the user importing the XLSX file shall be the responsible user for objects created via the data upload. If the checkbox is selected, the `ResponsibleUser` property will not be displayed in the **Class Properties** tab. If the **User Loading the Data is the Responsible User** attribute is not selected, the `ResponsibleUser` property must be explicitly selected in the **Class Properties** tab if the authorized user shall be available in the XLSX file.

- **Status Report Scope**: Specify which type of records shall be displayed in the status report (XLSX file) that provides information about the results of the import. The following is possible:
  - **Total Records**: Select if all processed and discarded records shall be included in the status report. The discarded records displayed in the status report can be corrected and reimported to the Alfabet database.
  - **Processed Records**: Select if only records that were successfully imported shall be included in the status report.
  - **Discarded Records**: Select if only records that failed the validation process and were discarded during the import shall be included in the status report. The discarded records displayed in the status report can be corrected and reimported to the Alfabet database.
- **Validate**: Select one or more validation scenarios that shall be executed in addition to the standard validation rules when the XLSX file is imported to Alfabet. This field will only be displayed if a relevant object class is selected in the **Class** field. Please note the following:
  - If you select either the classes **Domain** and **Business Process** in the **Class** field, a validation rule for the Level ID property can be selected. Select if the defined value must be valid for the domains/business processes based on the **Level ID** values for its parent-child relationships.
  - If you select either the classes **Information Flow**, **Business Support**, and **Tactical Business Support** in the **Class** field, a validation rule for date properties can be selected. Please note the following:
    - If the start date of the information flow is earlier than the start date of the source or target object of the information flow, then the information flow will not be created or updated. If the end date of the information flow is later than the end date of the source or target object of the information flow, then the information flow will not be created or updated.
    - If the start date of the business supports is earlier than the start date of either of the referenced objects, then the business support will not be created or updated. If the end date of the business supports is later than the end date of either of the referenced objects, then the business support will not be created or updated.

4) The **Class Properties** tab displays a row for each object class property that has been enabled for data capture templates. Please note that in order to prevent the ID of objects being erroneously changed, the **ID** property cannot be enabled as a class property.

The **Enable for Data Capture Templates** attribute must also be set to `True` for each object class property that shall be captured in a data capture template. If the object class property is an inherited property, then the **Enable for Data Capture Templates** attribute will be disabled. In this case, you must specify a local setting for the property that will apply to the property only in the context of this object class. To do so, expand the **Local Settings** section of the attribute window and set the **Enable for Data Capture Templates** attribute to `True`. For more information about configuring object classes and object class properties, see the chapter *Configuring the Class Model* in the reference manual *Configuring Alfabet with Alfabet Expand*.

The following is possible:

- Drag the horizontal scrollbar at the bottom of the dataset in order to view all columns in the **Class Properties** tab.
- Define the following columns in the **Class Properties** tab:

- **Name**: Displays the name of the data capture template record.
- **Caption**: If necessary, change the caption that shall be displayed as the column header in the XLSX file.
- **Mandatory**: Set a checkmark to specify that the property is mandatory and must be defined in the XLSX file for the record to be imported to the Alfabet database. Any object class property that is specified as mandatory in the class model will automatically have the checkmark set in the **Mandatory** column. This can be changed, as needed. If the property is set as mandatory, the checkmark in the **Include** column will also be set. The **Mandatory** column in the **Help** tab of the XLSX file will be set to **True** for a mandatory property.
- **Property**: Displays the name of the property targeted by the record.
- **Include**: Set a checkmark to specify that the property shall be included in the XLSX file. A mandatory property must be included in the XLSX file.
- **Property Type**: Displays the property type of the property.
- **Property Details**: For properties of type `Reference` or `ReferenceArray`: Displays the referenced property. If multiple object classes are referenced by the property of type `Reference` or `ReferenceArray`, a multi-select combo box will be displayed in the column and the referenced classes that are relevant for the data capture can be selected. If the **All Records** in the **Basic Data** tab and the **Property Details** tab is specified, only the relevant records will be included in the generated XLSX file.
- **Reference Class Filter Type**: For properties of type `Reference` or `ReferenceArray`: Select **All** to export all objects of the class specified in the **Property Details** column, select **Stereotype** to export all objects of the object class stereotype that you will specify in the **Reference Class Stereotype** column, or select **Report** to specify all objects found by the configured report that you will specify in the **Reference Class Filter** column.
- **Reference Class Filter**: For properties of type `Reference` or `ReferenceArray`: If you have selected **Report** in the **Reference Class Filter Type** column, select the configured report that shall find the objects targeted by the reference. The objects found by the configured report can be selected in a drop-down list in the XLSX file.

  The configured report should be of the type `Query` or `NativeSQL` and should return the set of objects that are targeted by the reference as well as the sort order of the returned objects. For more information about how to create a configured report, see the chapter *Configuring Reports* in the reference manual *Configuring Alfabet with Alfabet Expand*.

- **Reference Class Stereotype**: For properties of type `Reference` or `ReferenceArray`: If you have selected **Stereotype** in the **Reference Class Filter Type** column, select the object class stereotype of the objects targeted by the reference. The objects based on the specified stereotype will be sorted lexicographically and can be selected in a drop-down list in the XLSX file.
- **Editor Hint**: Enter text to assist the user in defining the property in the XLSX file. If no hint is defined, the value specified for the **Hint** attribute of the property in the class model will be displayed. The hints will be available in the **Help** tab of the XLSX file.
- **Allow Delta Objects**: For properties of type `Reference` or `ReferenceArray`: Set a checkmark to allow new referenced objects to be specified in the XLSX file. The new referenced objects are objects that did not exist at the time that the XLSX file was generated but have in the meantime been added to the Alfabet database. The user must type in the

exact name of the referenced object. If the name is correctly entered for the referenced object, the reference will be created between the base object and the new reference object upon import of the XLSX file.

- **Replace on Data Load**: For properties of type `ReferenceArray`: Role types where the **Person Plurality** and **Organization Plurality** attributes are set to `True` for the role type configuration can be updated via the data capture template. Set a checkmark to specify that existing all existing roles of the property shall be deleted and only one role will be added upon import. If a checkmark is not set, existing records will not be deleted in the Alfabet database and the new roles will be added to the existing roles upon import.

- To change the sequence of the properties, select a row and click either the **Move Up**  or **Move Down**  button in the toolbar of the dataset. The sequence specified in the data capture template will also be the sequence of the properties in the XLSX file.

- To set the checkmark in the **Include** column for all properties, select the **Select All** button above the dataset. To clear the checkmark in the **Include** column for all properties, select the **Exclude All** button above the dataset.

5) The **Role Type** tab displays a row for each permissible role type available for the object class/object class stereotype targeted by the data capture template. All role types assigned to the relevant class in the **Class Configuration** functionality will be displayed in the **Role Types** tab. For more information about configuring role types, see the section *Configuring Role Types to Define Roles in the Responsibilities Page View* in the reference manual *Configuring Evaluation and Reference Data in Alfabet*.

Please note the following:

- The data capture template will include a row for each possible role type available for the targeted class/stereotype. Therefore, all role types that may reference a person as well as all role types that may reference an organization stereotype will be displayed in the **Role Type** tab.

- Persons based on the stereotypes `User` and `Contact` may be specified for a role type in the context of the data capture template.

- Role types may be configured so that only one person or organization may be specified for a role for an object or multiple persons or organizations may be specified for a role for an object. If multiple persons are allowed, the (**Person Plurality** attribute for the role type will be set to `False` and if multiple organizations are allowed, the **Organization Plurality** attribute for the role type will be set to `False`. Please note the following:

  - If the **Organization Plurality** attribute for a role type is set to `False` for a role and the role is updated via the imported XLSX file, the old role definition will be deleted and a new role with the specified organization will be created. If the **Organization Plurality** attribute is set to `True`, a new role will be created for each specified organization.

  - If the **Person Plurality** attribute for a role type is set to `False` for a role and the role is updated via the imported XLSX file, the old role definition will be deleted and a new role with the specified person will be created. If the **Person Plurality** attribute is set to `True`, a new role will be created for each specified person.

- Roles can be deleted for objects in the context of a data capture template for roles based on role types where the **Person Plurality** and **Organization Plurality** attributes are set to `False` in the role type configuration.

The following is possible:

- Drag the horizontal scrollbar at the bottom of the dataset in order to view all columns in the **Role Type** tab.
- Define the following columns in the **Role Type** tab:
  - **Name**: Displays the name of the data capture template record. The name is a concatenation of <Class>:<Stereotype>:<Role Type>. In the XLSX file, the role types will be listed in the sequence <Class>:<Stereotype>:<Role Type>. The sequence can be changed via the **Move Up** or **Move Down** buttons, as needed.
  - **Caption**: If necessary, change the caption that shall be displayed as the column header in the XLSX file. The following default syntax may be displayed:
    - Role types for the class `Person` where the person is based on the stereotype `User`: `Person:User`:<Name of Role Type>
    - Role types for the class `Person` where the person is based on the stereotype `Contact`: `Person:Contact`:<Name of Role Type>
    - Role types for the class `OrgaUnit`: `OrgaUnit`:<Name of Role Type>
  - **Mandatory**: Set a checkmark to specify that the role type is mandatory and must be defined in the XLSX in order for the record to be imported to the Alfabet database. This can be changed, as needed. If a checkmark is set in the **Mandatory** column, the checkmark in the **Include** column will automatically be set. The **Mandatory** column in the **Help** tab of the XLSX file will be set to **True** for a mandatory role type.
  - **Include**: Set a checkmark to specify that the role type shall be included in the XLSX file. A mandatory role type must be included in the XLSX file.
  - **Property Details**: Displays the details of the role type. This will be a concatenation of `OrgaUnit` or `Person` and the name of the role type.
  - **Reference Class Filter**:Select the configured report that shall find the specific subset of role types that the objects targeted by the data capture template may be assigned to. The role types found by the configured report shall be available in the drop-down list to define the referenced object in the XLSX file.

    The configured report should be of the type `Query` or `NativeSQL` and should return the set of persons or organizations that may be defined for the role type as well as the sort order of the returned persons/organizations. For more information about how to create a configured report, see the chapter *Configuring Reports* in the reference manual *Configuring Alfabet with Alfabet Expand*.

  - **Reference Class Stereotype**: Displays the stereotype of the class `OrgaUnit` or `Person` that is referenced by the role type.
  - **Editor Hint**: Enter text to assist the user in defining the role type in the XLSX file. If no hint is defined, the value specified for the **Description** attribute of the role type will be displayed. The hints will be available in the **Help** tab of the XLSX file.

- **Allow Delta Objects**: Set a checkmark to allow new referenced role types to be specified in the XLSX file. The new referenced role types are role types that did not exist at the time that the XLSX file was generated but have in the meantime been added to the Alfabet database. The user must type in the exact name of the referenced role types. If the name is correctly entered for the referenced role types, the reference will be created between the base object and the new role type upon import of the XLSX file.

- To return to the default settings, click the **Reset** button.

- To change the sequence of the role types, select a row and click either the **Move Up**  or **Move Down**  button in the toolbar of the dataset. The sequence specified in the data capture template will also be the sequence of the role types in the XLSX file.

- To set the checkmark in the **Include** column for all role types, select the **Select All** button above the dataset. To clear the checkmark in the **Include** column for all role types, select the **Exclude All** button above the dataset.

- To set the checkmark in the **Allow All Delta Objects** column for all role types, select the **Select All Delta Objects** button above the dataset. To clear the checkmark in the **Allow All Delta Objects** column for all role types, select the **Exclude All** button above the dataset.

6) The **Indicator Type** tab displays a row for each permissible indicator type available for the object class targeted by the data capture template. All relevant indicator types associated with evaluation types that are assigned to the relevant class in the **Class Configuration** functionality will be displayed in the **Indicator Type** tab. For more information about configuring indicator types, see the section *Configuring Evaluation Types* in the reference manual *Configuring Evaluation and Reference Data in Alfabet*.

Please note the following:

- Indicator types that allow a value to be explicitly defined or a value to be selected from a configured range will be included in the data capture template.

- Indicator types based on a computation rule will not be included in the data capture template.

- If the **Hide Numbers** checkbox is selected for an indicator type in the **Indicator Types** editor, then only the semantic value will be displayed in the drop-down field in the XLSX file.

- The role types and indicator types defined in the XLSX file must exist in Alfabet at the time of import. A record will not be imported to Alfabet if the defined indicator type or role type has been deleted from the Alfabet database at the time of import.

The following is possible:

- Drag the horizontal scrollbar at the bottom of the dataset in order to view all columns in the **Indicator Type** tab.

- Define the following columns in the **Indicator Type** tab:

  - **Name**: Displays the name of the data capture template record. The name is a concatenation of <Evaluation Type Name>:<Indicator Type Name>. he indicator types will be lexicographically sorted in the XLSX file according to <Evaluation Type Name>:<Indicator

Type Name>. The sequence can be changed via the **Move Up** or **Move Down** buttons in the toolbar, as needed.

- **Caption**: If necessary, change the caption that shall be displayed as the column header in the XLSX file.
- **Mandatory**: Set a checkmark to specify that the indicator type is mandatory and must be defined in the XLSX in order for the record to be imported to the Alfabet database. This can be changed, as needed. If a checkmark is set in the **Mandatory** column, the checkmark in the **Include** column will automatically be set. The **Mandatory** column in the **Help** tab of the XLSX file will be set to **True** for a mandatory indicator type.
- **Include**: Set a checkmark to specify that the indicator type shall be included in the XLSX file. A mandatory indicator type must be included in the XLSX file.
- **Editor Hint**: Enter text to assist the user in defining the indicator type in the XLSX file. If no hint is defined, the value specified for the **Description** attribute of the indicator type will be displayed. The hints will be available in the **Help** tab of the XLSX file.

- To return to the default settings, click the **Reset** button.
- To change the sequence of the indicator types, select a row and click either the **Move Up** or **Move Down** button in the toolbar of the dataset. The sequence specified in the data capture template will also be the sequence of the indicator types in the XLSX file.
- To set the checkmark in the **Include** column for all indicator types, select the **Select All** button above the dataset. To clear the checkmark in the **Include** column for all indicator types, select the **Exclude All** button above the dataset.

7) The **Lifecycle** tab displays a row for each lifecycle phase specified for the object class targeted by the data capture template and allows you to specify the lifecycle phase to include in the XLSX file.

Please consider the following:

- The **Capture Lifecycle** attribute must be set to `True` in the **Basic Data** tab to display the lifecycle phases in the **Lifecycle** tab.
- Users will be able to define a start date for each lifecycle phase included in the XLSX file and the end date of the last lifecycle phase. A user can delete a lifecycle phase for an object by selecting **Update** in the **Operations** column and leaving the lifecycle phase's start date empty when the XLSX file is imported.
- At least one lifecycle phase's start date and the lifecycle end date must be defined at the time of import.
- Lifecycle phases are configured for the relevant object class in the XML object **ObjectLifecycleManager**. For more information about configuring lifecycle phases, see the section *Configuring Lifecycle Definitions for Object Classes* in the reference manual *Configuring Alfabet with Alfabet Expand*.

Define the following columns in the **Lifecycle** tab:

- **Name**: Displays the name of the data capture template record. The name will be displayed as a column caption in the XLSX file as a concatenation of **<Lifecycle Phase> Start Date** for all

lifecycle phases except for the last lifecycle phase. The column caption for the last lifecycle phase will be **Lifecycle End Date** for the last lifecycle phase.

- **Caption**: If necessary, change the caption that shall be displayed as the column header in the XLSX file.
- **Mandatory**: Set a checkmark to specify that the lifecycle phase is mandatory and must be defined in the XLSX in order for the record to be imported to the Alfabet database. This can be changed, as needed. If a checkmark is set in the **Mandatory** column, the checkmark in the **Include** column will automatically be set. The **Mandatory** column in the **Help** tab of the XLSX file will be set to **True** for a mandatory lifecycle phase.
- **Include**: Set a checkmark to specify that the lifecycle phase shall be included in the XLSX file. A mandatory lifecycle phase must be included in the XLSX file.
- **Editor Hint**: Enter text to assist the user in defining the lifecycle phase in the XLSX file. If no hint is defined, the value specified for the XML attribute `Hint` in the XML object **ObjectLifecycleManager** will be displayed. Please note that the default hint for the lifecycle end date is: The end date of the final lifecycle phase defined for the record. The hints will be available in the **Help** tab of the XLSX file.

8) The **Import/Export Asynchronously** tab allows the asynchronous import and export of data via data capture templates to be specified for the import/export of large sets of data. If asynchronous execution is activated, it will be implemented when the following actions are triggered:

- **Create MS Excel File from Data Capture Template**
- **Download Data Capture Template**
- **Import MS Excel File for Data Capture Template**

To activate asynchronous import and export for a data capture template, select the **Import/Export Asynchronously** checkbox. In the dataset, select the folder in the **Internal Document Selector** that the export file shall be created in. If the **Import/Export Asynchronously** checkbox is selected, the event feedback message will be displayed to the user triggering the import or export of the data capture template.

Please note that the following must be configured to implement asynchronous execution:

- The data capture template functionality requires a running Alfabet Server to be connected to the same database as the Alfabet Web Application providing the **Extended Data Capture Template** functionality on the user interface.
- The Alfabet Web Application must be configured to use the Alfabet Server for execution of ADIF jobs. The **User Server to Execute ADIF Jobs** attribute in the server alias of the Alfabet Web Application must be set to `True` for the server alias.
- The RESTful services of the Alfabet Web Application must be activated and configured as described in the chapter *Activating the Alfabet RESTful API on Server Side* of the reference manual *Alfabet RESTful API*. In the server alias of the Alfabet Web Application, the **Has ADIFAPIInvocation Access** option must be activated in the **API Access Options** attribute.
- A user must be configured for execution of RESTful service calls via self-reflective events. For more information, see *Configuring a User to Execute Self-Reflective Events* in the reference manual *Configuring Alfabet with Alfabet Expand*. The user

must have the **Has ADIFAPIInvocation Access** option activated in the **API Access Options** attribute in the **User** editor.

After the asynchronous import/export has been triggered, the user importing or exporting the data can continue to work in Alfabet. A slide-in message window providing information about the success of the execution of the asynchronous process as well as a link to the folder in the **Internal Document Selector** where the file has been saved will be displayed when the results for the asynchronously executed process are available. The message will be available for the user triggering the asynchronous import/export. For more information regarding user feedback messages, see the section *Using the Event Feedback Messages*.

The **Download Data Capture Template** option in the **New** menu will be enabled if the selected data capture template is defined to be executed asynchronously and the associated XSLX file has been generated and is ready to be downloaded. Otherwise, the **Download Data Capture Template** option will be disabled.

9) Click the **OK** button to save the data capture template configuration and close the editor.

Configuring a Reference-Based Data Capture Template

In order to specify the details of the referenced objects for the data import, a reference-based data capture template can be configured for each property of the type `ReferenceArray` specified for the class-based data capture template. The **Data Capture Template - Reference Arrays** editor is used to specify the relationship between the object of the base class and the object that it references. Class filters based on configured reports or stereotype definition may be specified to constrain the relevant objects for the relationship definition. The configuration of the **Allow Delta Objects** attribute in the data capture template allows objects to be used to build references in the XLSX file even though they did not exist in the database at the time the XLSX file was created.

A row will be added to the XLSX file for each existing instance of the `ReferenceArray` property specified in the reference-based data capture template. A column will be added to the XLSX file for the base class and the reference class of the `ReferenceArray` property.

To create a reference-based data capture template:

1) In the **Extended Data Capture Templates** functionality, click **New** > **Create New Data Capture Template**.
2) In the **Stereotype Selector**, select **Reference-Based Data Capture Template**. The **Data Capture Template - Reference Arrays** editor opens.
3) In the **Data Capture Template - Reference Arrays** editor, define the following fields:
   - **Name**: Mandatory: Provide a meaningful name for the data capture template.
   - **File Name Base**: Provide a string that will be used as the first part of the file name of the file that is generated based on the data capture template. The string will be appended with a GUID to assure that the file name is unique.
   - **Release Status**: Select a predefined release status for the data capture template.

Release status definitions must be configured in the XML **ReleaseStatusDefs** for the classes `ALFA_DATACAPTURETEMPLATE:Class`. For more information about specifying release statuses, see the section *Configuring Release Status Definitions for Object Classes* in the reference manual *Configuring Alfabet with Alfabet Expand*.

- **Base Class**: Mandatory: Select the base class of the reference that will be captured in the data capture template. Only object classes for which the **Enable for Data Capture Templates** attribute is set to `True` will be available in the drop-down list.
- **Reference Array Property**: Mandatory: Select the property of the type `ReferenceArray` for which data shall be captured. Only object classes for which the **Enable for Data Capture Templates** attribute is set to `True` will be available in the drop-down list.
- **Reference Class**: Mandatory: Select the class that is the target of the reference that will be captured in the data capture template. Only properties of the type `ReferenceArray` for which the **Enable for Data Capture Templates** attribute is set to `True` will be available in the drop-down list.
- **Base Class Filter Type**: Select **All** to export all objects of the class specified in the **Base Class** field, select **Stereotype** to export all objects of the object class stereotype that you will specify in the **Base Class Stereotype** field, or select **Report** to specify all objects found by the configured report that you will specify in the **Base Class Filter** field.
- **Base Class Filter**: If you have selected **Report** in the **Base Class Filter Type** field, select the configured report that shall find the base objects of the reference. The objects found by the configured report can be selected in a drop-down list in the XLSX file.

  The configured report should be of the type `Query` or `NativeSQL` and should return the set of objects that are the base object of the reference as well as the sort order of the returned objects. For more information about how to create a configured report, see the chapter *Configuring Reports* in the reference manual *Configuring Alfabet with Alfabet Expand*.

- **Base Class Stereotype**: If you have selected **Stereotype** in the **Base Class Filter Type** field, select the object class stereotype of the base objects of the reference. The objects based on the specified stereotype will be sorted lexicographically and can be selected in a drop-down list in the XLSX file.
- **Reference Class Filter Type**: Select **All** to export all objects of the class specified in the **Reference Class** field, select **Stereotype** to export all objects of the object class stereotype that you will specify in the **Reference Class Stereotype** field, or select **Report** to specify all objects found by the configured report that you will specify in the **Reference Class Filter** field.
- **Reference Class Filter**: If you have selected **Report** in the **Reference Class Filter Type** field, select the configured report that shall find the objects targeted by the reference. The objects found by the configured report can be selected in a drop-down list in the XLSX file. The entire reference array will be validated against the configured report specified in the **Reference Class Filter** attribute. The first record matching the reference class filter is displayed in the reference array property in the XLSX file.

  The configured report should be of the type `Query` or `NativeSQL` and should return the set of objects that are targeted by the reference as well as the sort order of the returned objects. For more information about how to create a configured report, see the chapter *Configuring Reports* in the reference manual *Configuring Alfabet with Alfabet Expand*.

- **Reference Class Stereotype**: If you have selected **Stereotype** in the **Reference Class Filter Type** field, select the object class stereotype of the objects targeted by the reference. The objects based on the specified stereotype will be sorted lexicographically and can be selected in a drop-down list in the XLSX file.

- **Base Class Column Label**: If necessary, change the caption that shall be displayed as the column header for the base class of the reference in the XLSX file.
- **Base Class Column Edit Hint**: Enter text to assist the user in defining the base class of the reference in the XLSX file. The hints will be available in the **Help** tab of the XLSX file.
- **Reference Class Column Label**: If necessary, change the caption that shall be displayed as the column header for the referenced class of the reference in the XLSX file.
- **Reference Class Column Edit Hint**: Enter text to assist the user in defining the referenced class of the reference in the XLSX file. If no hint is defined, the value specified for the **Hint** attribute of the property that is references will be displayed. The hints will be available in the **Help** tab of the XLSX file.
- **Sample Record Provider**: Sample data may be exported to the XLSX file to provide examples of existing data for users to understand how to create new data in the XLSX file. The sample data is exported to a **Sample Data** tab in the XLSX file where users can experiment with the data. The sample data will not be reimported to Alfabet and therefore the Alfabet database will not be impacted by changes made to the sample data. Select a configured report to populate a set of sample data in the XLSX file.
- **Permitted Operations**: Mandatory: Specify the operations that shall be permissible in the XLSX file. Each row in the **Operations** column will have a drop-down list that allows the user to specify which operation shall be applied to the record. The following operations are permissible.
  - **Update**: Allows one or more properties, role types, indicator types, or lifecycle phases to be modified for the object.
  - **Delete**: Allows the entire object to be deleted.
  - **No Change**: Allows no modification to be made to the object. Please note that if **No Change** is not selected, users must modify the record in some way.
- **Primary Language**: Mandatory: Select the primary language to be used to generate the information in the XLSX file. If a primary language other than English is specified for the data capture template, the data in the XLSX file will be displayed in the specified language if a translation is available for the string. Date property values will be generated in the Excel file based on the format configured for the specified primary language. A validation of the formatting of the data loaded with the XLSX file will be executed for the primary language specified in the data capture template. For more information about configuring the primary culture, see the section *Specifying the Cultures Relevant to Your Enterprise* in the reference manual *Configuring Alfabet with Alfabet Expand*.
- **Max. Number of Rows**: Enter the maximum number of records that may be imported via the XLSX file. The number of valid records starting with the first valid record in the XLSX file will be imported.
- **Allow Delta Objects**: Set a checkmark to allow objects not listed in the drop-down lists for the base class or referenced class to be specified in the XLSX file. This allows objects to be specified for the base class or referenced class that did not exist at the time that the XLSX file was generated but have in the meantime been added to the Alfabet database. The user must type in the exact name of the object as it is defined in Alfabet in order for the reference to be valid. If the name is correctly entered for the object, the reference will be created upon import of the XLSX file. If the name is not spelled correctly, the reference cannot be created, and the record will be discarded as an invalid reference value.
- **Status Report Scope**: Specify which type of records shall be displayed in the status report (XLSX file) that provides information about the results of the import. The following is possible:

- **Total Records**: Select if all processed and discarded records shall be included in the status report. The discarded records displayed in the status report can be corrected and reimported to the Alfabet database.
- **Processed Records**: Select if only records that were successfully imported shall be included in the status report.
- **Discarded Records**: Select if only records that failed the validation process and were discarded during the import shall be included in the status report. The discarded records displayed in the status report can be corrected and reimported to the Alfabet database.

4) Click the **OK** button to save the data capture template configuration and close the editor.

Configuring a Cost-Based Data Capture Template for Applications, Deployments, and ICT Objects

In order to specify the details of cost objects for the data import, one or more cost-based data capture templates can be configured for the class `BudgetValue`. The **Data Capture Template - Cost** editor allows you to specify the capture of request, current, or budget cost types in one or more currencies for the architecture classes **Application**, **Deployment**, or **ICT Object**. A data capture template can only be specified for one architecture class for which cost information shall be collected.

Data capture templates can be specified to allow users to potentially create new cost objects, modify data for existing cost objects, or specify that no changes shall be made to a cost object. A row will be added for each cost object that is exported to the XLSX file. Information about whether a definition is mandatory or not as well as hints specified to help users to define the properties for objects will be displayed in an additional **Help** tab in the XLSX file. When importing the XLSX file, the entire row will be imported in the Alfabet database, whereby each owner object/cost type/annual bucket is imported as one record. For example, if four annual buckets are specified for an owner object and cost type, then four records will be added to the Alfabet database. Any previously existing records for the same combination of owner, cost type, and annual bucket will be removed.

Please consider the following regarding the configuration:

- The data capture template is configured for the class `BudgetValue`. The **Enable for Data Capture Template** attribute must be set to `True` for the class `BudgetValue` as well as for the following properties of the class: `Value, Owner, MonetaryCodeId, Year, Currency, MonetaryType` in the configuration tool Alfabet Expand. For more information about configuring object classes and object class properties, see the chapter *Configuring the Class Model* in the reference manual *Configuring Alfabet with Alfabet Expand*
- A configured report must be specified to return the respective applications, deployments, or ICT objects as well as the cost types for which the data shall be collected. The report must return the references for `Owner` and `CostType`. The owner may be any of the following classes: `Application, Deployment, ICT Object`.
- The cost definition type (**Request, Current,** or **Budget** values) must be specified to export existing cost information to the XLSX file. The respective value can be modified as need and then re-imported to the Alfabet database. However, it is also possible to specify a cost definition type for the export and a different cost definition type for the import to the Alfabet database. This allows you to use existing values for one cost definition type as a basis for new values for a different cost definition type. For example, you could configure that the existing **Request** values that have been defined for applications are exported to the XLSX file. Using the arithmetic operations available in the XLSX file, you could calculate 80% of the exported Request values to capture new values that could then be imported for

the cost definition type **Budget**. In this way, you can easily and efficiently specify new budget values without needing to manually enter them in the user interface.

- Data may be captured for yearly buckets. The yearly buckets may be found via a configured report or specified as a time period.
- Data capture templates can specify that users may change the owning object or cost type of a record. In this case, a drop-down list would be displayed for the cells in the `Owner` column of the XLSX file allowing the owner object to be changed for a respective record and/or in the `CostType` column of the XLSX file allowing the cost type to be changed for a respective record. The objects found for the drop-down list can be limited via either a configured report or a stereotype definition.

To configure a data capture template to capture cost information:

1) In the **Extended Data Capture Templates** functionality, click **New** > **Create New Data Capture Template**.
2) In the **Stereotype Selector**, select **Cost-Based Data Capture Template**. The **Data Capture Template - Class** editor opens.
3) In the **Basic Data** tab, specify basic information about the cost-based data capture template. Define the following fields as needed:
   - **Name**: Mandatory: Provide a meaningful name for the data capture template.
   - **File Name Base**: Provide a string that will be used as the first part of the file name of the file that is generated based on the data capture template. The string will be appended with a GUID to assure that the file name is unique.
   - **Release Status**: Select a predefined release status for the data capture template.

     Release status definitions must be configured in the XML **ReleaseStatusDefs** for the classes `ALFA_DATACAPTURETEMPLATE:Class`. For more information about specifying release statuses, see the section *Configuring Release Status Definitions for Object Classes* in the reference manual *Configuring Alfabet with Alfabet Expand*.

   - **Class**: Mandatory: The class `BudgetValue` is displayed per default in the field.
   - **Description**: Enter a meaningful description that will clarify the purpose of the data capture template.
   - **Dataset Provider**:
     - Select **All Records** to populate the XLSX file will all relevant objects of the class. Specify the method to export the object data to the XLSX file. The objects will be displayed as records in the **Export** tab in the XLSX file.
     - Select **No Records** to generate an XLSX that has no instance data exported. The XLSX file will contain columns as defined in the data capture template but no records that have been exported from the Alfabet database Select to.
     - Select a configured report. The configured report must return a report providing the references for `Owner` and `CostType`. The owner may be any of the following classes: `Application, Deployment, ICT Object`. A  button will be displayed next to the **Report Provider** field to open an editor for specification of a parameter value. Enter the parameter value in the text field next to the parameter name. At the time of creation of the XLSX file, the parameter value will be injected in the report and the data capture template will filter the export result

based on the provided filter. The objects will be displayed as records in the **Export** tab in the XLSX file.

The configured report should be of the type `Query` or `NativeSQL` and should return the following information:

- The `REFSTR` of the `BudgetOwner`, which can be an application, deployment, or ICT object.
- The `REFSTR` of the `CostType`.

Please note that in native SQL queries, the first argument in the `SELECT` statement is not part of the visible result in the dataset and will be ignored in this case. It may return NULL. For more information about how to create a configured report, see the chapter *Configuring Reports* in the reference manual *Configuring Alfabet with Alfabet Expand*.

- **Sample Record Provider**: Sample data may be exported to the XLSX file to provide examples of existing data for users to understand how to create new data in the XLSX file. The sample data is exported to a **Sample Data** tab in the XLSX file where users can experiment with the data. The sample data will not be reimported to Alfabet and therefore the Alfabet database will not be impacted by changes made to the sample data. Configured reports for which the **Category** attribute has been set to ADIF will be listed in the **Sample Record Provider** field. Select a configured report to populate a set of sample data in the XLSX file.
- **Max. Number of Rows**: Enter the maximum number of records that may be imported via the XLSX file. The number of valid records starting with the first valid record in the XLSX file will be imported.
- **Status Report Scope**: Specify which type of records shall be displayed in the status report (XLSX file) that provides information about the results of the import. The following is possible:
  - **Total Records**: Select if all processed and discarded records shall be included in the status report. The discarded records displayed in the status report can be corrected and reimported to the Alfabet database.
  - **Processed Records**: Select if only records that were successfully imported shall be included in the status report.
  - **Discarded Records**: Select if only records that failed the validation process and were discarded during the import shall be included in the status report. The discarded records displayed in the status report can be corrected and reimported to the Alfabet database.
- **Currency**: Mandatory: Select one or more currencies that have been configured by your enterprise. If multiple currencies have been configured for the data capture template, a `Currency` column will be added to the XLSX file. The value shall be captured in the currency specified in the data capture template for each relevant annual cost bucket column. Please note the following:
  - The values as defined in the XLSX file will be imported to the Alfabet database. The currency of the imported values will be converted at runtime based on the exchange rate configured in the **Currency Exchange Rates** view when the relevant cost view is loaded in the Alfabet user interface. For more information about the configuration of currencies and currency exchange rates, see the section *Configuring Currencies and Currency Exchange Rates for Cost Management Capabilities* in the reference manual *Configuring Evaluation and Reference Data in Alfabet*.
  - The currencies specified for a cost type in the data capture template may differ from the currency specified for the cost type for an application, deployment, or ICT object in the

Operational Expenses Page View, where the user can specify the currency per cost type in the **Maintenance Costs** editor. Please be aware that the cost displayed in the *Operational Expenses Page View* may therefore visualize numerical values that differ from those specified in the XLSX file due to the incongruity of the currency value selected in the data capture template vs. the **Maintenance Costs** editor.

- **Cost Definition Type**: Mandatory: Specify the cost definition type (**Request**, **Current**, or **Budget** values) that shall be imported via the data capture template. Only one cost definition type may be imported per data capture template. The cost definition type represents the property `MonetaryID` in the class `BudgetValue`.

  The visibility and editability of cost definition types are configured in the XML object `CostManagerDef`. for more information, see the section *Configuring the Editability of Costs for Architecture Objects* in the reference manual *Configuring Alfabet with Alfabet Expand*.

  Please note that the value `Commited` is written to the `MonetaryID` column in the `BudgetValue` database table for the `Budget` cost definition type.

- **Export Cost Definition Type**: If necessary, specify a different cost definition type (**Request**, **Current**, or **Budget** values) to be exported than the one that shall be imported. In this case, for example, **Request** values could be exported to a data capture template and used as a base to capture **Current** values. Using functions and operations in the XLSX file, the user could efficiently modify the exported **Request** values in order to define and import **Current** values.
- **Annual Cost Buckets Provider**: Specify a configured report that returns the yearly cost buckets that shall be added as rows in the **Name** column of the **Class Properties** tab. Alternatively, a start date and end date can be specified respectively in the **Annual Cost Buckets Start Date** and **Annual Cost Buckets End Date** fields. Unnecessary years can be excluded from the export in the **Class Properties** tab.
- **Annual Cost Buckets Start Date**: Specify the start date for the yearly cost buckets listed in the **Name** column of the **Class Properties** tab.
- **Annual Cost Buckets End Date**: Specify the end date for the yearly cost buckets listed in the **Name** column of the **Class Properties** tab.
- **Permitted Operations**: Mandatory: Specify the operations that shall be permissible in the XLSX file. Each row in the **Operations** column will have a drop-down list that allows the user to specify which operation shall be applied to the record. The follow operations are permissible:
  - **Create**: Allows a new cost object to be created. Users must define the new object in an empty row in the XLSX file and define the relevant columns as needed.
  - **Update**: Allows one or more properties, role types, indicator types, or lifecycle phases to be modified for the cost object.
  - **No Change**: Allows no modification to be made to the cost object. Please note that if **No Change** is not selected, users must modify the record in some way.

4) The **Class Properties** tab displays a row for each property found by the configured report for the class `BudgetValue` as well as any annual buckets that have been specified in the **Basic Data** tab. The relevant owner object class (**Application**, **Deployment**, or **ICT Object**) must be specified for which the cost information shall be captured. The owner objects are found via the specified configured report. All relevant annual cost buckets found via the annual cost bucket specification in the **Basic Data** tab will be displayed as a row in the dataset in the **Class Properties** tab. Users can specify if a bucket shall be included in the data capture template and whether capturing data for the annual cost bucket is mandatory.

The following is possible:

- Drag the horizontal scrollbar at the bottom of the dataset in order to view all columns in the **Class Properties** tab.
- Define the following columns in the **Class Properties** tab:
  - **Name**: Displays `Owner`, `MonetaryType`, `Currency` (if more than one currency was selected), and the specified annual bucket (year).
  - **Caption**: If necessary, change the caption that shall be displayed as the column header in the XLSX file.
  - **Mandatory**: Set a checkmark to specify that the property is mandatory and must be defined in the XLSX file in order for the record to be imported to the Alfabet database. Any object class property that is specified as mandatory in the class model will automatically have the checkmark set in the **Mandatory** column. This can be changed, as needed. If the property is set as mandatory, the checkmark in the **Include** column will also be set. The **Mandatory** column in the **Help** tab of the XLSX file will be set to **True** for a mandatory property.
  - **Property**: Displays the name of the property targeted by the record.
  - **Include**: Set a checkmark to specify that the property shall be included in the XLSX file. A mandatory property must be included in the XLSX file.
  - **Property Type**: Displays the property type of the property.
  - **Property Details**: Select that architecture class (`Application`, `Deployment`, or `ICTObject`) for which the cost information shall be captured. If you need to capture cost information for multiple architecture classes, then a data capture template must be created for each class.
  - **Reference Class Filter Type**: For the property `Owner`: If a drop-down list shall be displayed for the cells in the `Owner` column of the XLSX file allowing the owner object to be changed for a respective record, select one of the options described below. For the property `MonetaryID`: If a drop-down list shall be displayed for the cells in the `CostType` column of the XLSX file allowing the cost type to be changed for a respective record, select one of the following options:
    - If no value is defined in the **Reference Class Filter Type** attribute, the drop-down list will not be available for the respective cells and users will not be able to change the respective cell value.
    - Select **All** to export all objects of the class specified in the **Property Details** column.
    - Select **Stereotype** to export all objects of the object class stereotype that you will specify in the **Reference Class Stereotype** column.
    - Select **Report** to specify all objects found by the configured report that you will specify in the **Reference Class Filter** column.
  - **Reference Class Filter**: If you have selected **Report** in the **Reference Class Filter Type** column, select the configured report that shall find the objects targeted by the reference.

    The configured report should be of the type `Query` or `NativeSQL` and should return the set of objects that are targeted by the reference as well as the sort order of the returned objects. For more information about how to create a configured

report, see the chapter *Configuring Reports* in the reference manual *Configuring Alfabet with Alfabet Expand*.

- **Reference Class Stereotype**: For properties of type `Reference` or `ReferenceArray`: If you have selected **Stereotype** in the **Reference Class Filter Type** column, select the object class stereotype of the objects targeted by the reference. The objects based on the specified stereotype will be sorted lexicographically and can be selected in a drop-down list in the XLSX file.
- **Editor Hint**: Enter text to assist the user in defining the property in the XLSX file. If no hint is defined, the value specified for the **Hint** attribute of the property in the class model will be displayed. The hints will be available in the **Help** tab of the XLSX file.

- To change the sequence of the properties, select a row and click either the **Move Up**  or **Move Down**  button in the toolbar of the dataset. The sequence specified in the data capture template will also be the sequence of the properties in the XLSX file.
- To set the checkmark in the **Include** column for all properties, select the **Select All** button above the dataset. To clear the checkmark in the **Include** column for all properties, select the **Exclude All** button above the dataset.

5) The **Import/Export Asynchronously** tab allows the asynchronous import and export of data via data capture templates to be specified for the import/export of large sets of data. If asynchronous execution is activated, it will be implemented when the following actions are triggered:

- **Create MS Excel File from Data Capture Template**
- **Download Data Capture Template**
- **Import MS Excel File for Data Capture Template**

To activate asynchronous import and export for a data capture template, select the **Import/Export Asynchronously** checkbox. In the dataset, select the folder in the **Internal Document Selector** that the export file shall be created in. If the **Import/Export Asynchronously** checkbox is selected, the event feedback message will be displayed to the user triggering the import or export of the data capture template.

Please note that the following must be configured to implement asynchronous execution:

- The data capture template functionality requires a running Alfabet Server to be connected to the same database as the Alfabet Web Application providing the **Extended Data Capture Template** functionality on the user interface.
- The Alfabet Web Application must be configured to use the Alfabet Server for execution of ADIF jobs. The **User Server to Execute ADIF Jobs** attribute in the server alias of the Alfabet Web Application must be set to `True` for the server alias.
- The RESTful services of the Alfabet Web Application must be activated and configured as described in the chapter *Activating the Alfabet RESTful API on Server Side* of the reference manual *Alfabet RESTful API*. In the server alias of the Alfabet Web Application, the **Has ADIFAPIInvocation Access** option must be activated in the **API Access Options** attribute.
- A user must be configured for execution of RESTful service calls via self-reflective events. For more information, see *Configuring a User to Execute Self-Reflective Events* in the reference manual *Configuring Alfabet with Alfabet Expand*. The user

must have the **Has ADIFAPIInvocation Access** option activated in the **API Access Options** attribute in the **User** editor.

After the asynchronous import/export has been triggered, the user importing or exporting the data can continue to work in Alfabet. A slide-in message window providing information about the success of the execution of the asynchronous process as well as a link to the folder in the **Internal Document Selector** where the file has been saved will be displayed when the results for the asynchronously executed process are available. The message will be available for the user triggering the asynchronous import/export. For more information regarding user feedback messages, see the section *Using the Event Feedback Messages*.

The **Download Data Capture Template** option in the **New** menu will be enabled if the selected data capture template is defined to be executed asynchronously and the associated XSLX file has been generated and is ready to be downloaded. Otherwise, the **Download Data Capture Template** option will be disabled.

6) Click the **OK** button to save the data capture template configuration and close the editor.

Configuring a Cost-Based Data Capture Template for Projects

You can configure a data capture template to capture cost information for projects or any stereotype configured for the class `Project`. A data capture template can be configured to capture business case, cost accrual, or cashout planning for one or more fiscal years for a specified set of projects and cost types. Multiple data capture templates can be configured.

New projects may be created without a parent project in the context of the XLSX file. For new projects created in the context of a data capture template, the validation of a parent project stereotype as specified in the configuration of the XML object **ProjectManager** will be skipped if the project in the data capture template does not specify a parent project.

Only projects for which the `Type` property is set to `Project` will be exported. The project types `Scenario`, `Solution`, `Obsolete`, and `Baseline` are not supported in data capture templates.

If changes are made to an existing data capture template (such as changes to mandatory fields), a new data capture template should be created.

Please consider the following regarding the configuration:

- To capture business case definitions or cost accrual data for projects: The **Enable for Data Capture Template** attribute must be set to `True` for the class `BudgetValue` and its properties `Value`, `Owner`, `MonetaryCodeId`, `Year`, `Currency`, `MonetaryType`.
- To capture cashout planning data for projects, the **Enable for Data Capture Template** attribute must be set to `True` for the class `CashoutValue` and its properties `Value`, `Owner`, `MonetaryCodeId`, `Year`, `Currency`, `MonetaryType`. For more information about configuring object classes and object class properties, see the chapter *Configuring the Class Model* in the reference manual *Configuring Alfabet with Alfabet Expand*.
- A configured report must be specified to return the respective projects. The configured report should be of the type `Query` or `NativeSQL` and should find objects of the class `Project`. In Alfabet queries the project should be the `FIND` class and in native SQL queries the first argument of the `SELECT` statement must be the `REFSTR` of the project.

To create a data capture template for project costs

1) In the **Extended Data Capture Templates** functionality, click **New** > **Create New Data Capture Template**.
2) In the **Stereotype Selector**, select **Cost-Based Data Capture Template**. The **Data Capture Template - Class** editor opens.
3) In the **Basic Data** tab, specify basic information like name, file name, release status, data set provider, currency, etc. as described in the section Configuring a Cost-Based Data Capture Template for Applications, Deployments, and ICT Objects.
4) You can specify a data capture template to capture either the business case, cost accrual, or cashout planning for a specified set of projects.

- **Business Case**: To capture the business case definition for one or more fiscal years for a specified set of projects and cost types, define the following:
  - **Project Cost Definition Type**: Select **Business Case**. The **Class** attribute is set to `BudgetValue`.
  - **Projects to Capture Costs**: Select a configured report that specifies which projects should be exported to the `Project` column in the XLSX file. For example, the drop-down field could show all projects that have no business case costs defined.
  - **Cost Type**: Specify whether cost types or income types shall be captured for the business case.
  - **Class Properties** tab: Displays a row for each fiscal year represented for all relevant projects. Select each relevant fiscal year to capture the business cases of projects.

    If you the **Data Provider** field is set to **All Records**, the option **Leaf-Level Cost Types** will be available in the **Reference Class Filter Type** field. If **Leaf-Level Cost Type** is selected, the generated XLSX file will include the actual cost types if costs are defined. Please note however that cost types that are not leaf-level cannot be selected.

- **Cost Accrual**: To capture cost accrual definition for one or more specified fiscal years for a specified set of projects and cost types, define the following:
  - **Project Cost Definition Type**: Select **Cost Accrual**. The **Class** attribute is set to `BudgetValue`.
  - **Projects to Capture Costs**: Select a configured report that specifies which projects should be exported to the `Project` column in the XLSX file.
  - **Cost Definition Type for Import**: Specify whether request, current, or budget values shall be captured and imported via the XLSX file. Only one cost definition type may be imported per data capture template. The cost definition type represents the property `MonetaryID` in the class `BudgetValue`.

    The visibility and editability of cost definition types are configured in the XML object `CostManagerDef`. for more information, see the section *Configuring the Editability of Costs for Architecture Objects* in the reference manual *Configuring Alfabet with Alfabet Expand*.

    Please note that the value `Commited` is written to the `MonetaryID` column in the `BudgetValue` database table for the `Budget` cost definition type.

- **Export Cost Definition Type**: If necessary, specify a different cost definition type (**Request**, **Current**, or **Budget** values) to be exported than the one that shall be imported. In this case, for example, **Request** values could be exported to a data capture template and used as a base to capture **Current** values. Using functions and operations in the XLSX file, the user could efficiently modify the exported **Request** values in order to define and import **Current** values. If no value is defined, then the value defined in the **Cost Definition Type** field will be used per default.

- **Cashout Planning**: To capture cashout planning data per month for a specified fiscal year for a specified set of projects and cost types:

  - **Project Cost Definition Type**: Mandatory: Select **Cashout Planning**. The **Class** attribute is set to `CashoutPlanning`.

  - **Dataset Provider**:

    - Select **All Records** to populate the XLSX file with all relevant objects of the class. Specify the method to export the object data to the XLSX file. The objects will be displayed as records in the **Export** tab in the XLSX file.

    - Select **No Records** to generate an XLSX that has no instance data exported. The XLSX file will contain columns as defined in the data capture template but no records that have been exported from the Alfabet database Select to.

    - Select a configured report. Select a configured report to export the projects and cost types as records to the XLSX file. The configured report must return a report providing the references for `Project` and `CostType`. Only projects for which the `Type` property is set to `Project` will be exported.

  - **Projects to Capture Costs**: Select a configured report that specifies which projects should be exported to the `Project` column in the XLSX file. For example, the drop-down field could show all projects that have no business case costs defined.

  - **Cost Definition Type for Import**: Specify whether request, current, or budget values shall be captured and imported via the XLSX file.

  - **Export Cost Definition Type**: If projects are exported via the configured report in the **Dataset Provider** field, the cost values for the fiscal year are exported as either request, current, or budget values. Specify whether request, current, or budget values shall be exported to the XLSX file. If no value is selected, the same value specified in the **Cost Definition Type for Import** field will be used.

  - **Fiscal Year**: Select the fiscal year for the data capture template. The months of the fiscal years will be displayed in the **Class Properties** tab. The calculation of the fiscal year is dependent on the configuration in the XML object **CostManagerDef**. For more information, see the section *Configuring the Fiscal Year for Cost Reporting in Your Enterprise* in the reference manual *Configuring Alfabet with Alfabet Expand*.

  - **Class Properties** tab: Displays a row for each month in the specified fiscal year. Select each relevant month in the fiscal year to capture the cashout planning of projects.

5) Click the **OK** button to save the data capture template configuration and close the editor.

Creating a Data Capture Template Based on an Existing Data Capture Template

You can create a copy of the data capture template via **New** > **Create Data Capture Template as Copy**. The fields defined in the data capture template will be copied to the new data capture template. Please note the following:

- For a class-based data capture template, the **Class** field will be copied and cannot be edited but all other aspects of the data capture template may be modified, as needed.
- For a reference-based data capture template, the **Base Class** field will be copied and cannot be edited but all other aspects of the data capture template may be modified, as needed.

Exporting and Capturing Data in the XLSX File

Once the data capture template has been defined, the data can be exported to the XLSX file. If a primary language other than English has been specified for the data capture template, the data in the XLSX file will be displayed in the specified language if a translation is available for the string.

To export the data, select the relevant data capture template in the table and click **New** > **Create MS Excel File from Data Capture Template**. In the **Download File** editor, click the **Download** button and save the XLSX file to your local drive or network. If asynchronous import/export is specified in the **Import/Export Asynchronously** tab in the **Data Capture Template - Class** editor, the user importing or exporting the data can continue to work in Alfabet. A slide-in message window providing information about the success of the execution of the asynchronous process as well as a link to the folder in the **Internal Document Selector** where the file has been saved will be displayed when the results for the asynchronously executed process are available. For more information regarding user feedback messages, see the section *Using the Event Feedback Messages*.

The **Download Data Capture Template** option in the **New** menu will be enabled if the selected data capture template is defined to be executed asynchronously and the associated XSLX file has been generated and is ready to be downloaded. Otherwise, the **Download Data Capture Template** option will be disabled.

Please note the following:

- If your company's Microsoft Office files include the feature to restrict file access via sensitivity labels, you must set the sensitivity label setting of the XLSX file to Public in order to import the file with your data.
- An error message will be displayed if you attempt to create or upload the XLSX file and the **Enable for Data Capture** attribute for a class has been changed from `True` to `False` after the data capture template was created.

To collect data in the XLSX file, open the file and define the fields as needed. Please note the following:

- The **Sample Data** tab will be available if the **Sample Record Provider** attribute is defined in the data capture template. The data displayed in the **Sample Data** tab can be defined by users in order to understand the available options or provide other users with sample data to define. The sample data will not be reimported to and therefore the Alfabet database will not be impacted by changes made to the sample data.
- The **Export** tab displays the objects targeted for data collection in the rows and the properties, role types, indicator types, and lifecycle phases to be defined for the targeted objects in the columns. The

sequence of the columns is determined by the sequence defined in the data capture template, starting with the properties followed by role types, indicator types, and lifecycle phases. Please note the following about defining the fields in the **Export** tab:

- The **Operations** column must be explicitly defined for each record that is to be imported. Click in the relevant cell in the **Operations** column to select a value from a drop-down list. You may see any of the following operations listed in the drop-down list:
  - **Create**: Select to create a new record. Define the new record in an empty row in the XLSX file and define the relevant columns as needed.
  - **Update**: Select if one or more properties, role types, indicator types, or lifecycle phases shall be modified for the record.
  - **Delete**: Select if the entire record shall be deleted. This option is not relevant for data capture templates configured to capture cost information.
  - **No Change**: Select if no modification shall be made to the record.

    If the operation is not defined for a record, the record will be invalid even if no change is made to record.

- To define a value, either select a value in the drop-down list of permissible values or enter a value directly in the relevant cell.
  - If a drop-down list is available for the cell, select a value in the drop-down list.
  - If a drop-down list is not available for the cell, enter a valid date, number, or string in the cell as required for the property being defined.
  - Reference properties that are undefined in the XLSX file imported for a data capture template will be set to NULL in the Alfabet database.
  - When defining lifecycle phases, please consider the following:
    - To import a lifecycle definition, at least one lifecycle phase's start date and the lifecycle end date must be defined. The end date of a lifecycle phase will be implicitly defined based on the start date of the previous lifecycle phase.
    - To delete a lifecycle phase for an object, select **Update** in the **Operations** column and leave the lifecycle phase's start date empty.
  - When defining cost information, please consider that the entire row will be imported in the Alfabet database, whereby each owner object/cost type/annual bucket is imported as one record. If four annual buckets are specified for a owner object and cost type, then four records will be added to the Alfabet database. Any previously existing records for the same combination of owner, cost type, and annual bucket will be removed.
- Please note the following about making changes to the columns in the **Export** tab:
  - Adding a new column: No impact.
  - Moving a column: No impact.
  - Changing the column header: The values in the column will be ignored upon import. If there are mandatory or other constraints associated with the column, the upload of the XLSX file will fail.

- Removing a column: If there are mandatory or other constraints associated with the column, the upload of the XLSX file will fail.

- The hints specified for the properties, role types, indicator types, and lifecycle phases will be displayed in the **Help** tab in the XLSX file. The value **True** will be displayed in the **Mandatory** column for any values that are mandatory. If a mandatory value is not defined, the record will not be imported to the Alfabet database.

The following validation rules will be applied upon import of the data to the Alfabet database. Any record that violates any of the validation rules will not be imported to the Alfabet database.

- The **Operations** column must be defined for each record to be imported.
- All mandatory properties must be specified for a record.
- A permissible value must be defined for a property associated with an enumeration.
- A permissible value must be defined for the object state or release status.
- A value defined for a property must comply with the property's data type and data size.
- Date properties must be defined in the format configured for the specified primary culture.
- Start and end dates must be aligned. An end date may not be before the start date.
- Start and end dates of information flows must correspond to the start and end dates of their source and target objects. If the start date of the information flow is earlier than the start date of the source or target object of the information flow, then the information flow will not be created or updated. If the end date of the information flow is later than the end date of the source or target object of the information flow, then the information flow will not be created or updated.
- Information flows have two pairs of references: **From (Source) Owner** (`FromOwner`) and **To (Source) Owner** (`ToOwner`), and **From (Source)** (`From`) and **To (Source)** (`To`). The properties **From (Source) Owner** (`FromOwner`)/ **To (Source) Owner** (`ToOwner`) capture the information flow on a higher level such as information flows between applications, components, and peripherals and the properties **From (Source)** (`From`)/ **To (Source)** (`To`) capture the information flow on a subordinate level such as between local components and devices and applications, components, and peripherals. Please note the following:
  - If the properties **From (Source) Owner** (`FromOwner`) and **To (Source) Owner** (`ToOwner`) are defined in the XLSX, the properties **From (Source)** (`From`)/ **To (Source)** (`To`) will be automatically filled with the corresponding values of the properties **From (Source) Owner** / **To (Source) Owner** upon import to the Alfabet database.
  - If the properties **From (Source) Owner** (`FromOwner`) and **From (Source)** (`From`) are defined in the XLSX, the compatibility of the values will be validated so that either the `FromOwner` property is the same as the `From` property, or the `FromOwner` property is the same as the `From.Owner` property (the parent of the source object).
  - If the properties **To (Target) Owner** (`ToOwner`) and **To (Target)** (`To`) are defined in the XLSX, the compatibility of the values will be validated so that either the `ToOwner` property is the same as the `To` property, or the `ToOwner` property is the same as the `To.Owner` property (the parent of the target object).

- Start and end dates of business supports must correspond to the start and end dates of the their referenced objects. If the start date of the business supports is earlier than the start date of either of the referenced objects, then the business support will not be created or updated. If the end date of the business supports is later than the end date of either of the referenced objects, then the business support will not be created or updated.

- The permissible hierarchy of stereotypes for domains, features, ICT objects, organizations, service products, projects, and value nodes will be validated when imported via extended data capture templates. The permissible hierarchy of stereotypes for each class is in the respective XML objects (**DomainManager, FeatureManager, ICTObjectManager, OrganizationManager, ServiceProductManager, ProjectManager, and ValueManager**). For more information, see the relevant section in the chapter *Configuring Alfabet Functionalities Implemented in the Solution Environment* in the reference manual *Configuring Alfabet with Alfabet Expand*.

- The references created via the `BelongsTo` and `ConsistsOf` properties for objects in object hierarchies (such as application group, organization, or domain hierarchies) must be consistent. Image properties of a class (such as `LevelID`, `Name` in the case of the class `Domain` or `BusinessProcess`) must bet correctly specified in a reference property column such as `BelongsTo`.

- If a property of type `Reference` or `ReferenceArray` is defined, the referenced object may be based on an object stereotype. If this is the case, the object must be based on a permissible stereotype.

- Role definitions must have a valid person or organization specified. If the referenced organization is based on a stereotype, the organization must be based on a permissible stereotype.

- Indicators defined for indicator types with a range definition must comply with the range definition.

- The role types and indicator types defined in the XLSX file must exist in Alfabet at the time of import. A record will not be imported to Alfabet if the defined indicator type or role type has been deleted from the Alfabet database at the time of import.

- Lifecycle definitions are specified via the start date of the lifecycle phase. At least one lifecycle phase's start date and the lifecycle end date must be defined at the time of import. The end date of a lifecycle phase will be implicitly defined based on the start date of the previous lifecycle phase.

- A validation mechanism ensures that duplicate records are not imported in duplicate to the Alfabet database. For example, rows that have the same combination of object, cost type, monetary type, and year for architecture costs or the same combination of project, cost type and year for business case costs will not be imported. If a record with the same combination of values is specified in the XLSX file, the duplicate record will not be imported and an error message indicating that the record is a duplicate will be displayed in the XLSX file that is generated via the **Download Import Status Report** functionality.

- A validation mechanism checks for cyclic references for class-based data capture templates that have references to the same class included in the definition. If a cyclic reference exists in the data being imported, an error message describing the cyclic reference will be displayed in the XLSX file that is generated via the **Download Import Status Report** functionality.

- A validation mechanism ensures that references specified for a record in the XLSX file to a non-existing object will not be imported and will be discarded with an appropriate error message. If the reference object no longer exists in the Alfabet database, an error message

describing the invalid reference will be displayed in the XLSX file that is generated via the **Download Import Status Report** functionality.

Validating the XLSX File Before Import

You can first validate the correctness of the intended data upload and be informed about all validation rule violations in an XLSX file. This allows you to understand and correct multiple validation rule violations before actually uploading the data. The validation checks the correctness of the specification of the operation, mandatory fields, DateTime attributes, data size, stereotype definitions, enums, indicators, information flows, and references.

To import the XLSX file, click **Import** > **Validate MS Excel File for Data Capture Template**. In the **Upload Files** editor, click the **Choose Files** button and select the XLSX file form your local drive or network. Click **Upload** to import the file. In the next window, click the **Download** button and save the XLSX file to your local drive or network.

If asynchronous import/export is specified in the **Import/Export Asynchronously** tab in the **Data Capture Template - Class** editor, the user importing or exporting the data can continue to work in Alfabet. A slide-in message window providing information about the success of the execution of the asynchronous process as well as a link to the folder in the **Internal Document Selector** where the file has been saved will be displayed when the results for the asynchronously executed process are available. For more information regarding user feedback messages, see the section *Using the Event Feedback Messages*.

The **Download Data Capture Template** option in the **New** menu will be enabled if the selected data capture template is defined to be executed asynchronously and the associated XSLX file has been generated and is ready to be downloaded. Otherwise, the **Download Data Capture Template** option will be disabled.

You can open the file to review violations of the validation rules. An X will be displayed in the **Error** column. Error messages will be displayed in the subsequent columns depending on the type of error that has occurred. The errors can be corrected directly in the XLSX file. Once the errors have been corrected, you can attempt to import the MS Excel file as described in the section Importing Data from the XLSX File and Correcting Invalid Data.

The following validation rules will be applied upon import of the data to the Alfabet database. Any record that violates any of the validation rules will not be imported to the Alfabet database.

- The **Operations** column must be defined for each record to be imported.
- All mandatory properties must be specified for a record.
- A permissible value must be defined for a property associated with an enumeration.
- A permissible value must be defined for the object state or release status.
- A value defined for a property must comply with the property's data type and data size.
- Date properties must be defined in the format configured for the specified primary culture.
- Start and end dates must be aligned. The end date may not be before the start date.
- Start and end dates of information flows must correspond to the start and end dates of their source and target objects. If the start date of the information flow is earlier than the start date of the source or target object of the information flow, then the information flow will not be created or updated. If the end date of the information flow is later than the end date of

the source or target object of the information flow, then the information flow will not be created or updated.

- Information flows have two pairs of references: **From (Source) Owner** (`FromOwner`) and **To (Source) Owner** (`ToOwner`), and **From (Source)** (`From`) and **To (Source)** (`To`). The properties **From (Source) Owner** (`FromOwner`)/ **To (Source) Owner** (`ToOwner`) capture the information flow on a higher level such as information flows between applications, components, and peripherals and the properties **From (Source)** (`From`)/ **To (Source)** (`To`) capture the information flow on a subordinate level such as between local components and devices and applications, components, and peripherals. Please note the following:
  - If the properties **From (Source) Owner** (`FromOwner`) and **To (Source) Owner** (`ToOwner`) are defined in the XLSX, the properties **From (Source)** (`From`)/ **To (Source)** (`To`) will be automatically filled with the corresponding values of the properties **From (Source) Owner** / **To (Source) Owner** upon import to the Alfabet database.
  - If the properties **From (Source) Owner** (`FromOwner`) and **From (Source)** (`From`) are defined in the XLSX, the compatibility of the values will be validated so that either the `FromOwner` property is the same as the `From` property, or the `FromOwner` property is the same as the `From.Owner` property (the parent of the source object).
  - If the properties **To (Target) Owner** (`ToOwner`) and **To (Target)** (`To`) are defined in the XLSX, the compatibility of the values will be validated so that either the `ToOwner` property is the same as the `To` property, or the `ToOwner` property is the same as the `To.Owner` property (the parent of the target object).
- Start and end dates of business supports must correspond to the start and end dates of the their referenced objects. If the start date of the business supports is earlier than the start date of either of the referenced objects, then the business support will not be created or updated. If the end date of the business supports is later than the end date of either of the referenced objects, then the business support will not be created or updated.
- The permissible hierarchy of stereotypes for domains, features, ICT objects, organizations, service products, projects, and value nodes will be validated when imported via extended data capture templates. The permissible hierarchy of stereotypes for each class is in the respective XML objects (**DomainManager, FeatureManager, ICTObjectManager, OrganizationManager, ServiceProductManager, ProjectManager, and ValueManager**). For more information, see the relevant section in the chapter *Configuring Alfabet Functionalities Implemented in the Solution Environment* in the reference manual *Configuring Alfabet with Alfabet Expand*.
- The references created via the `BelongsTo` and `ConsistsOf` properties for objects in object hierarchies (such as application group, organization, or domain hierarchies) must be consistent. Image properties of a class (such as `LevelID`, `Name` in the case of the class `Domain` or `BusinessProcess`) must bet correctly specified in a reference property column such as `BelongsTo`.
- If a property of type `Reference` or `ReferenceArray` is defined, the referenced object may be based on an object stereotype. If this is the case, the object must be based on a permissible stereotype.
- Role definitions must have a valid person or organization specified. If the referenced organization is based on a stereotype, the organization must be based on a permissible stereotype.

- Indicators defined for indicator types with a range definition must comply with the range definition.
- The role types and indicator types defined in the XLSX file must exist in Alfabet at the time of import. A record will not be imported to Alfabet if the defined indicator type or role type has been deleted from the Alfabet database at the time of import.
- Lifecycle definitions are specified via the start date of the lifecycle phase. At least one lifecycle phase's start date and the lifecycle end date must be defined at the time of import. The end date of a lifecycle phase will be implicitly defined based on the start date of the previous lifecycle phase.
- A validation mechanism ensures that duplicate records are not imported in duplicate to the Alfabet database. For example, rows that have the same combination of object, cost type, monetary type, and year for architecture costs or the same combination of project, cost type and year for business case costs will not be imported. If a record with the same combination of values is specified in the XLSX file, the duplicate record will not be imported and an error message indicating that the record is a duplicate will be displayed in the XLSX file that is generated via the **Download Import Status Report** functionality.
- A validation mechanism checks for cyclic references for class-based data capture templates that have references to the same class included in the definition. If a cyclic reference exists in the data being imported, an error message describing the cyclic reference will be displayed in the XLSX file that is generated via the **Download Import Status Report** functionality.
- A validation mechanism ensures that references specified for a record in the XLSX file to a non-existing object will not be imported and will be discarded with an appropriate error message. If the reference object no longer exists in the Alfabet database, an error message describing the invalid reference will be displayed in the XLSX file that is generated via the **Download Import Status Report** functionality.

Importing Data from the XLSX File and Correcting Invalid Data

Once the XLSX file has been completed, it can be imported to Alfabet.

The following requirements must be fulfilled in order to import objects as well as update their properties, role types, and indicator types:

- The **Enable Data Capture Template** attribute must be set to `True` for the targeted object class as well as all relevant properties in the data capture template at the time of import. An error message will be displayed when you attempt to create or upload the XLSX file if the **Enable for Data Capture** attribute for a class has been changed from `True` to `False` after the data capture template was created.
- Image properties of a class (such as `LevelID`, `Name` in the case of the class `Domain` or `BusinessProcess`) must be correctly specified in a reference property column such as `BelongsTo`. Records in the uploaded XLSX file will be rejected if the image properties are not correctly specified in the reference property column. Information explaining in which column the error has occurred will be written to the **Error Message** column of the XLSX file generated via the **Download Import Status Report** functionality.

Please note that if the data capture template was created with the Alfabet user interface rendered in a secondary language, the values specified for the data capture template will be imported for the secondary language. Therefore, you should ensure that the enterprise's primary language is included in the **Other Supported Languages** field. If no value is imported for the primary language, an error will occur and an error message will be displayed.

The following will be updated in the Alfabet database when the data is imported:

- The `LAST_UPDATE` and `LAST_UPDATE_USER` properties will be updated in the Alfabet database for all objects that are imported to Alfabet.The user that initiates the import of a data capture template will be specified as the user for the `LAST_UPDATE_USER` property for objects that are updated as well as for the `CREATION_USER` attribute for objects that created. This applies to data capture templates imported via synchronous execution and asynchronous execution if the **Use Event Queue for All Jobs** is activated in the server alias setting.
- The `ALFA_DCT_STATUS` database table stores information about which objects have been changed when a data capture template is uploaded. The table includes a reference to the `ALFA_DCT_STATUS` record that represents the data load, the type of operation (`CREATE`, `UPDATE`, `DELETE`), and a list of the `REFSTR` of the objects changed. A copy of the uploaded data capture template is stored in the **Internal Document Selector** and a reference to this document is available in the `ALFA_DCT_STATUS` record.

To import the XLSX file, click **Import** > **Import MS Excel File for Data Capture Template**. In the **Upload Files** editor, click the **Choose Files** button and select the XLSX file form your local drive or network. Click **Upload** to import the file. All valid records will be imported to the Alfabet database. Once the file has been imported, a new row will be added below the data capture template in the **Extended Data Capture Templates** functionality. Please consider the following:

- If asynchronous import/export is specified in the **Import/Export Asynchronously** tab in the **Data Capture Template - Class** editor, all valid records will be immediately imported to the Alfabet database and a message confirming the import will be displayed.
- If asynchronous import/export is specified, the user importing or exporting the data can continue to work in Alfabet. A slide-in message window providing information about the success of the execution of the asynchronous process as well as a link to the folder in the **Internal Document Selector** where the file has been saved will be displayed when the results for the asynchronously executed process are available. For more information regarding user feedback messages, see the section *Using the Event Feedback Messages*.
- If an XLSX file to capture data is uploaded and the data capture template object that the XLSX file was based on no longer exists in the Alfabet database, an informational message will be displayed to the user explaining that the data capture template no longer exists and the XLSX file cannot be uploaded to Alfabet.

Once the file has been imported, a new row will be added below the data capture template in the **Extended Data Capture Templates** functionality. The following rows will be updated in the dataset:

- **Uploaded Data**: Displays the timestamp of the most recent import for the data capture template.
- **Total Records**: Displays the total number of records in the imported XLSX file.
- **Processed Records**: Displays the number of records that were successfully imported to the Alfabet database.

- **Discarded Records**: Displays the number of records that were not successfully imported to the Alfabet database.

You can review a status report in order to understand which records were discarded as well as the reason for the error. The status report is an XLSX file with the same structure as the XLSX file that was supported for the data capture template. You can open the status report and correct the error for a record based on the information in the XLSX file. Multiple errors can be corrected in the XLSX file, which can then be re-uploaded to Alfabet.

To view the status report and correct errors:

1) Select the row with the relevant import to review based on the **Uploaded Data** column and click **Import** > **Download Import Status Report**.
2) In the **Download File** editor, click the **Download** button and save the XLSX file to your local drive or network.
3) Open the XLSX file. The **Processed** column will display **False** for all discarded records and an explanation of the error in the **Error Message** column. For an overview of potential errors, see the section Validating the XLSX File Before Import.

   Whether the file includes only records that were discarded, only records that were successfully imported, or both imported and discarded records will depend on the **Status Report Scope** attribute defined in the **Basic Data** tab of the **Class-Based Data Capture Template** editor.

4) Correct the erroneous data in the file based on the information in the **Error Message** column. An error message is displayed for only the first violation of a validation rule for a record.
5) Return to the functionality and click **Import** > **Import MS Excel File for Data Capture Template**. In the **Upload Files** editor, click the **Choose Files** button and select the XLSX file from your local drive or network. Click **Upload** to import the file.
6) The **Uploaded Data** column will display the new timestamp. Review the **Discarded Records** column. If discarded records still exist, you can repeat the procedure until all errors have been resolved.

Chapter 17: Tracking the Email Messages Sent in the Context of Alfabet Functionalities

The **Email Message Log** functionality displays all emails sent in the context of Alfabet. This includes, for example, emails sent for assignments, monitors, and workflows. etc. This ensures transparency and accountability regarding tasks and responsibilities in the IT architecture in the enterprise.

The **Email Message Log** view displays the sender as well as the users in the **CC** and **BCC** fields, the subject line, the time that the email was triggered and the execution was completed, the current status of the email execution, and any relevant error messages. If no recipient is specified in the email, for example, the email will not be sent and an error message will be logged in the **Email Message Log** functionality.

The table displays the emails matching the defined filter criteria. Define the following filters and click the **Update** button:

- **Message Recipient**: Specify the user who received the emails. This field is mandatory.
- **Status**: Specify the status of the emails.
- **Sent After**: Specify a date to display all emails sent after the specified date. This field is mandatory.
- **Sent Before**: Specify a date to display all emails sent before the specified date. This field is mandatory.
- **Subject**: Enter text displayed in the subject line of the emails. Depending on your configuration, you may be able to use the asterisk * as a wildcard.

The number of records matching the filter criteria is displayed in the first row of the dataset. The following columns are displayed:

Please note that the information about the email subject and message display the actual content of the email that has been sent. Therefore, the text will not be translated if the language is different than the current language used to render the Alfabet user interface.

- **From**: The email address of the user who has triggered the email.

  If no email address is defined for the user triggering the email, a failover address defined in the configuration of the Alfabet Server processing the email will be used as the sender address. If all emails have the same sender address, the Alfabet Server has been configured to use a fixed sender email address instead of the email address of the user triggering the email.

- **To**: The email address of the email recipient.
- **CC**: The email addresses of the recipients that are defined in the **CC:** field of the email. Recipients are defined in the **CC:** field in emails sent out via the discussion capability, for example.
- **BCC**: The email addresses of the recipients that are defined in the **BCC:** field of the email.
- **Subject**: The subject line of the email.
- **Occurrence Time**: The time when the email was triggered by a user in the Alfabet user interface.
- **Execution Time**: The time when sending of the email was started by an Alfabet Server.
- **Completion Time**: The time when sending of the email was completed.
- **Status**: The status of the processing of the email.
- **Message**: If sending the email fails, the error message will be displayed.

Chapter 18: Overriding Server Configurations for Testing Purposes

The **Alfabet Configuration Overrides** functionality allows you to specify settings that override server configurations.

When testing Alfabet functionality including the sending of emails, you can configure the Alfabet Web Application to send all emails to a specified test email account, thus preventing the email from being sent to Alfabet users. This allows testing to be performed without changing the email account configuration of the Alfabet users.

System administrators can configure the server alias of the Alfabet Web Application and the Alfabet Server to ignore email account settings of Alfabet users and to send all emails to one central email account. This is done in the Alfabet Administrator. The **Alfabet Configuration Overrides** functionality allows solution designers that do not have access to the Alfabet Administrator and need to test a configuration in which emails are triggered to override the test email account and sender email address made in the server alias configuration of the Alfabet Web Application.

The following information is displayed in the view:

- **Name**: The name of the alias setting override.
- **ID**: The ID of the alias setting override.
- **Alias Name**: The name of the alias setting that the override applies to.
- **Configuration Type**: Only the value `AliasMailAccounts` to override email account settings is currently available.
- **Is Active**: The column shows a checkmark if the alias setting override is currently activated. Only one alias setting override can be activated at a time.
- **Last Update**: The date when the last change has been made to the configuration of the alias setting override.

To alter the email settings defined in the server alias configuration:

1) In the **Alfabet Configuration Overrides** explorer, click the **Alias Setting Overrides** node.
2) In the toolbar, click **New** > **Create Alias Setting Overrides**.
3) In the **Alfabet Alias Settings** editor, define the following:

   Basic Data tab:

   - **Name**: Enter a unique name for the alias setting override.
   - **Description**: Enter a description that provides information about the purpose of the alias setting override.

   Override Email Options tab:

   - **System Email Account**: If an email address is defined in this field, the specified email address will be used as the sender address for all emails even and will override the originator of the email.
   - **Failover Email Account**: If the **System Sender Email Account** field is not defined, the email address of the user that is the originator of the email will be used as the sender address. In this case, a failover email address should be defined via the **Failover Sender Email Account** field. Otherwise, sending emails will fail if the originator cannot be determined.

For information about how to specify an email address for a user, see the section *Creating a New User*.

- **Test Email Account**: If an email address is defined in this field, all emails will be sent to this email account regardless of the email address defined for the Alfabet user that the email is sent to.

4) Click **OK** to close the editor. The new alias setting override is displayed in the table.
5) To activate the alias setting override, select it in the table and click **Enable/Disable** > **Activate Configuration**.
6) After testing is complete, you can deactivate the alias setting override. To do so, select the activated alias setting override in the view and click **Enable/Disable** > **Deactivate Configuration**.

Reviewing the Feedback Provided by the User Community

A Feedback Bot capability is available in the slide-in toolbar that allows end users to provide feedback about any view, configured report, object cockpit, guide view, etc. implemented in their Alfabet solution. Two different concepts are available for the Feedback Bot:

- The Feedback Bot may implement a star-rating concept for feedback based on a 5-star rating system, whereby the user rates the current view and optionally provides feedback but does not expect a response to the feedback.
- The Feedback Bot may allow for a more elaborate contact form concept. This feedback concept functions similar to first-level support, whereby the user provides feedback or asks a question and expects a response from a user responsible for the feedback.

The **Feedback Review** functionality displays the user feedback information created in the Alfabet user interface by the user community. Each user feedback information consists of information about who created the feedback as well as a bookmark to the location in the Alfabet user interface where the bookmark was created. Please note that you must have the relevant access permissions to access the bookmark.

Depending on the configuration, screenshots may be made in association with a feedback entry. The screenshot will be made of the part of the screen where the user has the mouse pointer focused at the time that the user feedback information was created. No user information displayed in the masthead at the time that the screenshot was created will be included in the screenshot.

For information about the configuration required for the Feedback Bot capability, see the section *Configuring the Feedback Bot* in the reference manual *Configuring Alfabet with Alfabet Expand*.

Feedback that has been provided for a view or report via the Feedback Bot can be displayed in the Alfabet user interface in a secondary view for those users responsible for reviewing and responding to the feedback. This allows the responsible users to navigate the Alfabet user interface and see the feedback for the relevant view where they currently are. A secondary view with the caption **Feedback for Current View** will be displayed with a link if feedback has been provided for the view, configured report, object cockpit, guide view, etc. Clicking the link will open the **Feedback Review** functionality in a new browser tab which displays all feedback in detail for the view. The following is required to implement the **Feedback for Current View** capability:

- The **Enable Feedback for View** must be set to `True` for the user profiles responsible for reviewing and responding to feedback provided via the Feedback Bot.

- The **Enable Check Feedback for View** checkbox must be selected in the **User Settings** editor for the relevant users responsible for reviewing and responding to feedback. For more information, see the section Reviewing the Feedback Provided by the User Community in the reference manual *User and Solution Administration*.

A wide variety of filter settings enable you to focus on the feedback according to the type of view that feedback was created for, the feedback created by specific users or user profiles, the objects for which feedback was created, and the date when the feedback was created. Each user feedback information that matches the filters will be displayed in the table. You can select any feedback and navigate to its object profile where you can associate attachments, annotations, or evaluations with the feedback.

Define the following filters to view user feedback information and click **Update** to display the results in the table:

- **Feedback Type**: Select the feedback type that you want to view in the table.
- **User**: Select the user to display the feedback provided by the user.
- **User Profile**: Select a user profile to display the feedback provided by the users with the user profile.
- **Star Rating**: Select the number of stars to display the feedback based on a star-rating concept. Please note that the average star rating is displayed in the last row of the Star Ratings column.
- **View Type**: Select the technical type of view to display the feedback provided for this view type.
- **View Name**: Enter the technical name of the view to display the feedback provided for this view. Wildcards can be used if only a part of the view name is known.
- **Function**: Enter the technical name of the functionality to display the feedback provided for this functionality. Wildcards can be used if only a part of the functionality name is known.
- **Base Object Reference**: Enter the REFSTR to display the feedback for the associated object. Wildcards can be used if only a part of the REFSTR is known.
- **Base Object Class Name**: Select the object class to display the feedback provided for views for which objects in the selected object class are the base object.
- **Base Object**: Select the object to display the feedback provided for views for which the object is the base object. The **Base Object Class Name** field must be defined in order to specify a base object.
- **Occurrence After**: Enter a date to display feedback that was provided on or after the specified date.
- **Occurrence Before**: Enter a date to display feedback that was provided on or before the specified date.

The table may display all user feedback information found via the search criteria. The columns available will depend on the value selected in the **Feedback Type** filter:

- **ID**: Displays the unique identification number of the user feedback information.
- **Occurrence Time**: Displays the timestamp when the user feedback information was created.
- **User Email**: Displays the email address of the users creating the user feedback information. (Only displayed if the contract-form concept is selected in the **Feedback Type** field.)
- **User Phone**: Displays the phone number of the user creating the feedback based on the contact form concept. (Only displayed if the contract-form concept is selected in the **Feedback Type** field.)

- **Preferred Contact Type**: Displays the user creating the feedback based on the contact form concept prefers to be contacted by email or phone. (Only displayed if the contract-form concept is selected in the **Feedback Type** field.)
- **Comment Type**: Displays what kind of comment the user provided. The available comment types are configured via the protected enumeration `Feedback_ContactUs_Type`. (Only displayed if the contract-form concept is selected in the **Feedback Type** field.)
- **Comment**: Displays the comment provided with the feedback.
- **Star Ratings**: Displays the number of stars to view the feedback based on a star-rating concept. Please note that the average star rating is displayed in the last row of the Star Ratings column. (Only displayed if the star-rating concept is selected in the **Feedback Type** field.)
- **User**: Displays the user who created the user feedback information.
- **User Profile**: Displays the user profile of the user who created the user feedback information.
- **View Type**: Displays the type of view for which the user feedback information was created.
- **View Name**: Displays the name of the view for which the user feedback information was created.
- **Last Function Used**: Displays the name of the most recent functionality that the user accessed before creating the user feedback information. This information can help you understand the context in which the user feedback information was created.
- **Base Object**: Displays the object that the user was working with when the user feedback information was created.
- **Bookmark**: Displays a hyperlinked bookmark (express view) targeting the view for which the user feedback information was created.

Please note that when a user creates a user feedback information, a hyperlink is created to the relevant view where the feedback is made. The user profile used to access the view is by default the user profile that the creator of the user feedback information was logged in with when he/she created the feedback. For more information about the configuration of express views, see the section *Configuring the Express View (Email) Capability* in the reference manual *System Administration*.

You can open the screenshot associated with the user feedback information by selecting the feedback in the table and clicking the **Open Document Using Default Program**  button. A copy of the respective screenshot will open. You must have the corresponding program on your system in order for the selected document to open. You can also navigate to the object profile of a user feedback information to view the associated screenshot available in the *Attachments Page View*.

Feedback that has been responded to can be removed from the view via the **Delete** button. The deletion of a feedback instance will also remove the related feedback conversation.

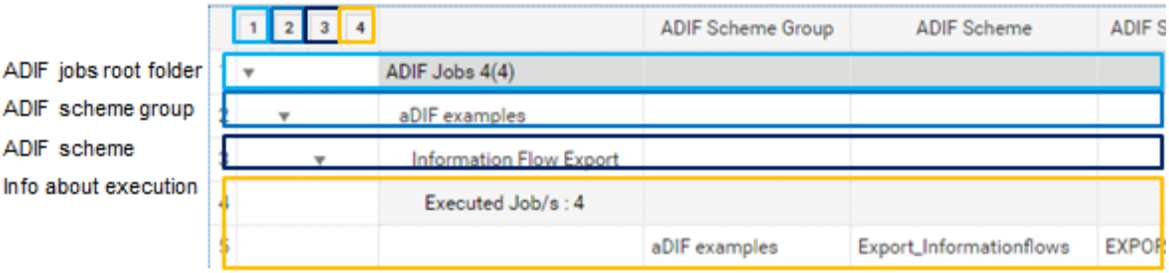
Chapter 19: Executing and Controlling ADIF Jobs

The **ADIF Jobs Administration** functionality allows you to execute or to test ADIF import and ADIF export and to check the success of the execution.

ADIF scheme configuration includes settings for the visibility and for the executability on the Alfabet user interface. Therefore, the view may only show a subset of ADIF schemes defined for your company and some of the ADIF schemes may be visible although you cannot execute ADIF import or ADIF export based on that ADIF scheme.

In the view, the ADIF schemes are listed in a view that shows all applicable ADIF schemes and the execution thereof in an expandable data set that shows information about ADIF job execution in the structure that is defined for the ADIF schemes in Alfabet Expand:

- The first level displays the root node and informs about the overall number of executed events matching the search conditions defined in the filter of the view.
- The second level displays ADIF scheme groups and ADIF schemes that are located directly under the root node.
- The third level displays the ADIF schemes in the ADIF group, or, for ADIF schemes located directly under the root node, the list of executed jobs.
- The fourth level displays the execution of ADIF jobs for ADIF schemes assigned to a group. The level has two sections **Executed Job/s** and **Started Job/s**. If an ADIF job is started asynchronously, the ADIF job is listed in the **Started Job/s** section during execution. After execution is finished, it is moved to the **Executed Job/s** section. Please note that asynchronous ADIF jobs are not listed in any of the sections as long as they are scheduled for execution at the server but execution has not been started yet. Synchronous execution of ADIF schemes inhibits reload of the view and therefore the jobs will never be visible in the **Started Job/s** section.



If ADIF jobs have been executed for an ADIF scheme one or multiple times, each execution is listed as a separate row under the heading **Executed Job/s** with the following information about the ADIF job execution:

- **ADIF Scheme Group**: The name of the ADIF scheme group the ADIF scheme is assigned to.
- **ADIF Scheme**: The caption of the ADIF scheme the ADIF job is executing.
- **ADIF Scheme Type**: Displays **EXPORT** for execution of an ADIF export scheme or **IMPORT** for execution of an ADIF import scheme.
- **Current Status**: Displays the overall status of the ADIF job execution:
  - **Started**: The ADIF job is currently executed. The ADIF job is listed in the section **Started Job/s**.
  - **Success**: The ADIF job has been successfully executed.
  - **Warning**: The ADIF job was executed, but a warning was written to the log file because a non-critical function failed.

    - **Failed**: The ADIF job has returned an error during execution.
- **Job Session**: The unique session ID of the ADIF job.
- **Start Time**: The time when the ADIF job execution started.
- **End Time**: The time when the ADIF job execution finished.
- **Executed By**: The way the ADIF job execution was started. The column returns one of the following:
    - `Expand ADIF Debugger`: The ADIF job was executed via the ADIF debugger of Alfabet Expand for testing purposes.
    - `User Interface`: The ADIF job execution was started from the Alfabet user interface via a button interaction. This is also displayed for test job execution on this view via the **Run Job** > **Non-Persistent Test Job** options.
    - `Rest API`: The ADIF job was triggered via a RESTful service call to the Alfabet RESTful API.
    - `Meta-Model Update`: The ADIF job was executed automatically during a meta-model update.
    - `Database Restore`: The ADIF job was executed automatically during a database restore from an Alfabet ADBZ file.
- **User Name**: The user name of the user that started the ADIF job. Please note the following about the user starting the ADIF job:
    - For ADIF jobs executed via a meta-model update or database restore that was started via the Alfabet Administrator the user information is empty because a user login is not required for the Alfabet Administrator.
    - For ADIF jobs executed via a RESTful service call to the endpoints `adifimport` or `adifexport` of the Alfabet RESTful API, the authentication user for the RESTful service call is identical with the user executing the ADIF job. This applies to RESTful service calls from an external RESTful client as well as for RESTful service calls triggered via events.

A filter is available to reduce the dataset to relevant content. Set the following filters fields on top of the table and click **Update** to view only data matching your filter settings:

- **ADIF Scheme Type**: Select **IMPORT** to view ADIF import schemes and ADIF jobs for ADIF export schemes only or **EXPORT** to view ADIF export schemes and ADIF jobs for ADIF export schemes only.
- **ADIF Schemes**: Select one or multiple ADIF schemes from the multi-select drop-down list to limit the display to the selected ADIF schemes and the ADIF jobs executed for the schemes.
- **ADIF Job State**: Select one of the following:
    - **Started** to view only currently running ADIF jobs.
    - **Success** to view only successfully executed ADIF jobs.
    - **Warning** to view only ADIF jobs executed with a warning message.
    - **Failed** to view only ADIF jobs terminated with an error.
    - **Execution Forcefully Terminated** to view only ADIF jobs that failed because the executing Alfabet Server was shut down either planned or forcefully during execution or the server thread for execution was forcefully terminated.

- **Start Date After**: Select a date from the calendar to view only ADIF jobs started at or after the selected date.
- **Start Date Before**: Select a date from the calendar to view only ADIF jobs started at or before the selected date.

You can do either of the following:

- Viewing Log Information for Executed ADIF Jobs
- Deleting Information About Executed ADIF Jobs
- Executing ADIF Jobs
- Testing ADIF Scheme Execution

Viewing Log Information for Executed ADIF Jobs

To view the log information about an executed ADIF job:

1) In the toolbar, click the **Show Log** button.
2) In the window that opens, click the **Download** button. The file is downloaded via the download mechanisms of your browser.

The button **Job Details** is only applicable for ServiceNow® integration. It provides information about the IDs of the log information returned by ServiceNow® that during execution of the ADIF job in verbose mode. The log information is stored in a table `ALFA_ADIF_SESSION_DETAIL` in the Alfabet database that is not visible on the Alfabet user interface or in Alfabet Expand. To see the log information, you can build a simple configured report based on the following query and export the information.

```
SELECT REFSTR,SESSION_ID,SCHEME_NAME,DETAIL_ID,IN_CONTENT,OUT_CONTENT
FROM ALFA_ADIF_SESSION_DETAIL
```

Deleting Information About Executed ADIF Jobs

You can delete information about ADIF job execution if it is not required any longer to keep the information stored in the underlying database table small and to enhance the clarity of information in the **ADIF Jobs Administration** and **My ADIF Jobs** functionalities.

To delete ADIF job session information, select one or multiple ADIF job information in the table and click the **Delete** button. The selected ADIF job session information is irrevocably deleted from the Alfabet database.

Executing ADIF Jobs

You can start ADIF jobs configured as executable via the user interface in the **ADIF Jobs Administration** functionality and not configured to require setting of mandatory parameters during execution.

ADIF jobs are executed asynchronously. They are queued for execution via an Alfabet Server. The execution might be delayed by other queued ADIF jobs. You can work on the Alfabet user interface in the current session while the ADIF job is queued and executed. The job will not be listed in the table of the **ADIF Jobs Administration** functionality while it is queued. It is listed in the section **Started Job/s** during execution and in the section **Executed Job/s** when finished. The user triggering the execution will also be informed about the success of the execution via an event feedback message.

To start execution of an ADIF job:

1) In the table, select the ADIF scheme that you would like to execute.
2) If the toolbar, click one of the following:
   - **Run Job Asynchronously via Server** > **Execute Import Job Asynchronously**: The ADIF import job will be executed asynchronously and only errors and warnings and the execution start and end time will be written to the log file.
   - **Run Job Asynchronously via Server** > **Execute Import Job Asynchronously with Verbose Logging**: The ADIF import job will be executed asynchronously, and detailed information will be written to the log file during execution.
   - **Run Job Asynchronously via Server** > **Execute Export Job Asynchronously**: The ADIF export job will be executed asynchronously and only errors and warnings and the execution start and end time will be written to the log file.
   - **Run Job Asynchronously via Server** > **Execute Export Job Asynchronously with Verbose Logging**: The ADIF export job will be executed asynchronously, and detailed information will be written to the log file during execution.

   If the button options are deactivated, the selected ADIF scheme is configured to be visible but not executable on the Alfabet user interface.
3) If you are executing an ADIF import scheme that imports data from file, a file selector window opens. Select the file that includes the data for upload and click **Upload**.
4) If you are executing an ADIF export scheme that exports data to a file, the export file is uploaded to the **Internal Document Selector** and will be available via the **Internal Documents** functionality. A dialog for selection of the allowed file locations in the Internal Document Selector opens. Select the checkbox of the target folder in the list of folders in the **IDOC Folder to Export** table and optionally specify a file name in the **Export File Name** field without file extension. If you do not specify a file name, the execution will generate a ZIP file with the name <NameOfADIFScheme>_<unformatted_timestamp>.zip.

Testing ADIF Scheme Execution

You can start all ADIF jobs configured as executable via the user interface in the **ADIF Jobs Administration** functionality in test mode. In test mode, information written to database tables is not persistent, but changes are reverted as last action of the ADIF job execution. The consequences of import can be tested via the log file without any risk to your data. For ADIF jobs exporting data to files, the test mode is not applicable. Using the test options will create the exported file as in normal execution mode.

1) In the table, select the ADIF scheme that you would like to execute.
2) If the toolbar, click any of the following:

- **Run Job Asynchronously via Server** > **Asynchronously Non-Persistent Import Test Job**: The ADIF import job will be executed asynchronously without persistently changing the data in the database and only errors and warnings and the execution start and end time will be written to the log file.
- **Run Job Asynchronously via Server** > **Asynchronously Non-Persistent Import Test Job with Verbose Logging**: The ADIF import job will be executed asynchronously without persistently changing the data in the database and detailed information will be written to the log file during execution.
- **Run Job Asynchronously via Server** > **Asynchronously Non-Persistent Export Test Job**: The ADIF import job will be executed asynchronously without persistently changing the data in the database and only errors and warnings and the execution start and end time will be written to the log file.
- **Run Job Asynchronously via Server** > **Asynchronously Non-Persistent Export Test Job with Verbose Logging**: The ADIF import job will be executed asynchronously without persistently changing the data in the database and detailed information will be written to the log file during execution.

3) If you are testing execution of an ADIF import scheme that imports data from file, a file selector window opens. Select the file that includes the data for upload and click **Upload**.
4) If you are executing an ADIF export scheme that export data to a file, the test mode will make no difference to the normal execution mode.

- The results are written into a persistent file. The export file is uploaded to the **Internal Document Selector** and will be available via the **Internal Documents** functionality. A dialog for selection of the allowed file locations in the **Internal Document Selector** opens. Select the checkbox of the target folder in the list of folders in the **IDOC Folder to Export** table and optionally specify a file name in the **Export File Name** field. The file name must have the extension .zip. If you do not specify a file name, the execution will generate a ZIP file with the name <NameOfADIFScheme>_<unformatted_timestamp>.zip.

Chapter 20: Managing Events

In the **Event Administration** functionality, you can check the success of events based on customer defined event templates. In addition, monitoring events can be triggered that check the availability of the Alfabet database.

The execution of some Alfabet functionalities such as the automatic translation capability or the job scheduler functionality depend on the scheduling of private events that are not visible to the user. Information about the execution of these events is not included in the **Event Administration** functionality.

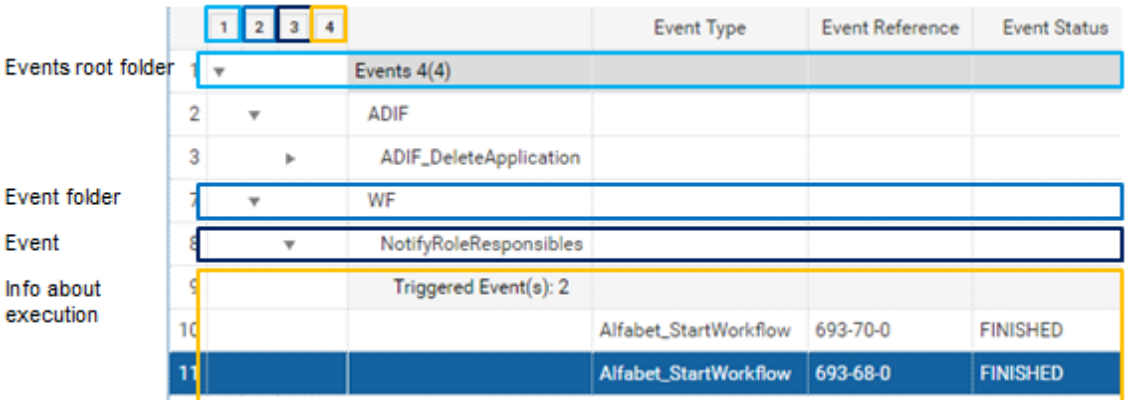
An event can trigger the start of a workflow, a REST API call, or the execution of an ADIF import or export. The execution takes place either in the same or in a different Alfabet database. An event will be triggered if a user performs tasks in the context of a workflow or wizard and a wizard step/workflow step is configured to trigger an event via an action to be executed such as when a wizard step is either entered, canceled, or exited, or a workflow step is entered, refused, expired or exited.

Events are based on configured event templates.

For information about configuring events, see *Configuring Events* in the reference manual *Configuring Alfabet with Alfabet Expand*.

The table in this view is an expandable dataset that shows information about event execution in the event template structure that is defined by a solution designer in the configuration tool Alfabet Expand:

- The first level displays the root node and provides information about the number of executed events.
- The second level displays event folders and events that are located directly under the root node.
- The third level displays the events in the event folder, or, for events located directly under the root node, a list of triggered events.
- The fourth level displays the triggering of events located in event folders.



If an event has been triggered one or multiple times, each triggering of the event will be listed as a separate row below the **Triggered Event(s)** row. The following information is displayed about the event execution:

- **Event Type**: Displays whether the event triggers the start of a workflow (`Alfabet_StartWorkflow`), a RESTful service call (`RestCall`), execution of an ADIF import (`Alfabet_StartAdifImport`), or execution of an ADIF export (`Alfabet_StartAdifExport`). Monitoring events are reported as the event type `Task`.
- **Event Reference**: The `REFSTR` of the triggered event. This is a unique identifier for the scheduled or executed event.

- **Event Status**: The status can be one of the following:
  - **Pending**: The event has been triggered and is queued for execution.
  - **Executing**: The event is currently executed.
  - **Finished**: The event has been successfully executed.
  - **Canceled**: The event has been canceled.
  - **Error**: The event has been executed and an error occurred during execution of the event.
  - **Execution Error**: The event has been executed and an error occurred during execution of the functionality triggered by the event.
  - **Execution Forcefully Terminated**: The execution of the event has been terminated in the context of a server shutdown.

    If monitoring events are scheduled for cyclic execution, regular executions will be managed via a single event. After the monitoring is triggered, the event will be immediately re-scheduled for execution ten minutes later. Therefore, the event status value of the event will remain **Pending** after monitoring has been triggered.

- **Event Callback Status**: This column is only filled for events triggering an ADIF import or export that are configured to wait for a callback from the triggered ADIF process. The event will have the status **Pending** as long as the ADIF process is executed. The status will then change to **Finished** or **Execution Error** depending on the success of the ADIF process execution.
- **Occurrence Time**: The date and time when the event was triggered.
- **Start Time**: This column is only filled for triggered cyclic monitoring events. Cyclic execution is based on a single event. After monitoring is executed, the event will not be terminated, but re-scheduled immediately for the next execution of monitoring. Therefore, the event status of the event will remain **Pending**. The next scheduled start time of monitoring will be displayed in the **Start Time** column.
- **Comment**: This field is currently not used.
- **Trigger Type**: Currently, the only trigger type is `ALFABET`. This means that the event is triggered from an Alfabet component.
- **Sender**: Currently, the only sender is `Alfabet`. This means that the event is triggered from an Alfabet component.
- **User**: The user triggering the event. This is the user who has executed the wizard or workflow that is configured to trigger the event.

A filter is available to limit the data in the view to a subset of triggered events:

- **Event Type**: Select a type of event in the drop-down list to limit the information to events triggering a defined action.
- **Event Names**: Select one or more event template names in the drop-down list to limit the information to events based on the selected event template.

  The drop-down list in this field is only filled if the **Event Type** field has been set to at least one event type and event templates of the selected event type are listed in the table displayed without filter settings.

- **Event Status**: Select one or more event statuses in the drop-down list to limit information to event executions that are currently in the selected status. For example, you can check for failed events by selecting the event status **Error** or check whether events are still queued or currently executed by selecting the event status **Pending** and **Executing**.
- **Event Message**: You can search for a specific event message or an event message containing a defined string by typing the search string in the field. Please note that wild cards are required at both ends of a string that is part of an error message to show results containing the string. Event messages are available for failed event executions and include error messages that were thrown during execution of the RESTful service call for the event. The event message is not displayed directly in the table. If you would like to view an event message, select the event in the table and click the **Show Event Message** button in the toolbar. The event message is written to a file that is then provided for download via the download mechanism of your browser.
- **Occurrence After**: Select a date in the calendar or enter a date in the correct format for the language you are using on the user interface. The data in the table will be limited to event execution at or after the selected date.
- **Occurrence Before**: Select a date in the calendar or enter a date in the correct format for the language you are using on the user interface. The data in the table will be limited to event execution at or before the selected date.
- **Connection**: Select a connection type in the drop-down list to limit display to events triggered for event templates that are self-reflective and exclusively target the same Alfabet database (`Self`) or that are using an Alfabet database connection to execute the event on the defined target database (`AlfaDBConnection`).
- **User**: Select a user from the selector or start typing a user name to see a drop-down list showing results that match the text you have entered. The data in the table will be limited to events triggered by the selected user.

Once the relevant filters have been defined, click **Update**. Event templates that do not have triggered events matching the defined search criteria as well as any triggered events within a **Triggered Event(s)** section that do not match the search criteria will be excluded from display.

If the execution of an event failed and the **Event Status** or **Event Callback Status** values are set to **Error** or **Execution Error**, a detailed error message including any error messages that were generated during the execution of the RESTful service call for the event can be downloaded for the failed event. Select the event in the table and click the **Show Event Message** button in the toolbar. The event message is written to a file that is then provided for download via the download mechanism of your browser.

After reviewing and correcting the cause for an error that occurred during event execution, the correction can be tested directly in the **Event Administration** functionality. Select the failed event in the table and click **Retry** in the toolbar to re-schedule the event for execution.

Chapter 21: Managing Automated Translation Strings

An automated translation capability is available that supports interoperability with a translation service such as Google Translate® and AWS Translate®. This capability allows the enterprise to provide a translated user interface for the supported secondary languages quickly, with a minimal amount of effort and cost. The **Automated Data Translations Functionality** allows you to review the translations that have been fetched for string for all classes that support automated translation. The translated strings will be available in each language associated with the cultures that support data translation. You can modify the translation, accept the translation and remove it from the view, or remove the automated translation.

- The following prerequisites must be fulfilled for strings to be successfully fetched for automated translations:
  - The enterprise must have a valid license to one of the supported translation services.
  - The connection to the translation service must be active in the XML object **AlfaTranslationServicesConfig**. For more information about configuring the XML object **AlfaTranslationServicesConfig**, see the chapter *Configuring Interoperability with a Translation Service* in the reference manual *API Integration with Third-Party Components*.
  - The Alfabet Server must be running and able to connect to the Internet.
  - The pre-conditions for activating the Rest API must be fulfilled. For more information, see the reference manual *Alfabet RESTful API*.
- A user with an administrative profile may review whether the event has been successfully triggered in the Events Administration Functionality. If the event is not displayed or has an `Error` status, then the prerequisites should be reviewed. For more information about events administration, see the chapter Managing Events in the reference manual *User and Solution Administration*.
- A user with an administrative profile may view the status of the ADIF import scheme (`Get Automated Instance Translations from Service Provider` or `Get Automated Translations for an Instance from Service Provider`) that are executed for the automated translations in the ADIF Jobs Administration Functionality. For more information about events administration, see the chapter Executing and Controlling ADIF Jobs in the reference manual *User and Solution Administration*.

  If automated translations have not been fetched due to Internet connection outages, a solution designer can trigger the ADIF import scheme `Get_Instance_Automated_Translations_For_Empty_Texts` or similar ADIF jobs to retrieve the automated data translations that have been missed as a result of the Internet connection outages. For more information about the predefined ADIF schemes for the automated translation capability, see the section *Predefined ADIF Schemes* in the reference manual *Alfabet Data Integration Framework*.

For more information about the configuration required to execute the automated translation capability for object classes, see the section *Configuring Automated Translation of Object Data* chapter *Localization and Multi-Language Support for the Alfabet Interface* in the reference manual *Configuring Alfabet with Alfabet Expand*.

One or more of the filters must be defined in order to view translation strings in the table and click **Update**:

- **Class**: Select an object class to display the automated translations for the objects in the specified class.
- **Property**: Select an object class property to display the automated translations for the objects in the object class selected in the **Class** field.

- **Secondary Language**: Select the language to display the automated translations for the specified language.
- **Original String**: Enter a term in the language of the user interface to find all original strings with the specified term. If the user interface is rendered in secondary language, the search results will include the strings translated to the primary language and secondary languages.
- **Translated String**: Enter a term in the language specified in the **Secondary Language** field to find all original strings with the specified term.
- **Object Reference/ID**: Enter either an object's ID or the REFSTR to find the object.
- **Translated Object**: In the selector, select the object whose translation you want to review and click **OK**.

The objects/strings matching the search criteria will be displayed. The table displays the following:

- The top level displays the number of automated translations that have been found.
- The second level displays the object class.
- The third level displays the objects in the object class having automated translations.
- Click the arrow on the third level to expand the table and view the properties that have been translated. A row will be displayed for each translation string. The following columns are displayed:
  - **Property**: Displays the property that has been translated.
  - **Culture**: Displays the culture code associated with the language of the translation.
  - **Automated Translation**: Displays the text of the automated translation.

The following information is available:

- Modifying the Text of an Automated Translation String
- Accepting the Automated Translation
- Removing the Automated Translation String

Modifying the Text of an Automated Translation String

You can edit the string that has been fetched by the automated translation, thus changing the translated string as needed. If a string is used in multiple object data, the same translation string will be changed for all occurrences of the string.

To edit the translation, select the automated translation string in the **Automated Translation** column that you want to modify and click the **Edit Translation** button in the toolbar. The **Automated Data Translation Editor** opens. Define the following fields, as needed and click **OK** to update the translation:

- **Instance Reference**: Displays the REFSTR of the object.
- **Property**: Displays the property that has been translated.
- **Primary Culture ID**: Displays the locale ID of the primary culture.
- **Primary Culture Name**: Displays the locale ID of the primary culture.

- **Translation Culture ID**: Displays the locale ID of the culture of the automated translation.
- **Translation Culture Name**: Displays the locale ID of the culture of the automated translation.
- **Current Date**:
  - **Primary Language Value**: Displays the original string that was translated.
  - Automated Translation: Displays the automated translation that you are modifying.
- **New Data**:
  - **Manual Translation Overwrite**: Enter the new translation that shall overwrite the automated translation.

Accepting the Automated Translation

In the process of reviewing the automated translation, you can specify that a translation has been accepted. By accepting the translation, you are indicating that it does not need to be changed anymore. To do so, select the automated translation string in the **Automated Translation** column that you want to accept and click **Accept Translation** button. The automated translation will be removed from the table but will remain in the Alfabet database.

Removing the Automated Translation String

You can remove an automated translation string and delete it from the Alfabet database. If a string is used in multiple object data, the same translation string will be deleted for all occurrences of the string.

To remove a translation fetched via the automated translation capability, select the automated translation string in the **Automated Translation** column that you want to remove and click **Manage Automated Translation** and select one of the following:

- **Remove Auto-Translation without Disabling Automatic Translation in Editor** to delete the automated translation string. The automated translation button for the respective field in the editor will not be disabled. If the value in the primary language is changed, a new automated translation will be fetched when the editor is closed.
- **Remove Auto-Translation and Disable Automatic Translation in Editor** to delete the automated translation string. The automated translation button for the respective field in the editor will be disabled. If the value in the primary language is changed, a new automated translation will not be fetched when the editor is closed.

Chapter 22: Managing the Translations for the Enterprise's Statutory Language

A statutory language capability is available in order to address regulatory requirements that mandate that objects are captured in a specific language. For example, if an enterprise has English as its primary language, but an organization in the enterprise is located in Germany and is required to capture data in German, than users can capture the data in the statutory language of German.

When a user creates an object based on a relevant object class/object class stereotype, a statutory language selector will open in which the statutory language can be selected. All languages supported by the enterprise will be available for selection. In the object editor that opens, the user can capture all attributes that support data translation in the statutory language. The automated translation capability will then translate the data to the primary language as well as other relevant secondary languages.

For more information about capturing data in a statutory language, see the section *Capturing Data in a Statutory Language* in the reference manual *Getting Started with Alfabet*. For more information about the configuration requirements to capture data in a statutory language, see the section *Specifying a Statutory Language for the Enterprise* in the reference manual *Configuring Alfabet with Alfabet Expand*.

The **Statutory Language Review for Data Translations** functionality allows you to review the values defined for the **Name** attribute of the objects with a statutory language definition and change them as needed. Furthermore, you can change the statutory language or remove the statutory language definition if needed.

One or more of the filters must be defined in order to view the objects with a statutory language definition in the table and click **Update**:

- **Class**: Select an object class to display the objects in the specified class.
- **Statutory Language**: Select the language to display the objects having the specified statutory language.
- **Object Reference/ID**: Enter either an object's ID or the REFSTR to find the object.
- **Object**: In the selector, select the object that you want to review and click **OK**.

The objects matching the search criteria will be displayed. The table displays the following:

- **Name**: Displays the name of the object. The name will be displayed in the language used to render the user interface.
- **ID**: Displays the object's identification number.
- **Statutory Language**: Displays the language that is specified as the statutory language for the object.

The following information is available:

- Modifying the Object's Name for the Statutory Language
- Modifying the Statutory Language for an Object
- Clearing the Statutory Language for an Object

Modifying the Object's Name for the Statutory Language

You can review the value defined for the **Name** attribute of an object specified in the statutory language and change the language as needed. Please note that the existing automated translations will not be removed, and the automated translation capability will not be triggered if the statutory language is changed.

1) Ensure that the Alfabet user interface is rendered in the statutory language that you want to capture the original data for.
2) In the **Statutory Language Review for Data Translations** functionality, select the object in the dataset whose **Name** attribute you want to revise and click the **Navigate**  button to open its object profile.
3) In the object profile, click the **Edit**  button to open the object's editor.
4) In the editor/wizard, ensure that the statutory language is selected in the language selector at the bottom of the editor/wizard.
5) Enter a value in the **Name** field.
6) Click the **OK** button in the editor or the **Next** button in the wizard to save the data. Existing automated translations will not be removed and the automated translation capability will not be triggered if the statutory language is changed.

Modifying the Statutory Language for an Object

You can change the object's statutory language to another language and specify the **Name** attribute of the object in the new statutory language. Please note that the existing automated translations will not be removed, and the automated translation capability will not be triggered if the statutory language is changed.

1) In the **Statutory Language Review for Data Translations** functionality, select the object whose statutory language you want to change.
2) In the toolbar, click the **Change Statutory Language** button. In the **Statutory Language** editor, select the language that you want to change the statutory language to and click **OK**.
3) Confirm the warning message by clicking the **Yes** button. Confirm the next warning by clicking the **Yes** button again. The language code is updated in the **Statutory Language** column.
4) In the toolbar, click the **Navigate**  button to open in the object's object profile.
5) Ensure that the Alfabet user interface is rendered in the new statutory language that you want to capture the data for.
6) In the object profile, click the **Edit**  button to open the object's editor.
7) In the editor/wizard, ensure that the statutory language is selected in the language selector at the bottom of the editor/wizard.
8) Enter a value in the **Name** field.
9) Click the **OK** button in the editor or the **Next** button in the wizard to save the data. Existing automated translations will not be removed, and the automated translation capability will not be triggered.

Clearing the Statutory Language for an Object

You can clear the object's statutory language thus removing the statutory language definition. Please note that the existing automated translations will not be removed, and the automated translation capability will not be triggered if the statutory language is cleared.

1) In the **Statutory Language Review for Data Translations** functionality, select the object whose statutory language you want to change.
2) In the toolbar, click the **Clear Statutory Language** button.
3) Confirm the warning message by clicking the **Yes** button. Confirm the next warning by clicking the **Yes** button again. The language code is removed from the **Statutory Language** column.

Chapter 23: Re-Enabling an Automated Assistant for a User

The **Administer Automated Assistants Functionality** allows you to reset an automated assistant that has been disabled by a user.

The **Enable Assistant** checkbox in the **User Settings** editor must be selected to enable all automated assistants that have been configured for your user profile. All assistants can be disabled by removing the checkmark for the **Enable Assistant** option. For more information, see the section *Defining Your User Settings in Alfabet*. For more information about the configuration required to implement the automated assistant capability, see the chapter *Providing Custom Online Help to the User Community* in the reference manual *Configuring Alfabet with Alfabet Expand*.

The view displays all automated assistants that have been disabled by users. Click any column caption to reorder the automated assistants based on the data in the column. The following information is displayed in the view:

- **Object Type**: Displays the type of configuration object that the automated assistant is assigned to.
- **Object Name**: Displays the object class that the configuration object that the automated assistant is assigned to if relevant.
- **Object Class**: Displays name of the configuration object that the automated assistant is assigned to.
- **Automated Help Link**: Displays the URL defined to target the content of the automated assistant.
- **Disabled Date**: Displays the date that the automated assistant was disabled.
- **User**: Displays the name of the user who disabled the automated assistant.

To limit the results in the table, define the following filters and click the **Update** button:

- **Object Type/Content**: Select the types of configuration objects that the automated assistant is assigned to.
- **Class**: Select the object classes that the configuration objects that the automated assistant is assigned to.
- **User**: Enter the name of the user who has disabled the automated assistant.
- **Date After**: Enter a date to display all automated assistants that have been disabled on or after the specified date.
- **Date Before**: Enter a date to display all automated assistants that have been disabled on or before the specified date.

To reset a disabled automated assistant so that it is displayed again for users, select the relevant row in the view and click the **Delete** button. The disabled assistant will be removed from the **Administer Automated Assistants Functionality**.

Chapter 24: Configuring and Managing Questionnaires

The questionnaire capability supports the enterprise to run campaigns to collect information relevant to the enterprise. The capability allows ad-hoc data-driven surveys with a potentially high volume of questions to be created. In contrast to the existing survey and compliance management capabilities available in Alfabet, the questionnaire capability provides great flexibility in specifying the scope of the assessment, allowing for various alternative answer schemes, changing and updating the questionnaire on the fly, and tracking the changes that occur to the assessment. Questions are bundled in question sets that are combined to create questionnaires. Answer schemes can be mapped to questions as needed. The following configuration is required in order to execute a questionnaire survey:

- Configure question sets, answer sets, and questionnaire policies in the **Questionnaire Structures & Policies** functionality. The questionnaire policies allow the target objects and questions relevant for the assessment to be identified as well as the users responsible for the questionnaire to be found.

  Questionnaire policies require a configured report with queries to find the relevant objects that are targeted by the questionnaire and, if necessary, to find the users responsible for answering the questions about the objects. The configured reports must be defined by the solution designer in the configuration tool Alfabet Expand and the **Category** attribute of the configured report must be set to a category defined for the questionnaire policy use case. For more information about creating the configured report as well as defining the **Category** attribute of the configured report, see the section *Configuring Reports Finding Objects Or Users for the Questionnaire Policy* in the reference manual *Configuring Alfabet with Alfabet Expand*.

- Create questionnaires in the **Questionnaire Creation** functionality. Multiple question sets can be assigned to the questionnaire. The questionnaire policies that are required to identify the target objects, questions to be answered, and users to answer the questions must be assigned to the questionnaire. Once a questionnaire has been created, the question sets, excluded questions, and questionnaire policies cannot be changed. Questionnaires can be versioned, thus allowing the designer to reuse indicators, remove irrelevant indicators, and add new indicators as needed in the new version of the questionnaire. Furthermore, you can specify whether the answers defined for the predecessor questionnaire version shall be copied to the new questionnaire version. If the answers shall be copied to the successor questionnaire version, it is possible to specify that individual answers shall not be propagated to a new questionnaire version.

  The questionnaire indicators will be generated asynchronously via the event template `CreateQuestionnaire`. The Alfabet Server must be running for the indicators to be created. You can review whether the event template was successfully triggered in the Events Administration Functionality.

- Assign the *My Questionnaire Indicators Page View* (`USER_QuestionaryIndicators`) to the user profiles that shall have access to questionnaires. Alternatively, a configured report based on the `QuestionnaireEvaluation` report template may be configured and implemented for users to answer the questions for questionnaires. The report template provides greater flexibility regarding how users answer questions in the context of the questionnaire capability. Please consider the following that may be configured for the configured report based on the `QuestionnaireEvaluation` report template:

  - The report shows a dataset that displays the question indicators to be answered for the selected object. Target questions that are dependent on a source question will be highlighted yellow in the dataset.
  - Navigation can be configured to any of the following displayed in the configured report:

    - Questionnaires
    - Questionnaire indicators, the objects targeted by the questionnaire indicator, and the questions associated with the questionnaire indicator. Please note that a `ReferenceArray` property `ObjectIssues` has been added to the class `QuestionaryIndicator` so that issues can be associated with questionnaire indicators via the object that is the target of the questionnaire indicator. A questionnaire indicator may reference multiple issues and each issue will be linked to an artifact object.
    - Questionnaire evaluations. Please note that a new questionnaire evaluation object view (`QUEV_ObjectView`) is available, allowing the solution designer to link the questionnaire evaluation object in configured reports based on the `QuestionnaireEvaluation` report template to its object profile. The new **Questionnaire Evaluation** editor allows a review comment and an approval comment to be provided for the questionnaire evaluation.
- A **Question** editor (`QUEST_Editor_Var`) is available that provides an Edit Seach interface control for the **Answer Set** field that eases selection if a large number of answer sets have been configured. The **Question** editor (`QUEST_Editor_Var`) can be configured to replace the standard **Question** editor (`QUEST_Editor`) in a custom class setting of the class `Question`.
- If issues have been defined for an object targeted by a questionnaire indicator, the issue and questionnaire indicator will also be associated. An `ObjectIssues` property of type `ReferenceArray` has been added to the class `QuestionaryIndicator` so that one or more issues defined for the object that is the target of the questionnaire indicator can be associated with the questionnaire indicator.
- For more information about configuring a report of the type `Custom` based on the `QuestionnaireEvaluation` report template, see the chapter *Configuring Reports* in the reference manual *Configuring Alfabet with Alfabet Expand*.

The following information is available:

- Creating the Questions, Answers, and Questionnaire Policies for the Questionnaire
    - Creating Question Set Categories
    - Creating Question Sets
    - Creating Questions
    - Specifying a Dependency for a Selected Question
    - Creating Answer Set Categories
    - Creating Answer Sets
    - Creating Answers
    - Configuring Questionnaire Policies
- Creating the Questionnaire
- Managing Questionnaire and Tracking Their Progress

Creating the Questions, Answers, and Questionnaire Policies for the Questionnaire

The **Questionnaire Structures & Policies** functionality allows question sets, answer sets, and questionnaire policies to be created and mapped. An answer set with potential answers must be assigned to each question in a question set. In addition to mapping the valid answer set to each question, you may also specify which answer assigned to the answer set constitutes the default answer. The order that questions are sequenced for a questionnaire are specified by defining the sort order for the questions in the question sets they are assigned to, the order of the question sets in the question set categories they are assigned to, and the order of the question set categories.

All question sets that are valid for the survey will be assigned to a questionnaire. Individual questions assigned to the question set that are not relevant for the questionnaire can later be excluded from the questionnaire.

A questionnaire policy must also be assigned to each questionnaire. A questionnaire policy allows the target objects and questions relevant for the assessment to be identified as well as the users responsible for the questionnaire to be found. For each questionnaire policy, you can specify either the authorized user and/or users with a specified role to answer the questions. Or, if other users are required to answer the questions, you can specify one or more queries to find the relevant users. Questionnaire policies may be reused in multiple questionnaires.

The configured reports must first be defined with the queries to find the relevant objects that are targeted by the questionnaire and, if necessary, to find the users responsible for answering the questions about the objects. The configured reports must be defined by the solution designer in the configuration tool Alfabet Expand.

The following information is available:

- Creating Question Set Categories
- Creating Question Sets
- Creating Questions
- Specifying a Dependency for a Selected Question
- Creating Answer Set Categories
- Creating Answer Sets
- Creating Answers
- Configuring Questionnaire Policies

Creating Question Set Categories

The **Question Set Categories** page view allows you to create question set categories. Question set categories allow you to organize and structure question sets.

A score management concept is available that allows questions and question sets to be weighted in order to compute the scores for the question set categories configured for the questionnaire. You can specify whether weighting of the question sets assigned to the selected question set category shall be equally distributed or whether the importance of the individual question sets can be weighted differently. Likewise, the user defining the questions sets can also specify whether weighting of the questions assigned to a selected question set shall

be equally distributed or whether the importance of the individual questions can be weighted differently. The weighting will be applied when the questionnaire evaluation is computed.

The specification of equal distribution for the individual questions in a question set is specified in the Question Sets Page View.

Question set categories associated with a questionnaire that has been activated can no longer be moved or deleted. In this case, the **New** menu and **Delete** button will disabled and greyed out.

To create a question set category:

1) Click the root node of the **Questionnaire Structures & Policies** explorer and click **Question Set Categories** to open the view.
2) In the toolbar, click **New** > **Create New Question Set Category**. The **Question Set Category** editor opens.
3) Define the following as needed and click **OK**.
   - **Name**: Enter a name for the question set category.
   - **Description**: Enter a description for the question set category.
   - **Equal Distribution**. Set a checkmark if the weighting of the questions and question sets assigned to the question set category shall be equally distributed (as a percentage) across all question sets assigned to the question set category. If the checkbox is selected, the **Edit Weighting** functionality will not be displayed in the Question Sets Page View.
4) Define the question sets for the selected question set category in the Question Sets Page View.
5) To compute all weights for all question set categories, question sets and questions based on the equal distribution concept, click the **Update All Weightings to Equal Distribution** button. Please note that all existing weighting definitions on subordinate question sets and questions will be irrevocably lost if you update all the weightings to equal distribution. Confirm the information message by clicking **Yes** to update the weightings to equal distribution or click **No** to exit without updating the weightings.
6) The sequence of the question set categories is based on the number displayed in the **Sort Number** column. To change the order that question set categories are sequenced for a questionnaire, click the **Define Sort Order** button. In the editor, select a question set category and click the **Up** / **Down** buttons to specify the sequence of the question set categories.

Creating Question Sets

The **Question Sets** page view allows you to create topical question sets that bundle the questions that will be posed in a questionnaire. The question sets that are relevant for the survey will be assigned to the questionnaire when the questionnaire is created in the Questionnaire Creation Functionality. A question set may only be assigned to one question set category. A question set may be reused in multiple questionnaires.

If the **Edit Weighting** button is displayed in the toolbar, you can specify whether different question sets shall have more weight than others when computing the score for the question set category that the question sets are assigned to.

Question sets associated with a questionnaire that has been activated can no longer be moved or deleted. In this case, the **New** menu and **Delete** button will disabled and greyed out.

To create a question set:

1) Click the root node of the **Questionnaire Structures & Policies** explorer and click **Question Set Categories**.
2) In the **Question Set Categories** view, select the question set category that you want to create the question sets for and click the **Navigate** button.
3) Click **Question Sets** to open the view.
4) In the toolbar, click **New** > **Create New Question Set**.

Alternatively, you can click **Move Existing Question Set Here** to move an existing question set to the selected question set category.

5) The **Question Set** editor opens. Define the following as needed and click **OK**.
   - **Name**: Enter a name for the question set.
   - **Description**: Enter a description for the question set.
   - **Equal Distribution**. Set a checkmark if the weighting of the questions assigned to the question set shall be equally distributed (as a percentage) across all questions assigned to the question set. If the checkbox is selected, the **Edit Weighting** functionality will not be displayed in the Questions Page View.
6) If the **Equal Distribution** checkbox has not be selected for the question set that the questions are assigned to, you will see the **Edit Weighting** button in the toolbar. The **Equal Distribution** checkbox is defined in the Question Set Categories Page View. If the weighting of the questions shall vary for individual questions, click the **Edit Weighting** button in the toolbar. The editor that opens contains a slider to adjust the weight value. Once a value has been specified for a question, the value can be locked by clicking it. A locked value will not change when the weighting is being specified for another question. The weighting of the questions will be applied when the question set is computed. Click **OK** to save the weighting of the questions for the question set.
7) The sequence of the question sets is based on the number displayed in the **Sort Number** column. To change the order that question sets are sequenced for a question set category, click the **Define Sort Order** button. In the editor, select a question set and click the **Up** / **Down** buttons to specify the sequence of the question sets for the selected question set category.

Creating Questions

The **Questions** page view allows you to create the questions that will be posed in a questionnaire. For each question created, you must assign the answer set containing the possible answers for the question. Optionally, you may specify a default answer in case no answer is selected by the responsible user. The questions that are relevant for the survey will be assigned via questions sets to the questionnaire when the questionnaire is created in the Questionnaire Creation Functionality. A question may only be assigned to one question set.

If the **Edit Weighting** button is displayed in the toolbar, you can specify whether different questions shall have more weight than others when computing the score for the question set that the questions are assigned to.

Furthermore, you can specify a condition for the selected question based on answers to the question. You can then specify one or more answers for the selected question that would exclude a target question from the questionnaire. Conditions for a selected question are defined in the Question Dependency Page View.

Questions associated with a questionnaire that has been activated can no longer be moved or deleted. In this case, the **New** menu and **Delete** button will disabled and greyed out. Furthermore, the **Answer Set**, **Values To Ignore for Score Computation**, **Enable Free-Form Text Answers**, and

Enable Multi-Select Answers fields will no longer be editable in the **Question** editor of a question that is part of an active questionnaire.

The table displays all questions that have been defined. Each column in the table is defined below:

- **Question Set Name**: Displays the name of the question set that the question belongs to.
- **Question Set Description**: Displays the information describing the purpose of the question set.
- **Short Question**: Displays the question.
- **Detailed Question**: Displays additional explanation of the question.
- **Answer Set Name**: Displays the answer set that is mapped to the question.
- **Default Answer**: Specifies the default answer in case no answer is selected for the question.

To create a question:

1) Click the root node of the **Questionnaire Structures & Policies** explorer and click **Question Set Categories** to open the view.
2) In the **Question Set Categories** view, select the question set that you want to create the questions for and click the **Navigate** button.
3) Click **Questions** to open the view.
4) In the toolbar, click **New** > **Create New Question**.

Alternatively, you can click **Move Existing Question Here** to move an existing question to the selected question set.

5) The **Question** editor opens. Define the following, as needed:
   - **Short Question**: Enter a short version of the question.
   - **Answer Set**: Select the answer set that includes the potential answers to the questions.
   - **Default Answer**: Select an answer in the answer set that should be the default answer if no answer is explicitly defined by the user responsible for the question.
   - **Values To Ignore for Score Computation**: Select the answers that will not be used in the score computation if no answer has been provided or an answer such as "not applicable" has been provided.
   - **Detailed Question**: Enter additional information that explains the question.
   - **Allow Copying the Answer to New Questionnaire Versions**: Select the checkbox if the answer to this question may be propagated to the next version of the questionnaire. The question will be available if it assigned to a questionnaire that is versioned and the **Copy Answers to New Questionnaire Versions** checkbox is selected in the **Questionnaire** editor of the predecessor questionnaire.
   - **Enable Multi-Select Answers**: Select the checkbox if the field to capture the answer shall be a multi-select field in the **Questionnaire Indicator** editor as well as in the context of configured reports based on the `Questionnaire Evaluation` report template. The value for a question with multiple answers will be the sum of the values specified for the question.
   - **Enable Free-Form Text Answers:** Select the checkbox in order to allow ad-hoc answers written in text form to be captured for the question in the **Questionnaire Indicator** editor as well as in

the context of configured reports based on the `Questionnaire Evaluation` report template. If the **Enable Free-Form Text Answers** checkbox is selected, all other fields in the **Question** editor will be hidden except for the **Short Question**, **Detailed Question**, and **Copy Answer to Questionnaire Versions** fields. Free-form text answers will not be included in the scoring mechanism.

6) Click **OK** to close the editor or **Cancel** to exit without saving your changes.

7) Create all questions for the question set as described above.

8) If the **Equal Distribution** checkbox has not be selected for the question set that the questions are assigned to, you will see the **Edit Weighting** button in the toolbar. The **Equal Distribution** checkbox is defined in the Question Sets Page View. If the weighting of the questions shall vary for individual questions, click the **Edit Weighting** button in the toolbar. The editor that opens contains a slider to adjust the weight value. Once a value has been specified for a question, the value can be locked by clicking it. A locked value will not change when the weighting is being specified for another question. The weighting of the questions will be applied when the question set is computed. Click **OK** to save the weighting of the questions for the question set.

9) The sequence of the questions is based on the number displayed in the **Sort Number** column. To change the order that questions are sequenced for a question set, click the **Define Sort Order** button. In the editor, select a question and click the **Up** / **Down** buttons to specify the sequence of the question for the selected question set.

Specifying a Dependency for a Selected Question

The **Question Dependencies** page view allows conditions for a selected question to be defined. You can select a target question that is dependent on the answer defined for the source question. Then you must specify one or more answers to the source question that determine whether the target (dependent) question is or is not included in the questionnaire. To do so, you can specify one or more answers in the **Excluding Answers** field, which determines that the target question will not be included in the questionnaire in the **My Questionnaires** functionality if one of the excluded answers is selected for the source question. Or, you can specify one or more answers in the **Including Answers** field, which determines that the target question will be included in the questionnaire in the **My Questionnaires** functionality if one of the included answers is selected for the source question. You can only specify either the **Excluding Answers** field or the **Including Answers** field. An error message will be displayed if both fields are defined.

To create a dependency:

1) Click the root node of the **Questionnaire Structures & Policies** explorer and click **Question Set Categories** to open the view.

2) In the **Question Set Categories** view, select the question set that you want to create the questions for and click the **Navigate** button.

3) Click **Questions** to open the view and click the question that you want to define the dependency for.

4) In the toolbar, click **New** > **Create New Question Dependency**.

5) The **Question Dependency** editor opens. Define the following, as needed:

- **Source Question**: Displays the question that is currently selected.
- **Target Question**: Select the target question that is dependent on the source question.
- **Excluding Answers**: Select one or more answers that make the target question irrelevant for the questionnaire. The target question will not be included in the questionnaire if one of the answers in the **Excluding Answers** field is selected by a user to answer the source question. You cannot

specify the **Including Answers** field if the **Excluding Answers** field has been defined. You cannot specify the **Excluding Answers** field if the **Including Answers** field has been defined.

- **Including Answers**: Select one or more answers that make the target question relevant for the questionnaire. The target question will be included in the questionnaire if one of the answers in the **Including Answers** field is selected by a user to answer the source question. You cannot specify the **Including Answers** field if the **Excluding Answers** field has been defined.

6) Click **OK** to close the editor or **Cancel** to exit without saving your changes.

Creating Answer Set Categories

The **Answer Set Categories** page view allows you to create answer set categories. Answer set categories allow you to organize and structure answer sets.

Answer set categories associated with a questionnaire that has been activated can no longer be moved or deleted. In this case, the **New** menu and **Delete** button will disabled and greyed out.

To create an answer set category:

1) Click the root node of the **Questionnaire Structures & Policies** explorer and click **Answer Set Categories** to open the view.
2) In the toolbar, click **New** > **Create New Answer Set Category**. The **Answer Set Category** editor opens.
3) Define a name and provide a description of the answer set category in the editor and click **OK**.

Creating Answer Sets

The **Answer Sets** page view allows you to create topical groups to bundle the answers that will be available to answer specific questions. The answer sets that are relevant for the survey will be assigned to a question. An answer set may only be assigned to one answer set category.

Answer sets associated with a questionnaire that has been activated can no longer be moved or deleted. In this case, the **New** menu and **Delete** button will disabled and greyed out.

To create a question set:

1) Click the root node of the **Questionnaire Structures & Policies** explorer and click **Answer Set Categories**.
2) In the **Answer Set Categories** view, select the answer set category that you want to create the answer sets for and click the **Navigate** button.
3) Click **Answer Sets** to open the view.
4) In the toolbar, click **New** > **Create New Answer Set**.

Alternatively, you can click **Move Existing Answer Set Here** to move an existing answer set to the selected answer set category.

5) The **Answer Set** editor opens. Define a name and provide a description of the answer set in the editor and click **OK**.

Creating Answers

The **Answers** page view allows you to create the answers that will be available to answer specific questions. The answers will be assigned to a relevant question via an answer set in the Questions Page View. For each question defined, one answer in the answer set will be assigned as the default answer for the question. An answer may only be assigned to one answer set.

Answers associated with a questionnaire that has been activated can no longer be moved or deleted. In this case, the **New** menu and **Delete** button will disabled and greyed out.

To create an answer:

1) Click the root node of the **Questionnaire Structures & Policies** explorer and click **Answer Set Categories** to open the view.
2) In the **Answer Set Categories** view, select the answer set that you want to create the answers for and click the **Navigate** button.
3) Click **Answers** to open the view.
4) In the toolbar, click **New** > **Create New Answer**.

   Alternatively, you can click **Move Existing Answer Here** to move an existing answer to the selected answer set.

5) The **Answer** editor opens. Define the following, as needed:
   - **Short Answer**: Enter a short concise answer.
   - **Detailed Answer**: Enter additional information that explains the answer.
6) Click **OK** to close the editor of **Cancel** to exit without saving your changes.
7) The sequence of the answers is based on the number displayed in the **Sort Number** column. To change the order that the answers are sequenced for an answer set, click the **Define Sort Order** button. In the editor, select an answer and click the **Up** / **Down** buttons to specify the sequence of the answers for the selected answer set.

Configuring Questionnaire Policies

The **Questionnaire Policies** page view allows you to configure questionnaire policies that can be assigned to questionnaires. The questionnaire policy describes a rule to determine the scope and responsibility for the questions in a questionnaire.

Questionnaire policies specify the target objects and users to answer the questions. The definition of a questionnaire policy includes the object class that is targeted by the questionnaire and the queries that have been defined by the solution designer in the configuration tool Alfabet Expand. The questionnaire policies that are needed for a questionnaire must be assigned to the question in the Questionnaire Creation Functionality.

- For each questionnaire policy, you can specify one or more configured reports each containing an object query that finds the objects targeted by the questionnaire.
- For each questionnaire policy, you can specify either the authorized user and/or users with a specified role to answer the questions. Or, if other users are required to answer the questions, you can specify one or more queries to find the relevant users.

  Once a questionnaire has been created, the user defining the questionnaire can specify that the questions can be answered by more than one user. This is defined via the **Collaborative Questionnaire** checkbox in the **Questionnaire** editor. For more information, see the section Creating a Questionnaire.

- Multiple questionnaire policies may be assigned to a questionnaire. Only one questionnaire policy per base class may be assigned to the questionnaire. When the questionnaire is activated, questionnaire indicators will be created for each questionnaire/object class/user combination.

Configured reports must first be defined with the queries to find the relevant objects that are targeted by the questionnaire and, if necessary, to find the users responsible for answering the questions about the objects. The configured reports must be defined by the solution designer in the configuration tool Alfabet Expand and the **Category** attribute must be set to a category defined for the questionnaire policy use case. For more information about creating the configured report as well as defining the **Category** attribute of the configured report, see the section *Configuring Reports Finding Objects Or Users for the Questionnaire Policy* in the reference manual *Configuring Alfabet with Alfabet Expand*.

The table displays all questionnaire policies that have been defined. Each column in the table is defined below:

- **Name**: Displays the name of the questionnaire policy.
- **Class**: Displays the object class that the questionnaire policy addresses.
- **Description**: Displays the description of the questionnaire policy.
- **Authorized User Is Responsible**: A checkmark is displayed if the authorized user of the object is specified to answer questions for the object.

To create a new questionnaire policy:

1) Click the root node of the **Questionnaire Structures & Policies** explorer and click **Question Policies** to open the view.
2) In the toolbar, click **New** > **Create New Question Policy**. The **Policy** editor opens.
3) Enter the information in the fields, as needed.

   Basic Data tab:

   - **Name**: Enter a unique name for the questionnaire policy.
   - **Class**: Select the object class that the questionnaire policy should address. The class you select here determines which queries can be selected in the **Object Queries** tab as well as the role types available in the **Simple Permissions** tab.
   - **Description**: Enter a meaningful description for the questionnaire policy.

   Object Queries tab: This tab allows you to specify one or more object queries to find the objects targeted by the questionnaire.

- **Object Queries**: Select one or more configured reports with the queries to use to search for the objects that should be the target of a questionnaire. The available queries are determined by the object class selected in the **Class** field in the **Attributes** tab.

  Once a questionnaire has been created, the user defining the questionnaire can specify that the questions can be answered by more than one user. This is defined via the **Collaborative Questionnaire** checkbox in the **Questionnaire** editor. For more information, see the section Creating a Questionnaire.

- **Authorized User Is Responsible**: Select this checkbox if the authorized user for an object is responsible to answer questions about the object he/she is responsible for.

- **Role Types**: Select the role types that are relevant to answer the questions. Users defined to have the selected role for a target object will be responsible for assessing the target object. The role types are determined by the object class selected in the **Class** field in the **Basic Data** tab.

  The role types you see here are configured for the object class you have defined in the **Class** field in the **Basic Data** tab. For more information about configuring role types in the **Configuration** module, see the chapter *Configuring Role Types to Define Roles in the Responsibilities Page View* in *Configuring Evaluation and Reference Data in Alfabet*.

  Once a questionnaire has been created, the user defining the questionnaire can specify that the questions can be answered by more than one user. This is defined via the **Collaborative Questionnaire** checkbox in the **Questionnaire** editor. For more information, see the section Creating a Questionnaire.

- **Permission Rules**: Select one or more queries to determine the persons that are responsible for answering questions about the target objects found by the queries in the **Object Queries** tab.

4) Click **OK** to save the questionnaire policy or **Cancel** to exit without saving.

Creating the Questionnaire

The **Questionnaire Creation** functionality allows you to create questionnaires to send out to your user community. Each questionnaire requires the specification of one or more question sets containing the questions and answers mapped to the questions as well as a questionnaire policy configured to find the objects targeted by the questionnaire as well as the users responsible for answering the questions.

Questionnaires can be versioned, thus allowing the designer to reuse questions and their answers, remove irrelevant questions, and add new questions as needed in the new version of the questionnaire. You can specify whether the answers defined for the predecessor questionnaire version shall be copied to the new questionnaire version. If the answers shall be copied to the successor questionnaire version, it is possible to specify that individual answers shall not be propagated to a new questionnaire version. The content of the **Configuration** tab in the editor will be copied from the predecessor questionnaire and can be modified for the new questionnaire version as needed.

An answer defined for a predecessor questionnaire will be copied to the new questionnaire version if the following is true:

- The **Allow Copying the Answer to New Questionnaire Versions** checkbox in the **Question** editor has been selected. If the checkbox is not selected, the answer to the question in the predecessor questionnaire will not be copied to the questionnaire version.

- The **Copy Answers to New Questionnaire Versions** checkbox in the **Questionnaire** editor has been selected. If the **Copy Answers to New Questionnaire Versions** checkbox is selected, the **Do Not Allow Copying to the New Questionnaire Version** checkbox will be available in the **Questionnaire Answer** editor and individual answers can be specified to not be copied to a new questionnaire version. Otherwise, the **Do Not Allow Copying to the New Questionnaire Version** checkbox will be disabled.
- Individual answers can be explicitly specified to not be copied to a new questionnaire version via the **Do Not Allow Copying to New Questionnaire Version** checkbox in the **Questionnaire Answer** editor available in the *My Questionnaires Page View* and Questionnaire Management Functionality.

The questionnaire indicators will be generated asynchronously via the event template `CreateQuestionnaire`. The Alfabet Server must be running for the indicators to be created. You can review whether the event template was successfully triggered in the Events Administration Functionality.

Once the questionnaire has been created and the **Questionnaire** editor is closed via the **OK** button, the event template `CreateQuestionnaire` will be triggered. The questionnaire is thus activated, and the questions will be made available to the users that are responsible for the questions. Once a questionnaire is activated, any questions, question sets, question set categories as well answers, answer sets, and answer set categories can no longer be moved or deleted and both the **New** menu as well as **Delete** button will be disabled in the relevant views. Users can answer the questions they are responsible for in the *My Questionnaires Page View*. You can track the completeness of the questionnaire in the Questionnaire Management Functionality.

A report can be configured based on the `QuestionnaireEvaluation` report template in order to provide further flexibility regarding how users answer questions in the context of the questionnaire capability. The report shows a dataset that displays the question indicators to be answered for the selected object. Target questions that are dependent on a source question will be highlighted yellow in the dataset. For more information about configuring a report of the type `Custom` based on the `QuestionnaireEvaluation` report template, see the chapter *Configuring Reports* in the reference manual *Configuring Alfabet with Alfabet Expand*.

The following information is available in the view.

- **Questionnaire Policy Class**: Displays the object class targeted by the questionnaire policy.
- **Questionnaire Policy Name**: Displays the name of the questionnaire policy.
- **Questionnaire Policy Description**: Displays the description of the questionnaire policy.
- **Questionnaire Name**: Displays the name of the questionnaire.
- **Questionnaire Description**: Displays the description of the questionnaire.
- **Questionnaire Target Date**: Displays the target date when the questionnaire should be completed.
- **Questionnaire Status**: Displays the release status of the questionnaire.
- **Previous Questionnaire Version**: Displays the preceding questionnaire version if the questionnaire is based on a previous version.
- **Next Questionnaire Version** Displays the subsequent questionnaire version if the questionnaire has been versioned.

To create the questionnaire:

1) In the toolbar, click one of the following: **New** > **Create New Questionnaire**.

- **New** > **Create New Questionnaire**, or
- **New** > **Create New Questionnaire Version**. Questionnaires can be versioned, thus allowing the designer to reuse indicators, remove irrelevant indicators, and add new indicators as needed in the new version of the questionnaire. The content of the **Configuration** tab in the editor will be copied from the predecessor questionnaire and can be modified for the new questionnaire version as needed.

2) The **Questionnaire** editor opens. Enter the information in the fields, as needed.

Basic Data tab:

- **Name**: Enter a unique name for the questionnaire.
- **Version**: A default version number following the version number of the predecessor questionnaire will be automatically displayed. This can be edited as needed.
- **Target Date**: Select the target date when the questionnaire should be completed by the responsible users.
- **Release Status**: Select the release status of the questionnaire.

The set of release statuses available for an object class are configured by your solution designer in the configuration tool Alfabet Expand. For more information, see the section *Configuring Release Status Definitions for Object Classes* in the reference manual *Configuring Alfabet with Alfabet Expand*. For general information about release statuses, see the section *Understanding Release Statuses* in the reference manual *Getting Started with Alfabet*.

- **Description**: Enter a meaningful description for the questionnaire policy.
- **Collaborative Question**: Select the checkbox if the questionnaire can be answered by multiple users who are part of the questionnaire policy. If the questionnaire is not collaborative, then only the user that has been specified in the indicator for the object will be able to answer the question.
- **Copy Answers to New Questionnaire Versions**: Select the checkbox if the questionnaires can be versioned, thus allowing the current questionnaire's questions and answers to be reused. Irrelevant questions can be removed, and new questions added as needed in the new version of the questionnaire. The content of the **Configuration** tab in the editor will be copied from the predecessor questionnaire and can be modified for the new questionnaire version as needed.

Configuration tab:

- **Questionnaire Policy**: Select one or more questionnaire policies to use to find the objects targeted by the questionnaire and the users responsible for answering the questions. Only one questionnaire policy per base class may be assigned to the questionnaire. When the questionnaire is activated, questionnaire indicators will be created for each questionnaire/object class/user combination.
- **Question Sets**: Select one or more question sets with the questions that shall be posed in the questionnaire.
- **Excluded Questions**: Select the questions that shall be excluded from the questionnaire. All questions assigned to the selected question sets are displayed in the drop-down list.
- **Selected Questions**. Displays all questions assigned to the questionnaire. Questions defined in the **Excluded Questions** field will be removed from the **Selected Questions** field.

3) Click **OK** to save the questionnaire. The event template `CreateQuestionnaire` will be triggered, and the questionnaire will be generated asynchronously. The questionnaire indicators will be made available to the users that are responsible for the questions. The Alfabet Server must be running for the questionnaire indicators to be created.

Managing Questionnaire and Tracking Their Progress

The **Questionnaires Management** functionality allows you to track the completeness of a questionnaire.

Users can answer the questions they are responsible for in the *My Questionnaires Page View* or in a configured report defined by a solution designer.

All questionnaires with their questions will be displayed in the view. To limit the data displayed in the table, define one or more of the following filters and click the **Update** button:

- **Questionnaire**: Select a questionnaire to display. The definition of either the **Questionnaire** filter or the **Target Class** filter is mandatory to display results in the dataset.
- **Question Set**: Select a question set to display. This field can only be defined if a questionnaire has been selected in the **Questionnaire** field.
- **Questions**: Select a question assigned to the selected question set to display all instances of the question. This field can only be defined if a question set has been selected in the **Question Set** field.
- **Question Completed**: Select **Show All Answered Questions** to display all questions that have been completed, select **Show All Unanswered Questions** to display all questions that have not been completed, or **Show All Questions** to display all questions regardless of their completion.
- **Target Date After**: Enter a target date to display all questionnaires with a target date on or after the specified date.
- **Target Date Before**: Enter a target date to display all questionnaires with a target date on or before the specified date.
- **Responsible User**: Select a user to display all questions assigned to the selected user.
- **Target Class**: Select an object class to view questions in questionnaires targeting the object class. The definition of either the **Questionnaire** filter or the **Target Class** filter is mandatory to display results in the dataset.
- **Target Object**. Enter the name of an object targeted by the questionnaire.

The following information is displayed in the view.

- **Questionnaire**: Displays the name of the questionnaire.
- **Questionnaire Target Date**: Displays the target date when the questionnaire should be completed.
- **Short Question**: Displays the question.
- **Short Answer**: Displays the answer for the question.
- **Question Completed**: Displays a checkmark if the question has been marked as complete.
- **Target Object**: Displays the name of the object that the question targets.

- **Person User Name**: Displays the name of the user responsible for answering the question.

To specify an answer for a question, select the question in the view and click **Edit**  button. In the **Questionnaire Answer** editor, define the following fields as needed and click the **OK** button:

- **Answer:** Select an answer in the **Answer** field. The **Answer** field will display a simple combo box if only one answer can be selected for the question and a multi-select combo box if multiple answers are possible.
- **Comment**: Provide additional information about the answer in the **Comment** field
- **Do Not Allow Copying to the New Questionnaire Version**: Select the checkbox if the answer shall not be propagated to the next version of the questionnaire. The checkbox can only be selected if questionnaire versioning has been configured for the questionnaire.
- **Completed**: The checkbox will be selected if the **Questionnaire Answer** editor has been closed and the answer to the question has been saved to the Alfabet database. A checkmark will also be set for the question in the **Question Completed** column.

Chapter 25: Configuring the AlfaBot Capability

The AlfaBot capability provides assistance via a textual chatbot to help Alfabet users in general tasks such as creating, editing, and finding objects as well as navigating to configured reports. The Alfabet capability is based on natural language processing technology provided by the third-party component Dialogflow™. Users can open the AlfaBot window, ask for help by typing in texts as "Create an application" or "Edit TradeNet" or "Open report". The AlfaBot will respond by either asking for more information to refine the search or by providing a list of objects matching the input.

For information about the prerequisites to set up the AlfaBot capability, see the section *Implementing the AlfaBot for Navigation via a Full-Text Command* in the reference manual *Configuring Alfabet with Alfabet Expand*. For end user information about accessing the AlfaBot capability in the user interface, see the section *Using the AlfaBot User Assistance* in the reference manual *Getting Started with Alfabet*.

A security warning may be displayed in the AlfaBot warning users that data input may potentially be used by a third-party service. A configuration mechanism is available that allows a static message in the AlfaBot to be displayed with information about the security issue and a link to additional relevant information. The warning box will be embedded in the AlfaBot below the header and will remain visible throughout the time that the AlfaBot is open. For more information about configuring the third-party data securing warning, see the section *Configuring Third-Party Data Security Warnings* in the reference manual *Configuring Evaluation and Reference Data in Alfabet*.

The following information is available:

Setting Up the AlfaBot in the AlfaBot Configuration Functionality

The **AlfaBot Configuration** functionality is required to activate the AlfaBot after the initial setup has been performed by a solution designer in Alfabet Expand. It also allows training phrases to be defined in order to enhance the response rate for user requests.

The AlfaBot capability is configured to understand a defined set of intents. An intent is a functionality request such as help to a defined page view or report. Training phrases must be initially defined for an intent. The Dialogflow™ component analyzes the user input and checks whether the phrase entered by the user matches the structure of a training phrase in order to identify the intent. The analysis result is provided back to Alfabet and the action specified in the analysis result is triggered in the Alfabet user interface.

You can do any of the following to set up and configure the AlfaBot functionality:

Initially Setting Up the AlfaBot

When the functionality opens without the AlfaBot being initialized, the explorer will be empty and a **Setup AlfaBot** button will be displayed. Click the button to initialize the AlfaBot. The explorer will be filled with the available intents and the button will no longer be displayed.

Creating Training Phrases for Intents

The functionality of the AlfaBot is based on intents. An intent is a defined functionality (such as the creation of a new object or the navigation to a configured report) that triggers a predefined response of the AlfaBot. Training phrases are defined for each intent. A training phrase is a phrase that might be entered by a user to trigger the intent. It can contain placeholders for all variables that may occur in the phrase. For example, if a user wants to create an object for an object class or object class stereotype, the placeholder can be entered in the training phrase at the position where the name of the object class or object class stereotype is expected. Placeholders have a predefined name starting with @. For example, a training phrase for the creation of an object could be:

```
I want to create a @classOrStereotype
```

Please note that the AlfaBot only understands English phrases.

If a user enters a phrase in the AlfaBot, the phrase will be sent to Dialogflow. Dialogflow analyzes the result and sends information about the identified intent and the placeholder values back to the AlfaBot. The AlfaBot will execute the intent if both the intent and the placeholders are meaningfully identified in a phrase. Otherwise, it will try to execute one of the intents that request additional information from the user.

The sentences the users type in the AlfaBot window as well as the interpretations returned by Dialogflow are stored in the Alfabet database and Alfabet will try to interpret user sentences based on the already stored and analyzed user sentences if a user uses the same wording for a similar action. If the meaning of the request can be evaluated based on stored prior requests, the sentence will not be sent to Dialogflow again. The costs for language analysis by Dialogflow can be reduced over time via this mechanism.

Internal processing of user input can be deactivated via the **Turn Config Mode On** button in the workspace of the root node of the explorer of the **AlfaBot Configuration** functionality.

For both internal processing and processing via Dialogflow, user input will be processed in different consecutive steps which are all executed with a fuzzy search mechanism to cope with misspellings:

- The user input is first compared to all available training phrases to identify the intent. If the training phrase matches an intent and placeholders are included in the training phrase, the words entered in the location of the placeholders will be identified.
- The captions of relevant configuration objects in the current configuration of the meta-model are provided to Dialogflow as a list of allowed values for the respective placeholders. For example, a value for the placeholder @classOrStereotype will only be resolved successfully if it is identical to the caption of an existing object class or object class stereotype in the current configuration. If user input cannot be

mapped to one of the intents via the training phrases, it will be compared with the lists of allowed values for the entity definitions. If it matches an entity definition, the AlfaBot will process it as information about the content provided by the entity. For example, if the AlfaBot asks the user for which object class he/she would like to create an object, the user can enter the object class caption without any additional text. The entry will correspond to an allowed value for object class caption and will be processed as such. Fuzzy search within entity definitions is used to cope with typos in the requests.

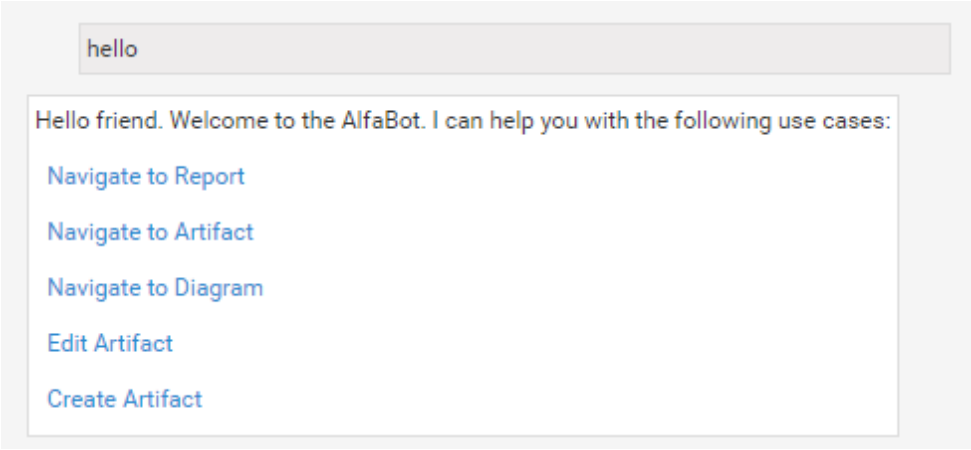
The entities must be updated if the meta-model configuration changes. Details are provided below.

- If no matching results are found and the user input consists of more than one word, the user input will be split into separate words and a keyword search will be performed. For configured reports, the search mechanisms performs a synonym search on the text provided in the **Business Problem Statement** attribute in addition to a keyword search on caption and description of the configured report.

The font color and font style of the link lists in the AlfaBot are configurable. The styling of intents and options is carried out via the **Alfabot Options Styles** section of the GUI scheme. For more information, see the section *Configuring GUI Scheme Definitions for the Alfabet Interface* in the chapter *Configuring the Visualization of the Alfabet Interface* as well as a detailed documentation of all GUI scheme attributes in the chapter *Overview of GUI Scheme Attributes* in reference manual *Configuring Alfabet with Alfabet Expand - Appendix*.

All intents that can be processed by the AlfaBot are listed in the explorer of the **AlfaBot Configuration** functionality. The following intents are available:

- **Welcome**: Phrases that the user might enter as a welcome text with no additional intention, like "Hello" or "Good Morning". The AlfaBot will answer with any of the configured responses followed by a list of the activities that are supported by the AlfaBot. The user can click any of the activities and will then be asked immediately to provide the required information about the object, class or view names relevant for the intent of the selected use case without needing to type a phrase that identifies the intent.



The font color and font style of the link list can be changed via the GUI scheme configuration by means of the attributes in the attribute section **Application** > **AlfaBot Option Styles** > **Intent Style**. For more information about the configuration of GUI schemes, see *Configuring GUI Scheme Definitions for the Alfabet Interface*.

- **Create Artifact**: If the user wants to create a new object, he/she can use any of the configured training phrases. Object creation can be supported either context-free or context-dependent: The

AlfaBot needs the caption of the object class or object class stereotype to process the intent. If the object class or object class stereotype is configured to allow creation via the AlfaBot, the AlfaBot will open the editor or wizard for creation of a new object. After the user has created the object, the object profile or object cockpit of the new object will open in the Alfabet user interface. If the object class or object class stereotype is configured to deny creation via the AlfaBot, the AlfaBot will inform the user.

- Context-free object creation: Any object that can be created without knowing the defined context (such as an application) can be created context-free. The AlfaBot requires the caption of the object class or object class stereotype to process the intent. If the object class or object class stereotype is configured to allow creation via the AlfaBot, the AlfaBot will open the editor or wizard to create the new object. After the user has created the object, the object profile or object cockpit of the new object will open in the user interface.
- Context-dependent object creation: Any object that requires a specific context (such as a business service which can only be created in the context of a providing application) is created context-dependent. The AlfaBot needs the caption of the object class or object class stereotype to process the intent. If the object class or object class stereotype is configured to allow creation via the AlfaBot, the AlfaBot will scan the configuration of the current user profile for the ability to create the object via the buttons in page views. If the object can be created in multiple views, the user will be asked for more details. The AlfaBot will also ask the user for the name of the object he/she wants to create a dependent object for. The AlfaBot will open the relevant view for the object that the user specifies as the context object.

If the object class or object class stereotype is configured to deny creation via the AlfaBot, the AlfaBot will inform the user.

- **Navigate to Artifact**: If the user wants to open a standard page view for an existing object or the object profile or object cockpit of an existing object, he/she can use any of the configured training phrases. The AlfaBot needs the caption of the object class or object class stereotype, the name of the object, and optionally the caption of a standard page view or object cockpit to process the intent. If the object class or object class stereotype is configured to allow navigation via the AlfaBot and the user profile configuration the user is currently logged in with provides access to the view, the AlfaBot will open the defined view. If the user does not provide information about a standard page view or object cockpit to open, the object profile or first defined object cockpit will open. If the object class or object class stereotype is configured to deny navigation via the AlfaBot, the AlfaBot will inform the user.
- **Edit Artifact**: If the user wants to edit an existing object, he/she can use any of the configured training phrases. The AlfaBot needs the caption of the object class or object class stereotype and the name of the object to process the intent. If the object class or object class stereotype is configured to allow editing via the AlfaBot, the AlfaBot will open the editor or wizard for the object. If the user profile of the user is configured to open a wizard for editing the object, the user can add the caption of the wizard view he/she would like to edit to the request. The wizard will then open displaying the requested wizard step. After the user has edited the object, the object profile or object cockpit of the edited object will open in the Alfabet user interface. If the object class or object class stereotype is configured to deny editing via the AlfaBot, the AlfaBot will inform the user.
- **Navigate to Diagram**: If the user wants to open a configured report, he/she can use any of the configured training phrases. The AlfaBot identifies the diagram via a combination of diagram name, diagram type, and the name of the object that the diagram was created for. The AlfaBot will open the requested diagram.
- **Navigate to Report**: If the user wants to open a configured report, he/she can use any of the configured training phrases. The AlfaBot needs the caption of the report to process the intent. The AlfaBot will open the requested configured report.
- **Start Workflow**: If the user wants to start a workflow, he/she can use any of the configured training phrases. Workflows can be started via the AlfaBot for all workflow templates that have been enabled to

start via the AlfaBot. In the AlfaBot, a user can ask to start a workflow for an existing object by entering the object class stereotype and object name. If multiple workflows can be started for the object, these will be listed in the AlfaBot and the user can start the relevant workflow. Alternatively, the user can ask to start a workflow by entering the workflow caption. Multiple captions defined to start a workflow with new objects or with existing objects will be taken into account. In this case, the AlfaBot will request information about the object that the workflow shall be started with. The caption of the workflow started via the AlfaBot is a concatenation of the caption configured for the workflow and the name of the object that the workflow has been started for.

- **Analyze**: If a user searches for specific information in reports that may not have relevant information in the title or description, he/she can use any of the configured training phrases. For example, a user searching for information about the usability of the enterprise's applications could find the report "Market Readiness of Applications" which includes an indicator for usability even though the term "usability" is neither in the title or description of the report.
- **Provide Keywords**: If the AlfaBot is not able to identify a configured report in the **Navigate to Report** request, for example, the user will be asked to provide keywords to identify the report. The user can answer this request with any of the configured training phrases for this request or provide the keywords without additional text. If multiple keywords are provided, they can be separated with either a comma, the word "`and`", or an ampersand (&).
- **Provide Class or Stereotype Name**: If the AlfaBot is not able to map the value of a placeholder that should return an object class caption or object class stereotype caption to any of the available object classes or object class stereotypes, the user will be asked to provide a correct caption. The user can answer this request with any of the configured training phrases for this request or provide the object class or object class stereotype caption without additional text.
- **Provide Object Name**: If the AlfaBot is not able to map the value of a placeholder that should return an object name to any of the available objects, the user will be asked to provide a correct object name. The user can answer this request with any of the configured training phrases for this request or provide the object name without additional text.
- **Fallback**: This intent will be executed if the phrase that the user enters in the AlfaBot cannot be mapped to any of the other intents. The AlfaBot will answer with any of the configured responses.
- **Provide Positive Confirmation**: If the AlfaBot requests a confirmation from the user that the interpretation of the user request is correct, the user can answer with any of the configured training phrases for this intent to confirm.
- **Provide Negative Confirmation**: If the AlfaBot requests a confirmation from the user that the interpretation of the user request is correct, the user can answer with any of the configured training phrases for this intent to confirm.
- **Cancel Conversation**: Phrases that the user might enter to indicate that he/she would like to end the conversation in the AlfaBot window. By default, no response is returned and the AlfaBot will close without a prior response. Optionally, responses can be configured. The AlfaBot will answer with any of the configured responses prior to ending the conversation.
- **Ask for Help**: Phrases that the user might enter if he/she does not know how to work with the AlfaBot. The AlfaBot will answer with any of the configured responses.

If you click an intent in the explorer, a workspace opens that lists all training phrases and, if applicable, responses that are available in a table. All training phrases are listed by default in ascending alphanumerical order. You can change the sort order by clicking the header of a column in the table. The table provides the following information:

- **Training Phrases** / **Responses**: The text of the training phrase or response is listed in the column. If an intent can have both training phrases and responses, the table will be an expandable dataset with a separate section for training phrases and responses.
- **Synchronized**: The column displays a selected checkbox for all training phrases and responses that have been sent to Dialogflow. If the selected checkbox is not displayed, the training phrase can currently not be used for the intent. Click the **Synchronize** button in the toolbar to send the phrases to Dialogflow.
- **Reserved**: If the column displays a selected checkbox, the training phrase or response is a standard phrase that cannot be edited or deleted.

The AlfaBot is preconfigured to understand standard training phrases. Some of these are reserved to ensure a basic functionality of the AlfaBot and cannot be changed or deleted. If the training phrases provided by default do not match the wording that is typically used by your users, you can add typical phrases that you expect to be entered in the AlfaBot for the selected intent.

It is recommended to use the **AlfaBot Interaction Analysis** functionality to analyze the outcome of user input in the AlfaBot and to decide about new training phrases based on actual user input. For more information, see the section Training the AlfaBot to User Input in the reference manual *User and Solution Administration*.

To create a new training phrase or a new response for the **Welcome**, **Fallback**, **Cancel Conversation**, and **Ask for Help** intents:

1) Click the **Intents** root node of the explorer.
2) Click the **Turn Configuration Mode On** button.

   If the button is called **Turn Configuration Mode Off**, the configuration mode is already turned on. In configuration mode, training phrases can be edited, and user input is exclusively processed via Dialogflow. Internal pre-processing based on existing intents is deactivated.

3) In the explorer, click the intent for which you want to add training phrases.
4) In the toolbar of the intent's workspace, click **New** > **Training Phrase** to enter a training phrase or **New** > **Response** to enter a response. An editor opens.
5) Enter text in the **Training Phrase** field for the training phrase or the **Response** field for the response. For a training phrase, enter a phrase that the user can enter to execute the intent. The training phrase must fulfill the following criteria:
   - The training phrase must be written in English.
   - The training phrase must not contain special characters and punctuation. A mechanism is implemented that strips the following characters from the training phrase or response on synchronization: comma (,) dot (.), semicolon (;), question mark (?), exclamation mark (!), slash (/), backslashes followed by quotation mark or single quotation mark (\', \"), curly brackets ({,}), parenthesis ((,)), square brackets ([,]), angle brackets (<,>), plus sign (+), equal sign (=), asterisk (*), underscore (_), dollar sign ($), hash (#), circumflex (^), percent sign (%).
   - The guidelines for the intent provide information about the placeholders that can be used in the training phrase. A training phrase can only include each placeholder once. It can include only a subset or none of the placeholders listed in the guidelines. The AlfaBot will then ask the user to provide the missing information.

     Typically, a placeholder is used for names of for example objects, object classes, or configured reports. An exception of this rule is the `Analyze` intent where the placeholder `@report` does not stand for the name of a configured report but the complete

information the user is looking for. Standard training phrases for this intent are for example "What are `@report`" or "Where are `@report`". This ensures that most of the user input is analyzed via the search mechanism.

- The training phrase must not consist of only placeholders. The identification of a placeholder without text is managed by a separate mechanism that does not require training phrases and will not work properly if a training phrase consists of only placeholders.

  The identification of only placeholder values as input requires additional configuration described below.

  For a response, enter a phrase that the AlfaBot can return to the user as a result of the intent. The phrase must not include placeholders and it must be written in English language.

6) Click **OK**. The training phrase or response is added to the table. The **Synchronized** column is empty.
7) In the toolbar, click **Synchronize**. The training phrase or response is sent to Dialogflow. The **Synchronized** column now displays a checkmark.
8) Click the **Intents** root node of the explorer.
9) Click the **Turn Config Mode Off** button.

Existing training phrases or responses that are not reserved can be edited via the **Edit**  button or deleted via the **Delete**  button. Both actions require that the changes to the training phrases for the intent are synchronized with Dialogflow via the **Synchronize** button afterwards.

Each time a customer-defined training phrase is changed or added in the AlfaBot configuration functionality or a standard training phrase is added or changed via an update of the meta-model (for example, on upgrade to a new Alfabet release), an example question will be added to the internal list of stored user input in the AlfaBot. This reduces the number of requests sent to Dialogflow®. If a user enters a request corresponding to the new training phrase in the AlfaBot, the internal pre-processing can interpret the request based on the example phrase and does need to send this request to Dialogflow®. The example phrases are not visible in the **AlfaBot Interaction Analysis** functionality.

Deactivating Intents

If an intent is deactivated, it is not taken into account by the AlfaBot when trying to match the user input to an intent. In addition, it is removed from the list of intents displayed to the user on first opening of the AlfaBot in a user session. For example, for companies that have not implemented workflows, it would be confusing for the users to see the intent about start of a workflow in the intent list of the AlfaBot and the workflow intent should therefore be deactivated. Only intents listed on start of the AlfaBot can be deactivated.

1) Click the root node of the explorer. A view with a table listing relevant intents and the information about activation or deactivation is displayed. By default, all intents are activated and show a hook in the column **Active**.
2) In the table listing the intents, select one or multiple intents that you would like to deactivate.
3) In the toolbar above the table, select **Deactivate**.
4) End the user session and re-start the Alfabet Web Application to apply the changes.

To re-activate a deactivated intent, select the intent in the table and click **Activate** in the toolbar. Re-activation also requires a re-start of the Alfabet Web Application to apply the changes.

Updating Entity Values for Intents

For some intents, the user may enter a placeholder value only. For example, if the AlfaBot asks the user to provide the name of an object class, it is highly probable that a user will not enter "The name is application", but "application" only.

Placeholders in training phrases correspond to entities in Dialogflow. Some of the entities for the intents that are preconfigured for Alfabet not only provide a name for the placeholder in the training phrase, but also provide a range of allowed content values. For example, for object classes, a list of object class captions is stored directly in Dialogflow. Captions of standard page views, wizard steps, and configured reports are also stored in the respective entities.

If a user enters text in the AlfaBot, Dialogflow first compares the text with the available training phrases. If no matches are found, the text is compared with the list of values defined for each entity. If the entry matches an entity value, the information is processed accordingly.

On update of the meta-model via AMM files, new entity values based on the changes applied to the meta-model are directly sent to Dialogflow if the connection to Dialogflow is configured and active while updating the meta-model. Customer configuration of the current database directly performed in Alfabet Expand must be updated manually. This includes stereotype definition, configured reports, and re-naming of standard page view captions and object class names via the vocabulary or via view customization.

Click the **Update AlfaBot Entities** button in the workspace of the root node of the explorer to update entity values for Alfabet -specific intents with strings resulting from configuration of the meta-model.

Running the AlfaBot in Offline Mode

The connection of the AlfaBot to Dialogflow can be deactivated temporarily or permanently if the high number of already processed intents enables the AlfaBot to handle user requests on basis of pre-processed intents only. The connection can be deactivated until the next web server restart in the **AlfaBot Configuration** functionality.

Alternatively, a solution designer can deactivate the connection to DialogFlow in the configuration of the AlfaBot in Alfabet Expand. The AlfaBot will then remain offline even after a web server restart. For more information, see *Running the AlfaBot in Offline Mode* in the reference manual *Configuring Alfabet with Alfabet Expand*.

To deactivate the connection until the next Web Server restart:

1) In the **AlfaBot Configuration** functionality click the button **Use AlfaBot Offline**.

   The Alfabet Web Application will check on each Web Server restart whether DialogFlow is accessible and will return to online mode if the check is positive.

Training the AlfaBot to User Input

The **AlfaBot Interaction Analysis** functionality allows the success of user input to be analyzed to enhance the future success rate for the AlfaBot when defining new training phrases.

The administrator responsible for the AlfaBot functionality can understand which user input delivered a result and which user input could not be processed successfully by the AlfaBot based on the data displayed in the table.

The table provides the following information about the phrases that a user entered in the AlfaBot:

- **Phrase**: The user input.

- **Intent**: The intent identified to match the user input.
- **NLP Provider**: `Dialogflow` is displayed if the phrase was analyzed by Dialogflow and `System` is displayed if the intent has been identified during Alfabet internal pre-processing without sending the user input to Dialogflow.
- **Match Count**: For phrases analyzed by Dialogflow and internal pre-processing later used for user input, the number of internal pre-processing based on this Dialogflow analysis is displayed.
- **User**: The user name of the user that entered the phrase.
- **Response**: The response of the AlfaBot as a reaction to user input.
- **Fallback Triggered**: A checkmark is displayed if the intent analyzed by Dialogflow returns an intent and entity specifications that cannot be processed. For example, a **Navigate to Report** intent returned with the entity identified as object class caption instead of as report caption will trigger a fallback.
- **Analyzed**: A checkmark is displayed for phrases marked as analyzed by the administrator via the **Mark as Analyzed** button.

  Please note that the **Show Analyzed Phrases** filter is set to hide rows marked as analyzed per default.

- **Timestamp**: The time that the user input has been entered in the AlfaBot.

The following filters allow you to analyze the user input for intents that require adaptation of the learning phrases:

- **Analysis Type**: Specify to filter for user input that did not end in a requested activity. You can set the filter to **All** to display all user input independent of the success of the request or select a relevant activity status to display the user input that ended with the selected result. The table will display a grouped dataset with the conversation on the first level and the problematic phrase on the second level.
  - **Abandoned**: No proper response was returned for the user input.
  - **Fallback**: Either the user entered a phrase for which the **Fallback** intent was identified, or a fallback was triggered by Alfabet internal mechanisms because the analysis returned by Dialogflow could not be resolved with a meaningful response. For example, a **Navigate to Report** intent returned with the entity identified as an object class caption instead of as a report caption will trigger a fallback.
  - **Canceled**: AlfaBot conversations canceled by the user.
  - **No Response**: No response was returned for the user input.
- **User**: Select a user to view only phrases entered by the specified user.
- **From**: Select a date in the calendar to view only phrases entered by a user on or after the specified date.
- **To**: Select a date in the calendar to view only phrases entered by a user on or before the specified date.
- **Show Analyzed Phrases**: Select the checkbox to view phrases that you have marked as analyzed via the **Mark as Analyzed** button in the toolbar.

To view user input in the context of the complete conversation, select the phrase in the table and click the **View Conversation**  button in the toolbar.

To train the AlfaBot to understand user input that currently failed to provide the requested action:

1) Analyze the user input to identify the correct intent.
2) Add the user input as a training phrase to the identified intent in the **AlfaBot Configuration** functionality. Resolve variables like object names and view captions to the correct placeholders at the correct position.

   Changes made to training phrases and intents in the **AlfaBot Configuration** functionality will be taken into account during the internal pre-processing of intents. If a training phrase is changed, all pre-processed intents that are based on this training phrase will be deleted and a new pre-processed intent will be created as an example for the new training phrase.

   For more information about adding training phrases to an intent, see Creating Training Phrases for Intents.

3) Return to the **AlfaBot Interaction Analysis** functionality, select the user input you added as a training phrase and do one of the following to remove the processed user input from the table:

   - To delete the user input, click the **Delete**  button in the toolbar. The phrase is deleted from the Alfabet database. If the phrase was analyzed internally based on prior user input, the prior user input will also be removed.
   - To retain the user input but mark it as processed, click the **Mark as Analyzed**  button in the toolbar. The phrase will still be stored in the Alfabet database, but it will not be displayed any longer in the table unless you select the **Show Analyzed Phrases** checkbox in the filter area and click the **Update** button.

Chapter 26: Scheduling ADIF Jobs and Batch Jobs via the Job Schedule Functionality

Batch processes that have to be executed regularly can be scheduled for execution via the **Job Schedule** functionality on the Alfabet user interface. The following processes can be scheduled for regular execution:

- ADIF Export
- ADIF Import
- Rescan of indicators
- Rescan of color rules
- Update of full-text search indices
- Creating publications
- Starting a workflow
- Triggering monitors
- Clearing ADIF session content

Please note that the **Job Schedule** functionality requires pre-configuration. For more information, see *Activating the Job Schedule Functionality* in the reference manual *Configuring Alfabet with Alfabet Expand*.

The following information is available:

Scheduling ADIF Jobs and Batch Jobs

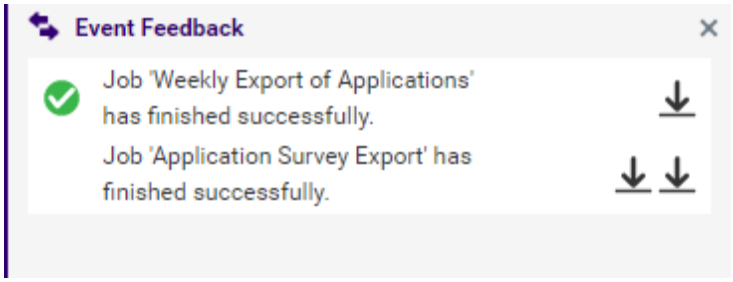
In the **Job Schedule** functionality, ADIF schemes and selected Alfabet batch jobs can be scheduled for execution either once at a defined date and time or in regular intervals. The user can define a job schedule providing details about the execution and the execution time table.

Please note that the **Job Schedule** functionality requires pre-configuration. For more information, see *Activating the Job Schedule Functionality* in the reference manual *Configuring Alfabet with Alfabet Expand*.

Jobs are are executed on server side and therefore the local time of the server is relevant for scheduling and displayed in all time specifications in the view. The current server time when opening the **Job Schedule** view is displayed above the filter area as reference.

After a job schedule is created, an event is scheduled that is in the state **Pending** and has a defined execution date. The job scheduler will scan the event queue in regular intervals for events that are due for execution. If an event is due, the ADIF job is executed and for recurrent job schedules, a new event is scheduled for the next execution.

Each time a scheduled job has been executed, the user that owns the job schedule will see a slide-in **Event Feedback** message window informing him/her whether the current execution was successful.



In the event feedback windows, successfully executed processes are listed with a checkmark in a green field while failed processes are listed with a red cross. On the right of the message, log files and process results are available via download links. Move the cursor over the download link symbol to view a tooltip describing which content is available. Click on the download link symbol to access the data.

The success of the event is only checked on change to another view or on refresh of the current view. Therefore, you will not see the event feedback message if you stay inactive on the current view. If you trigger a process and then change to another session, for example by logging out and logging in again, you will still be informed via the event feedback message when the triggered process is finished.

Clicking anywhere outside the event feedback message closes the window. It can be re-opened via the slide-in toolbar as long as you remain working in the current view. After changing to another view, the message can no longer be opened separately but will be available via the **My Last Event Feedback** notch in the slide-in toolbar containing a list of all event feedback messages that were presented to the current user during the last 24 hours.

The functionality lists all job schedules that have been defined. The table is an expandable dataset with 6 levels.

- The first level is the grouping level for the job schedule stereotype. There is one group for each job schedule stereotype. Currently, the following job schedule stereotypes are available:

- **ADIF Export Job Schedule** for execution of ADIF export.
- **ADIF Import Job Schedule** for execution of ADIF import.
- **Rescan Indicator Job Schedule** to re-compute all automatically computed indicators.
- **Rescan Color Rule Job Schedule** to update coloring based on color rules for new and changed objects.
- **Full Text Search Utility Job Schedule** to update the full text search index to current changes in the Alfabet database.
- **Alfa Batch Executor Job Schedule** for creating assignments and/or sending automatic email notifications to an object's authorized user if a monitor is triggered, an assignment approaches or reaches a defined due date, or an organizational or process change has occurred that affects an object.
- **Alfa Workflow Job Schedule** for starting workflows configured to be started automatically, for closing workflow steps configured to be closed automatically, for automatic deletion of finished workflows, and for rescanning and updating roles and responsibilities for workflows and workflow steps.
- **Publication Job Schedule** for publishing data based on a configured publication.
- **Clear ADIF Session Content Job Schedule** for removing old entries from the `ADIF_SESSION` table.

- The second level displays all defined job schedules that belong to the job schedule stereotype.
- The third level is the grouping level for all events of the job schedule.
- The fourth level displays each execution of a job with the job schedule execution status.

  Only the last 10 executions are displayed. If you would like to see all executions, click on the job schedule name in the third level and click the **Navigate** button. In the object view of the job schedule, click **Job Execution Status**.

- The fifth level is the grouping level for the job run status.
- The sixth level displays the job schedule execution status and provides the ability to open a log file returned from the executed ADIF scheme.

The table provides the following information about the execution of the job schedules:

- **Type**: The column displays the job schedule stereotype as group level for all stereotypes for that job schedules are available. The **Type** column displays **Events** as group level for all events of a job schedule. The **Type** column displays **Job Run Status** for the group level of the job target execution status.
- **Job Schedule Name**: The name of the job schedule. The name is defined by the user that creates the job schedule.
- **Stereotype**: The stereotype of the job schedule.
- **Next Run Time**: The time the job triggered by the job schedule will be executed because of this event.
- **Event Status**: The status of the event. The status can be one of the following:

  - **Pending**: The job is scheduled for execution at a later point of time.
  - **Executing**: The job is currently executed.
  - **Finished**: Job execution has been successful. If an event has the status **Finished**, the scheduling of the job was finished successfully, and a **Job Run Status** section is added as subordinate information level. The execution status in the Job Run Status section should then be checked to see whether execution of the scheduled functionality has been successful, too.
  - **Error**: An error related to scheduling of the job or triggering of the functionality to be executed has occurred.
  - **Execution Error**: An error has occurred during execution of the functionality.
- **Occurrence Time**: The time when the job event was scheduled. A job event is scheduled directly after creation of the job schedule for the first execution. For recurrent schedules, new event is scheduled for the next execution as soon as the first execution is done.
- **Execution Time**: The time when the execution of the job triggered by the event was started.
- **Completion Time**: The time when the execution of the job triggered by the event has finished.

A filter is available on top of the table to search for specific events in the table:

- **Stereotype**: Select a job schedule stereotype to view only job schedules that belong to the selected stereotype.
- **Name**: Enter the name of a job schedule to view only events for the selected job schedule.
- **Job Schedule Event Status**: Select a job schedule status to view only events that are currently in the selected status. Please note that this filter is only applied to the event status of the scheduling events listed in the **Events** sections.
- **Execution After**: Enter a date to view only job schedule events executed at or after the selected date.
- **Execution Before**: Enter a date to view only job schedule events executed at or before the selected date.

You can do one of the following for existing job schedules.

- To view the log file of a job executed via an event, click the job run status in the table and click **Show Log** in the toolbar.
- To cancel a pending event, select the event in the table and click **New** > **Cancel Planned Job**.
- To cancel all pending events for a job schedule, select the job schedule in the table and click **New** > **Cancel All Planned Jobs**.
- After having cancelled pending events, you should select the job schedule in the table and click **New** > **Resume Job Schedule** to resume the execution.
- To shift the execution time of all currently scheduled jobs by one or multiple hours, click **Action** > **Shift Execution Time for All Jobs** in the toolbar and select the amount of time in hours each job should be shifted in the editor that opens. You can define a negative integer to execute jobs earlier. This action will stop all pending job executions, change the start time defined for each job schedule and re-schedule the jobs for execution.
- To schedule an event for immediate execution of the job independent from the defined execution time-table for the job schedule, select the job schedule in the table and click **New** > **Execute Job Schedule** in the toolbar. This option can be used to do one of the following:

- If a job schedule has reached the end date of its execution cycle or has been configured to trigger a single execution, the ADIF job or batch job triggered by the job schedule can be scheduled for immediate execution using the **Execute Job Schedule** option.
- If pending events for a job schedule were cancelled, the **Execute Job Schedule** option can be used instead of the **Resume Job Schedule** option to resume the execution. While the **Resume Job Schedule** option will resume the execution at the next scheduled execution date, the **Execute Job Schedule** option will execute the job immediately independent from the configured execution times in the job schedule and at the same time create an event for the execution at the next scheduled execution date.

- To delete a job schedule, select the job schedule in the table and click the **Delete** button in the toolbar.
- To change a job schedule, select the job schedule in the table and click the **Edit** button in the toolbar.

  If you change a job schedule with a pending event, the pending event will be canceled and a new event will be scheduled that contains the changes you made in the editor.

- To create a new job schedule on basis of the data of an existing job schedule, select the job schedule that shall be used as template in the table and click **New** > **Create Job Schedule as Copy** in the toolbar.

To create a job schedule, do one of the following:

- Creating a Job Schedule for ADIF Export
- Creating a Job Schedule for ADIF Import
- Creating a Job Schedule for the Generation of a Full-Text Search Index
- Creating a Job Schedule for Re-Computing of Indicators
- Creating a Job Schedule for Updating of Coloring Based on Color Rules
- Creating a Job Schedule for Workflow
- Creating a Job Schedule to Send Emails for Monitors, Assignments and Organizational Changes
- Creating a Job Schedule for Batch Deletion of Old ADIF Session Information

Creating a Job Schedule for ADIF Export

Please note that files resulting from scheduled ADIF export can only be exported to the **Internal Document Selector** of the Alfabet database. They can be downloaded from the **Internal Document Selector** in the **Internal Documents** functionality or via a RESTful service call to the `idocdownload` end point of the Alfabet RESTful services.

To schedule an ADIF export job:

1) In the toolbar, click **New** > **Create Schedule**.
2) In the **Stereotype Selector** that opens, select **ADIF Export Job Schedule**.
3) In the editor, define the following attributes:

Basic Data tab:

- **Name**: Define a name for the job schedule. The job schedule will be listed in the table of the **Job Schedule** functionality with this name. The name must be unique. It is not possible to define two job schedules with the same name, even if the type of job schedule is different. This attribute is mandatory.
- **Description**: Provide a meaningful description about the purpose of the job schedule.
- **Verbose Logging**: Select the checkbox if additional information about the running process shall be logged. Verbose logging is in most cases not required and can lead to a decrease in performance. The default setting for this attribute is `False`.
- **ADIF Export Scheme**: Select the ADIF export scheme that shall be executed via the job schedule from the drop-down list.

  Only ADIF schemes that are assigned to a job schedule category can be executed via the **Job Schedule** functionality. For more information see *Configuration Required for Scheduling ADIF Jobs* in the reference manual *Configuring Alfabet with Alfabet Expand*.

- **Parameters**: If the selected ADIF export scheme includes parameter definitions, the attribute displays a table that informs about the name, type and default value for the defined parameters and whether the parameters are mandatory. Enter a parameter value at least for all parameters not having a default defined into the fields in the column **Parameter Value**. Note the following about the definition of parameter values:
  - Parameter values for data types like string or date are defined without single quotes at the beginning and end. If single quotes are required in the query for the data type, they will be automatically added by the ADIF mechanisms.
  - For the definition of dates enter the date in the **Value** field in the format defined in the culture setting of the language you are currently using to render the Alfabet user interface.
  - String array values must be separated with \r\n.
  - % can be used as wildcard in strings and texts. It is not allowed to define a wildcard in a value of a string array.
  - The definition of values for string arrays and reference arrays is currently not supported.

Schedule tab:

- **Schedule Time**: Enter the start time for the job execution into the **Start Time** field.
- **Recurrence Pattern**: Select one of the following check boxes and provide additional data, if applicable:
  - **Hourly**: The interval between job executions is the time selected in the field **of every**.
  - **Daily**: The interval between job executions is the number of days specified in the field **day(s)** as an integer.
  - **Weekly**: The interval between job executions is the number of weeks specified in the field **week(s) on** as an integer. In addition, select the checkbox of the day of the week that the job shall be executed for each recurrence.
  - **Monthly**: There are two methods to select from. If you select the checkbox **Date-Based**, define a day of the month in the **day** field behind the checkbox. For example, if you want the

job to be executed on the fifteenth of each month, enter 15. If you select the checkbox **Weekday-Based**, you can schedule the job for a specific day of a specific week in the month. In the first field behind the checkbox, select the week of the month the job shall be executed. In the second field behind the checkbox, select the day of the week the job shall be executed. The interval between job executions is the number of weeks specified in the field **month(s)** field of the selected scheduling method.

- **Once**: The job is only executed once.

- **Range of Recurrence**: This definition is mandatory. Select the start date for the job schedule from the calendar in the **Start Date** field. For the definition of the end of the recurrence period, either select **End after:** and define the maximum number of occurrences in the **Occurrences** field or select **End by:** and select the end date for the job scheduling from the calendar in the field next to the checkbox entry. If you have selected **Once** in the **Recurrence Pattern** field, only define the date of execution via the **Start Date** field.

File Info tab: This tab is only relevant for export to file.

- **File Name Base**: Enter the file name without extension for the ZIP file that shall contain the exported data. If you do not define the name, the file name will be the name of the executed ADIF export scheme in capital letters.

- **File Suffix**: Select from the drop-down list whether and how the **File Name Base** shall be complemented with a unique marker on export. This setting is relevant for regular exports that should each generate a file without overwriting the previously exported files. The file name can either be complemented with a timestamp or with a unique GUID.

- **IDOC Folder to Export**: Select the checkbox on the right of the folder in the **Internal Document Selector** the file shall be stored in. The list does not display the complete structure of the **Internal Document Selector** but only the folders defined to be used for job scheduling. The file names displayed in the **IDOC Folder to Export** field can deviate from file names in the **Internal Document Selector**, because a show name can be configured for the folders in Alfabet Expand.

  For more information, see *Configuring Access Permissions to Folders in the Internal Document Selector for the Job Schedule* in the reference manual *Configuring Alfabet with Alfabet Expand*.

  If no folder is selected, the file is saved in a folder ADIF_SYS on the root level of the **Internal Document Selector**.

Execution Info tab:

This tab provides the ability to adapt the job scheduling to environmental conditions, like for example the scheduling of other jobs or maintenance windows during which execution shall be postponed.

- **Maximum Wait Time [min]**: Enter a maximum time in minutes that the job execution can be postponed if one of the following applies:

  - The execution of jobs for the current job schedule depends on the execution of one or multiple other jobs to be finished because the execution result of one job is required to be available to execute this job or because both jobs perform conflicting actions. If a job of the current schedule shall be executed while a job defined in the **Dependent Jobs** field is still running, the job will not be executed but re-scheduled for execution five minutes later until the dependent jobs are finished.

- A job for the current job schedule cannot be executed because the current execution time lies within a maintenance window. Maintenance windows are time periods that are blocked to avoid interruption of the job by maintenance work. If a job is due within a maintenance window period, the job will not be executed immediately. If the maintenance window is configured to re-schedule jobs, it will be re-scheduled for execution one minute after the end of the maintenance window. If the maintenance window is configured to skip jobs, the job execution during the maintenance window will be skipped and the next execution of the job is scheduled according to the settings in the **Schedule** tab of the job schedule editor.

  Maintenance windows must have been configured for the Alfabet components in order to adapt job scheduling to the maintenance window via the **Maximum Wait Time [min]** setting. For information about the definition of maintenance windows, see *Defining Maintenance Windows* in the reference manual *Configuring Alfabet with Alfabet Expand* or *Defining Maintenance Windows for Scheduled Jobs* in the reference manual *System Administration*.

If the time between the originally due job execution and the end of the maintenance window or the end of execution of a dependent job exceeds the maximum wait time, job execution will be cancelled, and an error message is written into the log file. The next execution of the job is scheduled according to the settings in the **Schedule** tab of the job schedule editor.

- **Dependent Jobs**: If the execution of jobs for the current job schedule shall be postponed if another job is still running, click **New** > **Create New Job Dependency**. Select the job schedule the current job schedule depends on in the list and click **OK**. If a job is running for one of the dependent job schedules, the execution of the job for the current job schedule will be shifted to the scheduled time plus minutes until the execution of all dependent jobs is finished. If the maximum wait time defined with **Maximum Wait Time [min]** is exceeded while dependent jobs are still running, the execution of the current job is cancelled and the next execution scheduled according to the settings in the **Schedule** tab of the job schedule editor.
- **Expected Execution Time [min]**: If a maximum time for execution of the job is defined in minutes in this field, a warning is written into the log file available via the **Show Log** button if the execution of a job for the current schedule exceeds the maximum wait time. The job is nevertheless further executed.
- **Executing User**: Select a user for which the wakeup events of the job schedule shall be executed. If no user is specified, the user editing the job schedule will be the execution user for the job schedule. Each time a scheduled job has been executed, the executing user will see a slide-in **Event Feedback** message window informing him/her whether the current execution was successful. If a result is available for download, a download link is displayed. By default, the slide-in message will be displayed to the user that created the job schedule when he/she is logged in with the same user profile as on creation of the job schedule. This default behavior can be changed. Select a user from the drop-down list to show the **Event Feedback** message to that user instead. Please note that the selected user must be logged in with the user profile defined in the **Executing User Profile** attribute to receive the **Event Feedback** messages for this job schedule.
- **Executing User Profile**: The user profile that the user defined with the **Executing User** attribute must be logged in with to receive **Event Feedback** messages for execution of this job schedule. By default, a message will be displayed to the user that created the job schedule when he/she is logged in with the same user profile as on creation of the job schedule. This attribute is mandatory if **Executing User** is defined.
- **On Completion Jobs**: You can configure jobs to be started after execution of the current job. This required configuration in both jobs.

- Configure the indirectly triggered job without an own job schedule. In the **Basic Data** tab of this job, select the **Indirectly Triggered** checkbox.
- Click the **New** button above the **On Completion Jobs** table of the current job and select **Add**. Select the job that shall be started on completion of the current job.

4) Click **OK** to save your changes.

After having created the job schedule, you can use it as a template for creating new job schedules. Select the job schedule in the table and click **New** > **Create Job Schedule As Copy** from the toolbar.

Creating a Job Schedule for ADIF Import

For ADIF import from file, the files can either be located on the local file system in a folder accessible for the Alfabet components or in the Internal Document Selector of Alfabet.

To schedule an ADIF import job:

1) In the toolbar, click **New** > **Create Schedule**.
2) In the **Stereotype Selector** that opens, select **ADIF Import Job Schedule**.
3) In the editor, define the following attributes:

   Basic Data tab:

- **Name**: Define a name for the job schedule. The job schedule will be listed in the table of the **Job Schedule** functionality with this name. The name must be unique. It is not possible to define two job schedules with the same name, even if the type of job schedule is different. This attribute is mandatory.
- **Description**: Provide a meaningful description about the purpose of the job schedule.
- **Verbose Logging**: Select the checkbox if additional information about the running process shall be logged. Verbose logging is in most cases not required and can lead to a decrease in performance. The default setting for this attribute is `False`.
- **Import Scheme**: Select the ADIF import scheme that shall be executed via the job schedule from the drop-down list.

  Only ADIF schemes that are assigned to a job schedule category can be executed via the **Job Schedule** functionality. For more information see *Configuration Required for Scheduling ADIF Jobs* in the reference manual *Configuring Alfabet with Alfabet Expand*.

- **Parameters**: If the selected ADIF import scheme includes parameter definitions, the attribute displays a table that informs about the name, type and default value for the defined parameters and whether the parameters are mandatory. Enter a parameter value at least for all parameters not having a default defined into the fields in the column **Parameter Value**. Note the following about the definition of parameter values:
  - Parameter values for data types like string or date are defined without single quotes at the beginning and end. If single quotes are required in the query for the data type, they will be automatically added by the ADIF mechanisms.
  - For the definition of dates enter the date into the Value field in the format defined in the culture setting of the language you are currently using to render the Alfabet user interface.

- String array values must be separated with \r\n.
- % can be used as wildcard in strings and texts. It is not allowed to define a wildcard in a value of a string array.
- The definition of values for reference arrays is currently not supported.

Schedule tab:

- **Schedule Time**: Enter the start time for the job execution into the **Start Time** field.
- **Recurrence Pattern**: Select one of the following check boxes and provide additional data, if applicable:
  - **Hourly**: The interval between job executions is the time selected in the field **of every**.
  - **Daily**: The interval between job executions is the number of days specified in the field **day(s)** as an integer.
  - **Weekly**: The interval between job executions is the number of weeks specified in the field **week(s) on** as an integer. In addition, select the checkbox of the day of the week that the job shall be executed for each recurrence.
  - **Monthly**: There are two methods to select from. If you select the checkbox **Date-Based**, define a day of the month in the **day** field behind the checkbox. For example, if you want the job to be executed on the fifteenth of each month, enter 15. If you select the checkbox **Weekday-Based**, you can schedule the job for a specific day of a specific week in the month. In the first field behind the checkbox, select the week of the month the job shall be executed. In the second field behind the checkbox, select the day of the week the job shall be executed. The interval between job executions is the number of weeks specified in the field **month(s)** field of the selected scheduling method.
  - **Once**: The job is only executed once.
- **Range of Recurrence**: This definition is mandatory. Select the start date for the job schedule from the calendar in the **Start Date** field. For the definition of the end of the recurrence period, either select **End after:** and define the maximum number of occurrences in the **Occurrences** field or select **End by:** and select the end date for the job scheduling from the calender in the field next to the checkbox entry. If you have selected **Once** in the **Recurrence Pattern** field, only define the date of execution via the **Start Date** field.

File Info tab: this tab is only relevant for import from file.

- **File Location**: Select where the import ZIP file is located.
  - **System File**: The file is available on the local file system.
  - **System Folder**: The file is available in a defined folder. The latest file in the folder will be used to perform the import.
  - **IDOC File**: The file is available on the local file system.
  - **IDOC Folder**: The file is available in a defined folder of the Internal Document Selector. The latest file in the folder will be used to perform the import.
- **System File/Folder Path**: If the **File Location** is **System File**, enter the absolute path and name of the file to be imported. If the **File Location** is **System Folder**, enter the absolute path to the folder containing the import file.

- **Internal Documents**: If the **File Location** is **IDOC File**, select the checkbox behind the file in the internal document selector that contains the data to be imported. The file type must be ZIP. If the **File Location** is **IDOC Folder**, select the checkbox behind the folder that will contain the import file.

Execution Info tab:

This tab provides the ability to adapt the job scheduling to environmental conditions, like for example the scheduling of other jobs or maintenance windows during which execution shall be postponed.

- **Maximum Wait Time [min]**: Enter a maximum time in minutes that the job execution can be postponed if one of the following applies:
  - The execution of jobs for the current job schedule depends on the execution of one or multiple other jobs to be finished because the execution result of one job is required to be available to execute this job or because both jobs perform conflicting actions. If a job of the current schedule shall be executed while a job defined in the **Dependent Jobs** field is still running, the job will not be executed but re-scheduled for execution five minutes later until the dependent jobs are finished.
  - A job for the current job schedule cannot be executed because the current execution time lies within a maintenance window. Maintenance windows are time periods that are blocked to avoid interruption of the job by maintenance work. If a job is due within a maintenance window period, the job will not be executed immediately. If the maintenance window is configured to re-schedule jobs, it will be re-scheduled for execution one minute after the end of the maintenance window. If the maintenance window is configured to skip jobs, the job execution during the maintenance window will be skipped and the next execution of the job is scheduled according to the settings in the **Schedule** tab of the job schedule editor.

    Maintenance windows must have been configured for the Alfabet components in order to adapt job scheduling to the maintenance window via the **Maximum Wait Time [min]** setting. For information about the definition of maintenance windows, see *Defining Maintenance Windows* in the reference manual *Configuring Alfabet with Alfabet Expand* or *Defining Maintenance Windows for Scheduled Jobs* in the reference manual *System Administration*.

  If the time between the originally due job execution and the end of the maintenance window or the end of execution of a dependent job exceeds the maximum wait time, job execution will be cancelled, and an error message is written into the log file. The next execution of the job is scheduled according to the settings in the **Schedule** tab of the job schedule editor.

- **Dependent Jobs**: If the execution of jobs for the current job schedule shall be postponed if another job is still running, click **New** > **Create New Job Dependency**. Select the job schedule the current job schedule depends on in the list and click **OK**. If a job is running for one of the dependent job schedules, the execution of the job for the current job schedule will be shifted to the scheduled time plus minutes until the execution of all dependent jobs is finished. If the maximum wait time defined with **Maximum Wait Time [min]** is exceeded while dependent jobs are still running, the execution of the current job is cancelled and the next execution scheduled according to the settings in the **Schedule** tab of the job schedule editor.
- **Expected Execution Time [min]**: If a maximum time for execution of the job is defined in minutes in this field, a warning is written into the log file available via the **Show Log** button if the execution of a job for the current schedule exceeds the maximum wait time. The job is nevertheless further executed.

- **Executing User**: Select a user for which the wakeup events of the job schedule shall be executed. If no user is specified, the user editing the job schedule will be the execution user for the job schedule. Each time a scheduled job has been executed, the executing user will see a slide-in **Event Feedback** message window informing him/her whether the current execution was successful. If a result is available for download, a download link is displayed. By default, the slide-in message will be displayed to the user that created the job schedule when he/she is logged in with the same user profile as on creation of the job schedule. This default behavior can be changed. Select a user from the drop-down list to show the **Event Feedback** message to that user instead. Please note that the selected user must be logged in with the user profile defined in the **Executing User Profile** attribute to receive the **Event Feedback** messages for this job schedule.
- **Executing User Profile**: The user profile that the user defined with the **Executing User** attribute must be logged in with to receive **Event Feedback** messages for execution of this job schedule. By default, a message will be displayed to the user that created the job schedule when he/she is logged in with the same user profile as on creation of the job schedule. This attribute is mandatory if **Executing User** is defined.
- **On Completion Jobs**: You can configure jobs to be started after execution of the current job. This required configuration in both jobs.
  - Configure the indirectly triggered job without an own job schedule. In the **Basic Data** tab of this job, select the **Indirectly Triggered** checkbox.
  - Click the **New** button above the **On Completion Jobs** table of the current job and select **Add**. Select the job that shall be started on completion of the current job.

4) Click **OK** to save your changes.

After having created the job schedule, you can use it as a template for creating new job schedules. Select the job schedule in the table and click **New** > **Create Job Schedule As Copy** from the toolbar.

Creating a Job Schedule for the Generation of a Full-Text Search Index

The full-text search in Alfabet is based on search groups that are defined by a solution designer in Alfabet Expand. For each search group, a search index must be defined and updated in regular intervals to include current changes in the Alfabet database into account.

For more information about the definition of search groups, see *Configuring the Full-Text Search Capability* in the reference manual *Configuring Alfabet with Alfabet Expand*.

There are two types of search groups:

- Globally defined search groups provide a full-text search for objects of defined object classes or subsets of objects of defined object classes. The user can perform the search in the **Full-Text Search** functionality.

  The **Full-Text Search** functionality available to standard users does not provide a means to create and update search indexes. The search indexes for these search groups can be created and updated in regular intervals by means of a job schedule.

  Alternatively, the following methods are available to create and update globally defined search indexes:

- The command line tool `FullTextSearchUtil.exe` can alternatively be used to update the full-text search indexes. For more information, see *Updating Indexes with the FullTextSearchUtil.exe* in the reference manual *System Administration*.
- An administrative user can create and update the indexes in the administrative **Full-Text Search** functionality. For more information, see Creating an Index for the Full-Text Search.

- Object-centric search groups provide a full-text search for objects of defined object classes that are related to a single base object. Any user with access permissions to the base object can perform the search in the **Full-Text Search** page view added to the object profile of the base object class. Object-centric search groups are not accessible via the **Full-Text Search** functionality.

  For object-centric search groups, a separate index is created for each base object. The search index for object-centric search groups is therefore exclusively created and updated directly in the **Full-Text Search** page view in the object profile of the base object by the user performing the search. An index for an object centric search group cannot be created or updated via the job schedule because information about a base object is not provided.

To schedule a job for the update or generation of full-text search indexes:

1) In the toolbar, click **New** > **Create Schedule**.
2) In the **Stereotype Selector** that opens, select **Full-Text Search Utility Job Schedule**.
3) In the editor, define the following attributes:

   Basic Data tab:

   - **Name**: Define a name for the job schedule. The job schedule will be listed in the table of the **Job Schedule** functionality with this name. The name must be unique. It is not possible to define two job schedules with the same name, even if the type of job schedule is different. This attribute is mandatory.
   - **Description**: Provide a meaningful description about the purpose of the job schedule.
   - **Search Group Names**: Select the search group for that an index shall be created or updated. If you do not select a search group, all available search groups are updated.

     Please note that object-centric search groups cannot be created or updated via job schedule and are therefore not included into the list of selectable search groups.

   - **Language**: Select the language for which the full-text search index(es) shall be created or updated. If no language is selected, the English (US) search index(es) will be created or updated.

   Schedule tab:

   - **Schedule Time**: Enter the start time for the job execution into the **Start Time** field.
   - **Recurrence Pattern**: Select one of the following check boxes and provide additional data, if applicable:
     - **Hourly**: The interval between job executions is the time selected in the field **of every**.
     - **Daily**: The interval between job executions is the number of days specified in the field **day(s)** as an integer.

- **Weekly**: The interval between job executions is the number of weeks specified in the field **week(s) on** as an integer. In addition, select the checkbox of the day of the week that the job shall be executed for each recurrence.
- **Monthly**: There are two methods to select from. If you select the checkbox **Date-Based**, define a day of the month in the **day** field behind the checkbox. For example, if you want the job to be executed on the fifteenth of each month, enter 15. If you select the checkbox **Weekday-Based**, you can schedule the job for a specific day of a specific week in the month. In the first field behind the checkbox, select the week of the month the job shall be executed. In the second field behind the checkbox, select the day of the week the job shall be executed. The interval between job executions is the number of weeks specified in the field **month(s)** field of the selected scheduling method.
- **Once**: The job is only executed once.

- **Range of Recurrence**: This definition is mandatory. Select the start date for the job schedule from the calendar in the **Start Date** field. For the definition of the end of the recurrence period, either select **End after:** and define the maximum number of occurrences in the **Occurrences** field or select **End by:** and select the end date for the job scheduling from the calendar in the field next to the checkbox entry. If you have selected **Once** in the **Recurrence Pattern** field, only define the date of execution via the **Start Date** field.

Execution Info tab:

This tab provides the ability to adapt the job scheduling to environmental conditions, like for example the scheduling of other jobs or maintenance windows during which execution shall be postponed.

- **Maximum Wait Time [min]**: Enter a maximum time in minutes that the job execution can be postponed if one of the following applies:
  - The execution of jobs for the current job schedule depends on the execution of one or multiple other jobs to be finished because the execution result of one job is required to be available to execute this job or because both jobs perform conflicting actions. If a job of the current schedule shall be executed while a job defined in the **Dependent Jobs** field is still running, the job will not be executed but re-scheduled for execution five minutes later until the dependent jobs are finished.
  - A job for the current job schedule cannot be executed because the current execution time lies within a maintenance window. Maintenance windows are time periods that are blocked to avoid interruption of the job by maintenance work. If a job is due within a maintenance window period, the job will not be executed immediately. If the maintenance window is configured to re-schedule jobs, it will be re-scheduled for execution one minute after the end of the maintenance window. If the maintenance window is configured to skip jobs, the job execution during the maintenance window will be skipped and the next execution of the job is scheduled according to the settings in the **Schedule** tab of the job schedule editor.

    Maintenance windows must have been configured for the Alfabet components in order to adapt job scheduling to the maintenance window via the **Maximum Wait Time [min]** setting. For information about the definition of maintenance windows, see *Defining Maintenance Windows* in the reference manual *Configuring Alfabet with Alfabet Expand* or *Defining Maintenance Windows for Scheduled Jobs* in the reference manual *System Administration*.

  If the time between the originally due job execution and the end of the maintenance window or the end of execution of a dependent job exceeds the maximum wait time, job execution will be cancelled, and an error message is written into the log file. The next execution of the job is scheduled according to the settings in the **Schedule** tab of the job schedule editor.

- **Dependent Jobs**: If the execution of jobs for the current job schedule shall be postponed if another job is still running, click **New** > **Create New Job Dependency**. Select the job schedule the current job schedule depends on in the list and click **OK**. If a job is running for one of the dependent job schedules, the execution of the job for the current job schedule will be shifted to the scheduled time plus minutes until the execution of all dependent jobs is finished. If the maximum wait time defined with **Maximum Wait Time [min]** is exceeded while dependent jobs are still running, the execution of the current job is cancelled and the next execution scheduled according to the settings in the **Schedule** tab of the job schedule editor.

- **Expected Execution Time [min]**: If a maximum time for execution of the job is defined in minutes in this field, a warning is written into the log file available via the **Show Log** button if the execution of a job for the current schedule exceeds the maximum wait time. The job is nevertheless further executed.

- **Executing User**: Select a user for which the wakeup events of the job schedule shall be executed. If no user is specified, the user editing the job schedule will be the execution user for the job schedule. Each time a scheduled job has been executed, the executing user will see a slide-in **Event Feedback** message window informing him/her whether the current execution was successful. If a result is available for download, a download link is displayed. By default, the slide-in message will be displayed to the user that created the job schedule when he/she is logged in with the same user profile as on creation of the job schedule. This default behavior can be changed. Select a user from the drop-down list to show the **Event Feedback** message to that user instead. Please note that the selected user must be logged in with the user profile defined in the **Executing User Profile** attribute to receive the **Event Feedback** messages for this job schedule.

- **Executing User Profile**: The user profile that the user defined with the **Executing User** attribute must be logged in with to receive **Event Feedback** messages for execution of this job schedule. By default, a message will be displayed to the user that created the job schedule when he/she is logged in with the same user profile as on creation of the job schedule. This attribute is mandatory if **Executing User** is defined.

- **On Completion Jobs**: You can configure jobs to be started after execution of the current job. This required configuration in both jobs.

    - Configure the indirectly triggered job without an own job schedule. In the **Basic Data** tab of this job, select the **Indirectly Triggered** checkbox.

    - Click the **New** button above the **On Completion Jobs** table of the current job and select **Add**. Select the job that shall be started on completion of the current job.

4) Click **OK** to save your changes.

After having created the job schedule, you can use it as a template for creating new job schedules. Select the job schedule in the table and click **New** > **Create Job Schedule As Copy** from the toolbar.

Creating a Job Schedule for Re-Computing of Indicators

Indicator types can be configured to be calculated based on computation rules. For indicators based on these indicators, a user cannot set the indicator to any or a range of allowed values. Instead, he/she can trigger calculation of the indicator values on basis of the underlying computation rules for a single object via the button **Calculate Indicators** on the **Evaluations** page view of the object.

Calculated indicators can also be calculated via a job schedule. The calculation can then be done in batch for all indicators or all indicators of a defined evaluation type for a specific object class or a subset of objects of a specific object class.

Alternatively, the command line tool `RescanIndicatorsConsole.exe` can be used to batch calculate computed indicators. For more information, see *Batch-Calculation of Indicators with RescanIndicatorsConsole.exe* in the reference manual *System Administration*.

To schedule a job for the re-computing of indicators:

1) In the toolbar, click **New** > **Create Schedule**.
2) In the **Stereotype Selector** that opens, select **Rescan Indicator Job Schedule**.
3) In the editor, define the following attributes:

   Basic Data tab:

- **Name**: Define a name for the job schedule. The job schedule will be listed in the table of the **Job Schedule** functionality with this name. The name must be unique. It is not possible to define two job schedules with the same name, even if the type of job schedule is different. This attribute is mandatory.
- **Description**: Provide a meaningful description about the purpose of the job schedule.
- In the **Configuration** field, define the following attributes:
  - **Class**: Select the object class or object class stereotype that indicators shall be re-computed for. The drop-down list includes all object classes which have computed indicators assigned. This setting is mandatory.
  - **Evaluation Type**: Select an evaluation type to limit rescan of indicators to indicators assigned to the defined evaluation type. The drop-down list includes all evaluation types of the selected object class which have computed indicators assigned. This setting is mandatory.
  - **Indicator Type**: Select an indicator type to limit rescan of indicators to indicators assigned to the defined indicator type. The drop-down list includes all computed indicators available for the selected evaluation type. This setting is mandatory.
  - **Report**: If only a subset of objects of the object class defined with the Class attribute shall be computed, select the configured report that returns the relevant subset of objects.

    The drop-down list only displays configured reports that have been configured for the rescan of indicators via job schedule by a solution designer in the configuration tool Alfabet Expand. For information about the definition of configured reports for the rescan indicators job schedule, see *Configurations for Scheduling Rescan of Indicators* in the reference manual *Configuring Alfabet with Alfabet Expand*.

   Schedule tab:

- **Schedule Time**: Enter the start time for the job execution into the **Start Time** field.
- **Recurrence Pattern**: Select one of the following check boxes and provide additional data, if applicable:
  - **Hourly**: The interval between job executions is the time selected in the field **of every**.
  - **Daily**: The interval between job executions is the number of days specified in the field **day(s)** as an integer.

- **Weekly**: The interval between job executions is the number of weeks specified in the field **week(s) on** as an integer. In addition, select the checkbox of the day of the week that the job shall be executed for each recurrence.
- **Monthly**: There are two methods to select from. If you select the checkbox **Date-Based**, define a day of the month in the **day** field behind the checkbox. For example, if you want the job to be executed on the fifteenth of each month, enter 15. If you select the checkbox **Weekday-Based**, you can schedule the job for a specific day of a specific week in the month. In the first field behind the checkbox, select the week of the month the job shall be executed. In the second field behind the checkbox, select the day of the week the job shall be executed. The interval between job executions is the number of weeks specified in the field **month(s)** field of the selected scheduling method.
- **Once**: The job is only executed once.

- **Range of Recurrence**: This definition is mandatory. Select the start date for the job schedule from the calendar in the **Start Date** field. For the definition of the end of the recurrence period, either select **End after:** and define the maximum number of occurrences in the **Occurrences** field or select **End by:** and select the end date for the job scheduling from the calendar in the field next to the checkbox entry. If you have selected **Once** in the **Recurrence Pattern** field, only define the date of execution via the **Start Date** field.

Execution Info tab:

This tab provides the ability to adapt the job scheduling to environmental conditions, like for example the scheduling of other jobs or maintenance windows during which execution shall be postponed.

- **Maximum Wait Time [min]**: Enter a maximum time in minutes that the job execution can be postponed if one of the following applies:
  - The execution of jobs for the current job schedule depends on the execution of one or multiple other jobs to be finished because the execution result of one job is required to be available to execute this job or because both jobs perform conflicting actions. If a job of the current schedule shall be executed while a job defined in the **Dependent Jobs** field is still running, the job will not be executed but re-scheduled for execution five minutes later until the dependent jobs are finished.
  - A job for the current job schedule cannot be executed because the current execution time lies within a maintenance window. Maintenance windows are time periods that are blocked to avoid interruption of the job by maintenance work. If a job is due within a maintenance window period, the job will not be executed immediately. If the maintenance window is configured to re-schedule jobs, it will be re-scheduled for execution one minute after the end of the maintenance window. If the maintenance window is configured to skip jobs, the job execution during the maintenance window will be skipped and the next execution of the job is scheduled according to the settings in the **Schedule** tab of the job schedule editor.

    Maintenance windows must have been configured for the Alfabet components in order to adapt job scheduling to the maintenance window via the **Maximum Wait Time [min]** setting. For information about the definition of maintenance windows, see *Defining Maintenance Windows* in the reference manual *Configuring Alfabet with Alfabet Expand* or *Defining Maintenance Windows for Scheduled Jobs* in the reference manual *System Administration*.

  If the time between the originally due job execution and the end of the maintenance window or the end of execution of a dependent job exceeds the maximum wait time, job execution will be cancelled, and an error message is written into the log file. The next execution of the job is scheduled according to the settings in the **Schedule** tab of the job schedule editor.

- **Dependent Jobs**: If the execution of jobs for the current job schedule shall be postponed if another job is still running, click **New** > **Create New Job Dependency**. Select the job schedule the current job schedule depends on in the list and click **OK**. If a job is running for one of the dependent job schedules, the execution of the job for the current job schedule will be shifted to the scheduled time plus minutes until the execution of all dependent jobs is finished. If the maximum wait time defined with **Maximum Wait Time [min]** is exceeded while dependent jobs are still running, the execution of the current job is cancelled and the next execution scheduled according to the settings in the **Schedule** tab of the job schedule editor.

- **Expected Execution Time [min]**: If a maximum time for execution of the job is defined in minutes in this field, a warning is written into the log file available via the **Show Log** button if the execution of a job for the current schedule exceeds the maximum wait time. The job is nevertheless further executed.

- **Executing User**: Select a user for which the wakeup events of the job schedule shall be executed. If no user is specified, the user editing the job schedule will be the execution user for the job schedule. Each time a scheduled job has been executed, the executing user will see a slide-in **Event Feedback** message window informing him/her whether the current execution was successful. If a result is available for download, a download link is displayed. By default, the slide-in message will be displayed to the user that created the job schedule when he/she is logged in with the same user profile as on creation of the job schedule. This default behavior can be changed. Select a user from the drop-down list to show the **Event Feedback** message to that user instead. Please note that the selected user must be logged in with the user profile defined in the **Executing User Profile** attribute to receive the **Event Feedback** messages for this job schedule.

- **Executing User Profile**: The user profile that the user defined with the **Executing User** attribute must be logged in with to receive **Event Feedback** messages for execution of this job schedule. By default, a message will be displayed to the user that created the job schedule when he/she is logged in with the same user profile as on creation of the job schedule. This attribute is mandatory if **Executing User** is defined.

- **On Completion Jobs**: You can configure jobs to be started after execution of the current job. This required configuration in both jobs.

    - Configure the indirectly triggered job without an own job schedule. In the **Basic Data** tab of this job, select the **Indirectly Triggered** checkbox.

    - Click the **New** button above the **On Completion Jobs** table of the current job and select **Add**. Select the job that shall be started on completion of the current job.

4) Click **OK** to save your changes.

After having created the job schedule, you can use it as a template for creating new job schedules. Select the job schedule in the table and click **New** > **Create Job Schedule As Copy** from the toolbar.

Creating a Job Schedule for Updating of Coloring Based on Color Rules

A color rule is based on one or more queries. Each query finds objects and highlights them by assigning a specified color. Color rules can be assigned to a map view in order to color objects displayed in the map view's *Business Support Map Report* or to a diagram view in order to color objects in standard Alfabet diagrams. A color rule defined for applications, for example, could be displayed in both the business support map as well as in relevant diagrams.

Color rules can be defined in the **Color Rules** view available via the root node of the **Master Plans** explorer or **Diagram Views** explorer or in the **Color Rules Manager** functionality in the **Configuration** module. For more

information, see the section *Creating a Color Rule* in the reference manual *Configuring Evaluation and Reference Data in Alfabet*.

Once a color rule is created, it must be executed regularly to adapt the coloring for objects to recent changes in the database. Update of coloring based on color rules can be performed via a job schedule.

Alternatively, the following methods are available to execute coloring based on color rules:

- The command line tool `RescanColorRules.exe` can be used to execute color rules. For more information, see *Batch Evaluation of Color Rules with RescanColorRules.exe* in the reference manual *System Administration*.
- Color rules can be executed manually in the **Color Rules** view available via the root node of the **Master Plans** explorer or **Diagram Views** explorer or in the **Color Rules Manager** functionality in the **Configuration** module. For more information about the manual activation of the color rule, see the section *Activating the Color Rule and Executing the Query*.

To schedule a job for the update of coloring based on color rules:

1) In the toolbar, click **New** > **Create Schedule**.
2) In the **Stereotype Selector** that opens, select **Rescan Color Rule Job Schedule**.
3) In the editor, define the following attributes:

   Basic Data tab:

   - **Name**: Define a name for the job schedule. The job schedule will be listed in the table of the **Job Schedule** functionality with this name. The name must be unique. It is not possible to define two job schedules with the same name, even if the type of job schedule is different. This attribute is mandatory.
   - **Description**: Provide a meaningful description about the purpose of the job schedule.

   Schedule tab:

   - **Schedule Time**: Enter the start time for the job execution into the **Start Time** field.
   - **Recurrence Pattern**: Select one of the following check boxes and provide additional data, if applicable:
     - **Hourly**: The interval between job executions is the time selected in the field **of every**.
     - **Daily**: The interval between job executions is the number of days specified in the field **day(s)** as an integer.
     - **Weekly**: The interval between job executions is the number of weeks specified in the field **week(s) on** as an integer. In addition, select the checkbox of the day of the week that the job shall be executed for each recurrence.
     - **Monthly**: There are two methods to select from. If you select the checkbox **Date-Based**, define a day of the month in the **day** field behind the checkbox. For example, if you want the job to be executed on the fifteenth of each month, enter 15. If you select the checkbox **Weekday-Based**, you can schedule the job for a specific day of a specific week in the month. In the first field behind the checkbox, select the week of the month the job shall be executed. In the second field behind the checkbox, select the day of the week the job shall be executed.

The interval between job executions is the number of weeks specified in the field **month(s)** field of the selected scheduling method.

- **Once**: The job is only executed once.

- **Range of Recurrence**: This definition is mandatory. Select the start date for the job schedule from the calendar in the **Start Date** field. For the definition of the end of the recurrence period, either select **End after:** and define the maximum number of occurrences in the **Occurrences** field or select **End by:** and select the end date for the job scheduling from the calendar in the field next to the checkbox entry. If you have selected **Once** in the **Recurrence Pattern** field, only define the date of execution via the **Start Date** field.

Execution Info tab:

This tab provides the ability to adapt the job scheduling to environmental conditions, like for example the scheduling of other jobs or maintenance windows during which execution shall be postponed.

- **Maximum Wait Time [min]**: Enter a maximum time in minutes that the job execution can be postponed if one of the following applies:

  - The execution of jobs for the current job schedule depends on the execution of one or multiple other jobs to be finished because the execution result of one job is required to be available to execute this job or because both jobs perform conflicting actions. If a job of the current schedule shall be executed while a job defined in the **Dependent Jobs** field is still running, the job will not be executed but re-scheduled for execution five minutes later until the dependent jobs are finished.

  - A job for the current job schedule cannot be executed because the current execution time lies within a maintenance window. Maintenance windows are time periods that are blocked to avoid interruption of the job by maintenance work. If a job is due within a maintenance window period, the job will not be executed immediately. If the maintenance window is configured to re-schedule jobs, it will be re-scheduled for execution one minute after the end of the maintenance window. If the maintenance window is configured to skip jobs, the job execution during the maintenance window will be skipped and the next execution of the job is scheduled according to the settings in the **Schedule** tab of the job schedule editor.

    Maintenance windows must have been configured for the Alfabet components in order to adapt job scheduling to the maintenance window via the **Maximum Wait Time [min]** setting. For information about the definition of maintenance windows, see *Defining Maintenance Windows* in the reference manual *Configuring Alfabet with Alfabet Expand* or *Defining Maintenance Windows for Scheduled Jobs* in the reference manual *System Administration*.

  If the time between the originally due job execution and the end of the maintenance window or the end of execution of a dependent job exceeds the maximum wait time, job execution will be cancelled and an error message is written into the log file. The next execution of the job is scheduled according to the settings in the **Schedule** tab of the job schedule editor.

- **Dependent Jobs**: If the execution of jobs for the current job schedule shall be postponed if another job is still running, click **New** > **Create New Job Dependency**. Select the job schedule the current job schedule depends on in the list and click **OK**. If a job is running for one of the dependent job schedules, the execution of the job for the current job schedule will be shifted to the scheduled time plus minutes until the execution of all dependent jobs is finished. If the maximum wait time defined with **Maximum Wait Time [min]** is exceeded while dependent jobs are still running, the execution of the current job is cancelled and the next execution scheduled according to the settings in the **Schedule** tab of the job schedule editor.

- **Expected Execution Time [min]**: If a maximum time for execution of the job is defined in minutes in this field, a warning is written into the log file available via the **Show Log** button if the execution of a job for the current schedule exceeds the maximum wait time. The job is nevertheless further executed.
- **Executing User**: Select a user for which the wakeup events of the job schedule shall be executed. If no user is specified, the user editing the job schedule will be the execution user for the job schedule. Each time a scheduled job has been executed, the executing user will see a slide-in **Event Feedback** message window informing him/her whether the current execution was successful. If a result is available for download, a download link is displayed. By default, the slide-in message will be displayed to the user that created the job schedule when he/she is logged in with the same user profile as on creation of the job schedule. This default behavior can be changed. Select a user from the drop-down list to show the **Event Feedback** message to that user instead. Please note that the selected user must be logged in with the user profile defined in the **Executing User Profile** attribute to receive the **Event Feedback** messages for this job schedule.
- **Executing User Profile**: The user profile that the user defined with the **Executing User** attribute must be logged in with to receive **Event Feedback** messages for execution of this job schedule. By default, a message will be displayed to the user that created the job schedule when he/she is logged in with the same user profile as on creation of the job schedule. This attribute is mandatory if **Executing User** is defined.
- **On Completion Jobs**: You can configure jobs to be started after execution of the current job. This required configuration in both jobs.
  - Configure the indirectly triggered job without an own job schedule. In the **Basic Data** tab of this job, select the **Indirectly Triggered** checkbox.
  - Click the **New** button above the **On Completion Jobs** table of the current job and select **Add**. Select the job that shall be started on completion of the current job.

4) Click **OK** to save your changes.

After having created the job schedule, you can use it as a template for creating new job schedules. Select the job schedule in the table and click **New** > **Create Job Schedule As Copy** from the toolbar.

Creating a Job Schedule for Workflow

Alfabet supports your enterprise in defining and maintaining workflows in which you can track and coordinate the activities that should be performed by various persons in a particular sequence.

You can schedule the following batch processes via a scheduled job. Please note that none of the actions will be performed without running the batch job.

- **Sending reminder emails and setting the workflow step to Expired if a performance duration is configured for the workflow step**: A workflow step can be configured so that reminder emails can be automatically sent to the user responsible for the workflow step as well as to the workflow owner if there is no action taken on the workflow step for a configured period. This is configured by means of the **Performance Duration** attribute for the relevant workflow step. After the period configured via the **Performance Duration** attribute has elapsed, the workflow step will be set to **Expired** and reminder emails sent out. Reminder emails will only be sent if this attribute is defined, and the relevant email templates are available. (For more information, see the chapter *Configuring Workflows* in the reference manual *Configuring Alfabet with Alfabet Expand*.).

It is recommended that you execute the workflow batch job on a daily basis as workflows may begin at any time which may require workflow reminder emails to be sent.

- **Automatic start of workflows for objects defined with a query**: A workflow template can be configured to allow for the automatic initiation of a workflow for all objects found in an Alfabet query. A workflow will be automatically started for each object found by the query. To configure the automatic start of workflows for found objects, the **Automatic Start** attribute must be defined `True` for the relevant workflow template and a query must be defined for the **Base Objects via Query** attribute. Alternatively, you can restrict the execution of the batch process to workflow templates specified in the command line when starting `AlfaWorkflowCommandPromt.exe`. Please note that workflows cannot be created via batch process for workflow templates that have the **Workflow State** attribute set to `Plan` or `Retired`.

  If the **Automatic Start** attribute has been set to `True`, a workflow administrator can also execute the batch initiation of workflows via the **Start Automatically** button in the **Workflow Administration** functionality available via an administrative profile. For more information, see the section Tracking and Managing Workflows in the reference manual *User and Solution Administration*.

- **Automatic closure of workflow steps**

  A workflow step can be configured to be automatically closed and advance to the next workflow step if all post-conditions have been satisfied. Otherwise, the workflow step must be manually closed by the responsible user in order to advance to the next workflow step. To configure the automatic closure of workflow steps, the **Allow Automatic Closure** attribute must be defined `True` for the relevant workflow step.

- **Automatic deletion of finished workflows**

  A workflow template can be configured to allow for the automatic deletion of all of its workflows that are in the state **Finished**. To configure the automatic deletion of finished workflows, the **Auto Delete** attribute must be defined `True` for the relevant workflow template.

- **Rescan and update of responsible users for current workflow steps in a workflow**

  If the roles and responsibilities in your enterprise have changed, it may be necessary to update the Alfabet users who are responsible for a workflow. The batch utility can reassess the current workflow step of a selected workflow and update the assignment of responsible users to these workflow steps. Any new users found by the queries defined for the responsibility definition of the relevant workflow steps since the query was last executed will be sent the notification configured for the `OnEnterStep` workflow step action and the workflow step will be displayed in the **My Workflow Activities** view for all users found by the associated query.

- **Rescan and update workflow responsibilities**

  If the roles and responsibilities in your enterprise have changed, it may be necessary to update the Alfabet users who are responsible for a workflow. The batch utility reassesses the current workflow step of a selected workflow and updates the assignment of responsible users to these workflow steps.

  The instructions associated with the responsibility definition of the workflow step as well as the query defined for the corresponding workflow step action of type Notification will be executed to find new responsible users.

Any new users found by the query since its last execution will be sent the notification configured for the `OnEnterStep` workflow step action and the workflow step will be displayed in the **My Workflow Activities** view for all users found by the associated query.

Alternatively, the following methods are available to execute batch processes for workflows:

- The command line tool `AlfaWorkflowCommandPrompt.exe` can be used to execute batch jobs for workflows. For more information, see *Batch Processes for Workflows with AlfaWorkflowCommandPrompt.exe* in the reference manual *System Administration*.

To schedule a job for workflow related batch processes:

1) In the toolbar, click **New** > **Create Schedule**.
2) In the **Stereotype Selector** that opens, select **Workflow Job Schedule**.
3) In the editor, define the following attributes:

   Basic Data tab:

- **Name**: Define a name for the job schedule. The job schedule will be listed in the table of the **Job Schedule** functionality with this name. The name must be unique. It is not possible to define two job schedules with the same name, even if the type of job schedule is different. This attribute is mandatory.
- **Description**: Provide a meaningful description about the purpose of the job schedule.
- **Workflow Definition**: For each workflow for which you would like to execute a batch job via the same job schedule, do the following:
  1) Click **New** > **Add Workflow** and select a workflow.
  2) Select the workflow in the table and click **Action** > **Set Workflow Batch Job Type** and select one of the jobs described above.
  3) If you selected **Reassign Workflow Responsibilities** as batch job type, click **Action** > **Set Workflow Rescan Mode** and select one of the following:
     - Select `ResetDelegation` to update the responsible users of all workflows steps including the workflow steps that have been reassigned to a different user through the step delegation action. Reassignments are not taken into account.
     - Select `ReapplyStepDelegation` to update the responsible users of all workflows steps and reapply any step delegation decisions users were previously performed through the step delegation action. In this case, responsible users may be added to or removed from a delegated workflow step.
     - Select `SkipDelegatedSteps` to update the responsible users of all workflows steps except the steps that have been reassigned to a different user through the step delegation action.
- **Verbose Logging**: Select the checkbox to activate verbose logging to analyze execution issues. Only activate verbose logging temporarily for troubleshooting. Verbose logging leads to a decrease in performance and should not be used for the daily execution of jobs.
- **Indirectly Triggered**: Select the checkbox if the job has been added to the **Execution Info** > **On Completion Jobs** table of another job. This job will then be executed automatically after the other job has been finished. If you have selected this checkbox and configured another job to start this one, you do not need to define a schedule in the **Schedule** tab.

- **Priority**: If event queueing is used to execute job schedules and two jobs are scheduled to be executed at the same time, job schedules with a higher priority are executed first. This attribute should only be re-set if the execution at the scheduled time is very important and a delay is not acceptable.

Schedule tab:

- **Schedule Time**: Enter the start time for the job execution into the **Start Time** field.
- **Recurrence Pattern**: Select one of the following check boxes and provide additional data, if applicable:
  - **Hourly**: The interval between job executions is the time selected in the field **of every**.
  - **Daily**: The interval between job executions is the number of days specified in the field **day(s)** as an integer.
  - **Weekly**: The interval between job executions is the number of weeks specified in the field **week(s) on** as an integer. In addition, select the checkbox of the day of the week that the job shall be executed for each recurrence.
  - **Monthly**: There are two methods to select from. If you select the checkbox **Date-Based**, define a day of the month in the **day** field behind the checkbox. For example, if you want the job to be executed on the fifteenth of each month, enter 15. If you select the checkbox **Weekday-Based**, you can schedule the job for a specific day of a specific week in the month. In the first field behind the checkbox, select the week of the month the job shall be executed. In the second field behind the checkbox, select the day of the week the job shall be executed. The interval between job executions is the number of weeks specified in the field **month(s)** field of the selected scheduling method.
  - **Once**: The job is only executed once.
- **Range of Recurrence**: This definition is mandatory. Select the start date for the job schedule from the calendar in the **Start Date** field. For the definition of the end of the recurrence period, either select **End after:** and define the maximum number of occurrences in the **Occurrences** field or select **End by:** and select the end date for the job scheduling from the calendar in the field next to the checkbox entry. If you have selected **Once** in the **Recurrence Pattern** field, only define the date of execution via the **Start Date** field.

Execution Info tab:

This tab provides the ability to adapt the job scheduling to environmental conditions, like for example the scheduling of other jobs or maintenance windows during which execution shall be postponed.

- **Maximum Wait Time [min]**: Enter a maximum time in minutes that the job execution can be postponed if one of the following applies:
  - The execution of jobs for the current job schedule depends on the execution of one or multiple other jobs to be finished because the execution result of one job is required to be available to execute this job or because both jobs perform conflicting actions. If a job of the current schedule shall be executed while a job defined in the **Dependent Jobs** field is still running, the job will not be executed but re-scheduled for execution five minutes later until the dependent jobs are finished.
  - A job for the current job schedule cannot be executed because the current execution time lies within a maintenance window. Maintenance windows are time periods that are blocked to avoid interruption of the job by maintenance work. If a job is due within a maintenance window period, the job will not be executed immediately. If the maintenance window is

configured to re-schedule jobs, it will be re-scheduled for execution one minute after the end of the maintenance window. If the maintenance window is configured to skip jobs, the job execution during the maintenance window will be skipped and the next execution of the job is scheduled according to the settings in the **Schedule** tab of the job schedule editor.

Maintenance windows must have been configured for the Alfabet components in order to adapt job scheduling to the maintenance window via the **Maximum Wait Time [min]** setting. For information about the definition of maintenance windows, see *Defining Maintenance Windows* in the reference manual *Configuring Alfabet with Alfabet Expand* or *Defining Maintenance Windows for Scheduled Jobs* in the reference manual *System Administration*.

If the time between the originally due job execution and the end of the maintenance window or the end of execution of a dependent job exceeds the maximum wait time, job execution will be cancelled, and an error message is written into the log file. The next execution of the job is scheduled according to the settings in the **Schedule** tab of the job schedule editor.

- **Dependent Jobs**: If the execution of jobs for the current job schedule shall be postponed if another job is still running, click **New** > **Create New Job Dependency**. Select the job schedule the current job schedule depends on in the list and click **OK**. If a job is running for one of the dependent job schedules, the execution of the job for the current job schedule will be shifted to the scheduled time plus minutes until the execution of all dependent jobs is finished. If the maximum wait time defined with **Maximum Wait Time [min]** is exceeded while dependent jobs are still running, the execution of the current job is cancelled and the next execution scheduled according to the settings in the **Schedule** tab of the job schedule editor.
- **Expected Execution Time [min]**: If a maximum time for execution of the job is defined in minutes in this field, a warning is written into the log file available via the **Show Log** button if the execution of a job for the current schedule exceeds the maximum wait time. The job is nevertheless further executed.
- **Executing User**: Select a user for which the wakeup events of the job schedule shall be executed. If no user is specified, the user editing the job schedule will be the execution user for the job schedule. Each time a scheduled job has been executed, the executing user will see a slide-in **Event Feedback** message window informing him/her whether the current execution was successful. If a result is available for download, a download link is displayed. By default, the slide-in message will be displayed to the user that created the job schedule when he/she is logged in with the same user profile as on creation of the job schedule. This default behavior can be changed. Select a user from the drop-down list to show the **Event Feedback** message to that user instead. Please note that the selected user must be logged in with the user profile defined in the **Executing User Profile** attribute to receive the **Event Feedback** messages for this job schedule.
- **Executing User Profile**: The user profile that the user defined with the **Executing User** attribute must be logged in with to receive **Event Feedback** messages for execution of this job schedule. By default, a message will be displayed to the user that created the job schedule when he/she is logged in with the same user profile as on creation of the job schedule. This attribute is mandatory if **Executing User** is defined.
- **On Completion Jobs**: You can configure jobs to be started after execution of the current job. This required configuration in both jobs.
  - Configure the indirectly triggered job without an own job schedule. In the **Basic Data** tab of this job, select the **Indirectly Triggered** checkbox.
  - Click the **New** button above the **On Completion Jobs** table of the current job and select **Add**. Select the job that shall be started on completion of the current job.

4) Click **OK** to save your changes.

A batch job must be scheduled to create assignments are created for users and/or sent out email notifications to an object's authorized user if a monitor is triggered, an assignment approaches or reaches a defined due date, or an organizational or process change has occurred that affects an object. The scheduling of a batch job in regular intervals is mandatory in order to execute these functionalities. For example, if a user specifies a monitor for an object, he/she specifies the monitored context and the group of listeners that are notified by email about the monitor, and the frequency that the monitor should be executed (for example, a week before a targeted deadline). However, these specifications alone do not activate the monitor. A batch job must be executed in the specified frequency interval in order to actually execute the monitor. Without the batch job, the monitor will remain inactive. It is therefore recommended that you run the batch job at least once a day to ensure that, for example, all specified monitors are executed in the given frequency.

The following types of batch jobs are available:

- **Object Date Monitor** for the execution of date monitors specified by Alfabet users. The batch job generates email notifications triggered by the monitor.
- **Object Activity Monitor** for the execution of activity monitors specified by Alfabet users. The batch job generates email notifications triggered by the monitor.
- **Object Inactivity Monitor** for the execution of inactivity monitors specified by Alfabet users. The batch job generates email notifications triggered by the monitor.

  - For general information about the implementation of activity, inactivity, and date monitors, see the section *Keeping Track of Objects via Monitors* in the reference manual *Getting Started with Alfabet*.
  - The object classes and their monitor contexts for which the monitor functionality is available must be defined in the configuration tool Alfabet Expand. For more information, see section *Configuring Monitors* in the reference manual *Configuring Alfabet with Alfabet Expand*.
  - The text in email notifications is also determined in the configuration tool Alfabet Expand. For more information, see section *Configuring Text Templates for Email Notifications* in the reference manual *Configuring Alfabet with Alfabet Expand*.

- **Consistency Monitor** for the execution of consistency monitors. The batch job executes the Alfabet queries defined for the consistency monitors. Assignments are automatically generated for each object found by the Alfabet query. Email notifications can additionally be generated via batch process for the consistency monitor.

  - The text templates used for the assignment description are based on the text template configured in the **M_CON** text template folder in the configuration tool Alfabet Expand. The relevant text template must be selected in the **Consistency Monitor** editor in the **Consistency Monitors** functionality in the **Admin** application. For more information, see the section *Configuring Monitors* in the reference manual *User and Solution Administration*.
  - The text template used for the email notification is the text template **ConsistencyMonitorMail** located in the **MON** text template folder in the configuration tool Alfabet Expand. The relevant text template must be selected in the **Consistency Monitor** editor in the **Consistency Monitors** functionality in the **Admin** application. For more information, see *Configuring Monitors* in the reference manual *Configuring Alfabet with Alfabet Expand*

- **Notification Monitor** for the execution of notification monitors. The batch job executes the Alfabet queries defined for the notification monitors. Email notifications are automatically generated for each object found by the Alfabet query.

  - For general information about the configuration and implementation of notification monitors, see the section Defining Notification Monitors in the reference manual *User and Solution Administration*.

- **System Date Monitor** for the execution of system-wide date monitors. The batch job generates email notifications triggered by the monitor. This is specified by an administrator in the **System Date Monitors** functionality.

  - For general information about the configuration and implementation of system-wide date monitors, see the section Defining System Date Monitors in the reference manual *User and Solution Administration*.
  - The assignments generated in the context of system-wide date monitors require further configuration in the configuration tool Alfabet Expand. For more information, see the section *Configuring Assignments for System-Wide Date Monitors* in the reference manual *Configuring Alfabet with Alfabet Expand*.
  - The text in email notifications is also determined in the configuration tool Alfabet Expand. For more information, see *Configuring Text Templates for Email Notifications* in the reference manual *Configuring Alfabet with Alfabet Expand*.

- **Organizational Change** for the dispatch of email notifications about the assignments generated due to organizational changes. An XML object for the definition of organizational changes must be configured in Alfabet Expand to execute this type of batch job. See *Configuring the Propagation of Organizational Changes* in the reference manual *Configuring Alfabet with Alfabet Expand*.
- **Business Process Change** for the dispatch of email notifications about the assignments generated due to business process changes. An XML object for the definition of organizational changes must be configured in Alfabet Expand to execute this type of batch job. See *Configuring the Propagation of Business Process Changes* in the reference manual *Configuring Alfabet with Alfabet Expand*.
- **Expired Report Deletion Job** for the deletion of all expired APF publications from the **Internal Document Selector**. Publications are generated on basis of a publication definition based on a Microsoft® Word template. Publication can either be triggered by a batch job or by a Alfabet user opening a configured report that is defined to trigger publications. For more information about the configuration of APF publications, see *Publishing Data in Microsoft Word or PowerPoint Format* in the reference manual *Configuring Alfabet with Alfabet Expand*.

  An expiration date of one month after creation date is configured for publications. Once the expiration date is reached, the publication will no longer be displayed to authorized users in the Alfabet interface. However, it will remain in the **Internal Document Selector** until deleted via the batch process.

Alternatively, the following methods are available to execute the jobs:

- The command line tool `AlfaBatchExecutor.exe` can be used to execute the above listed jobs. For more information, see *Batch Processing for Monitors and Change Management with AlfaBatchExecutor.exe* in the reference manual *System Administration*.

To schedule one of the above listed jobs:

1) In the toolbar, click **New** > **Create Schedule**.

2) In the **Stereotype Selector** that opens, select **Batch Executor Job Schedule**.

3) In the editor, define the following attributes:

Basic Data tab:

- **Name**: Define a name for the job schedule. The job schedule will be listed in the table of the **Job Schedule** functionality with this name. The name must be unique. It is not possible to define two job schedules with the same name, even if the type of job schedule is different. This attribute is mandatory.
- **Description**: Provide a meaningful description about the purpose of the job schedule.
- **Job Class**: Select which of the above-listed jobs shall be executed.
- **Parameters**: If monitors shall be executed, you can limit the execution to a subset of existing monitors. Provide the names of the monitors that shall be executed in a semicolon separated list.
- **Verbose Logging**: Select the checkbox to activate verbose logging to analyze execution issues. Only activate verbose logging temporarily for troubleshooting. Verbose logging leads to a decrease in performance and should not be used for the daily execution of jobs.
- **Indirectly Triggered**: Select the checkbox if the job has been added to the **Execution Info** > **On Completion Jobs** table of another job. This job will then be executed automatically after the other job has been finished. If you have selected this checkbox and configured another job to start this one, you do not need to define a schedule in the **Schedule** tab.
- **Priority**: If event queueing is used to execute job schedules and two jobs are scheduled to be executed at the same time, job schedules with a higher priority are executed first. This attribute should only be re-set if the execution at the scheduled time is very important, and a delay is not acceptable.

Schedule tab:

- **Schedule Time**: Enter the start time for the job execution into the **Start Time** field.
- **Recurrence Pattern**: Select one of the following check boxes and provide additional data, if applicable:
  - **Hourly**: The interval between job executions is the time selected in the field **of every**.
  - **Daily**: The interval between job executions is the number of days specified in the field **day(s)** as an integer.
  - **Weekly**: The interval between job executions is the number of weeks specified in the field **week(s) on** as an integer. In addition, select the checkbox of the day of the week that the job shall be executed for each recurrence.
  - **Monthly**: There are two methods to select from. If you select the checkbox **Date-Based**, define a day of the month in the **day** field behind the checkbox. For example, if you want the job to be executed on the fifteenth of each month, enter 15. If you select the checkbox **Weekday-Based**, you can schedule the job for a specific day of a specific week in the month. In the first field behind the checkbox, select the week of the month the job shall be executed. In the second field behind the checkbox, select the day of the week the job shall be executed. The interval between job executions is the number of weeks specified in the field **month(s)** field of the selected scheduling method.
  - **Once**: The job is only executed once.

- **Range of Recurrence**: This definition is mandatory. Select the start date for the job schedule from the calendar in the **Start Date** field. For the definition of the end of the recurrence period, either select **End after:** and define the maximum number of occurrences in the **Occurrences** field or select **End by:** and select the end date for the job scheduling from the calendar in the field next to the checkbox entry. If you have selected **Once** in the **Recurrence Pattern** field, only define the date of execution via the **Start Date** field.

Execution Info tab:

This tab provides the ability to adapt the job scheduling to environmental conditions, like for example the scheduling of other jobs or maintenance windows during which execution shall be postponed.

- **Maximum Wait Time [min]**: Enter a maximum time in minutes that the job execution can be postponed if one of the following applies:
  - The execution of jobs for the current job schedule depends on the execution of one or multiple other jobs to be finished because the execution result of one job is required to be available to execute this job or because both jobs perform conflicting actions. If a job of the current schedule shall be executed while a job defined in the **Dependent Jobs** field is still running, the job will not be executed but re-scheduled for execution five minutes later until the dependent jobs are finished.
  - A job for the current job schedule cannot be executed because the current execution time lies within a maintenance window. Maintenance windows are time periods that are blocked to avoid interruption of the job by maintenance work. If a job is due within a maintenance window period, the job will not be executed immediately. If the maintenance window is configured to re-schedule jobs, it will be re-scheduled for execution one minute after the end of the maintenance window. If the maintenance window is configured to skip jobs, the job execution during the maintenance window will be skipped and the next execution of the job is scheduled according to the settings in the **Schedule** tab of the job schedule editor.

    Maintenance windows must have been configured for the Alfabet components in order to adapt job scheduling to the maintenance window via the **Maximum Wait Time [min]** setting. For information about the definition of maintenance windows, see *Defining Maintenance Windows* in the reference manual *Configuring Alfabet with Alfabet Expand* or *Defining Maintenance Windows for Scheduled Jobs* in the reference manual *System Administration*.

  If the time between the originally due job execution and the end of the maintenance window or the end of execution of a dependent job exceeds the maximum wait time, job execution will be cancelled, and an error message is written into the log file. The next execution of the job is scheduled according to the settings in the **Schedule** tab of the job schedule editor.

- **Dependent Jobs**: If the execution of jobs for the current job schedule shall be postponed if another job is still running, click **New** > **Create New Job Dependency**. Select the job schedule the current job schedule depends on in the list and click **OK**. If a job is running for one of the dependent job schedules, the execution of the job for the current job schedule will be shifted to the scheduled time plus minutes until the execution of all dependent jobs is finished. If the maximum wait time defined with **Maximum Wait Time [min]** is exceeded while dependent jobs are still running, the execution of the current job is cancelled and the next execution scheduled according to the settings in the **Schedule** tab of the job schedule editor.

- **Expected Execution Time [min]**: If a maximum time for execution of the job is defined in minutes in this field, a warning is written into the log file available via the **Show Log** button if the execution of a job for the current schedule exceeds the maximum wait time. The job is nevertheless further executed.

- **Executing User**: Select a user for which the wakeup events of the job schedule shall be executed. If no user is specified, the user editing the job schedule will be the execution user for the job schedule. Each time a scheduled job has been executed, the executing user will see a slide-in **Event Feedback** message window informing him/her whether the current execution was successful. If a result is available for download, a download link is displayed. By default, the slide-in message will be displayed to the user that created the job schedule when he/she is logged in with the same user profile as on creation of the job schedule. This default behavior can be changed. Select a user from the drop-down list to show the **Event Feedback** message to that user instead. Please note that the selected user must be logged in with the user profile defined in the **Executing User Profile** attribute to receive the **Event Feedback** messages for this job schedule.
- **Executing User Profile**: The user profile that the user defined with the **Executing User** attribute must be logged in with to receive **Event Feedback** messages for execution of this job schedule. By default, a message will be displayed to the user that created the job schedule when he/she is logged in with the same user profile as on creation of the job schedule. This attribute is mandatory if **Executing User** is defined.
- **On Completion Jobs**: You can configure jobs to be started after execution of the current job. This required configuration in both jobs.
  - Configure the indirectly triggered job without an own job schedule. In the **Basic Data** tab of this job, select the **Indirectly Triggered** checkbox.
  - Click the **New** button above the **On Completion Jobs** table of the current job and select **Add**. Select the job that shall be started on completion of the current job.

4) Click **OK** to save your changes.

After having created the job schedule, you can use it as a template for creating new job schedules. Select the job schedule in the table and click **New** > **Create Job Schedule As Copy** from the toolbar.

Creating a Job Schedule for Batch Deletion of Old ADIF Session Information

ADIF session information older than a defined number of days can be batch deleted for a defined ADIF scheme via a job schedule. The information about job execution can either be deleted completely or deletion can be limited to the log file content while the information that the job was executed, and the resulting execution state is retained.

To schedule a job for deletion of ADIF session information:

1) In the toolbar, click **New** > **Create Schedule**.
2) In the **Stereotype Selector** that opens, select **Clear ADIF Session Content Job Schedule**.
3) In the editor, define the following attributes:

   Basic Data tab:

- **Name**: Define a name for the job schedule. The job schedule will be listed in the table of the **Job Schedule** functionality with this name. The name must be unique. It is not possible to define two job schedules with the same name, even if the type of job schedule is different. This attribute is mandatory.
- **Description**: Provide a meaningful description about the purpose of the job schedule.

- **ADIF Scheme**: Select the ADIF scheme for which job execution information shall be deleted from the Alfabet database. If no ADIF scheme is selected, the job execution information will be deleted for all ADIF schemes.
- **Number of Days**: The number of days for that job execution information shall be retained. The number of days must be a positive integer. If you enter 1, information about job execution for the current date and the day before the current date is retained. If you enter 0, all information including the information of job execution on the current date is removed.
- **Level**: Select **Log** to remove only the log file content stored in the database while the information about the job execution date and the execution state is retained. Select **All** to remove all information about the job execution.

Schedule tab:

- **Schedule Time**: Enter the start time for the job execution into the **Start Time** field.
- **Recurrence Pattern**: Select one of the following check boxes and provide additional data, if applicable:
  - **Hourly**: The interval between job executions is the time selected in the field **of every**.
  - **Daily**: The interval between job executions is the number of days specified in the field **day(s)** as an integer.
  - **Weekly**: The interval between job executions is the number of weeks specified in the field **week(s) on** as an integer. In addition, select the checkbox of the day of the week that the job shall be executed for each recurrence.
  - **Monthly**: There are two methods to select from. If you select the checkbox **Date-Based**, define a day of the month in the **day** field behind the checkbox. For example, if you want the job to be executed on the fifteenth of each month, enter 15. If you select the checkbox **Weekday-Based**, you can schedule the job for a specific day of a specific week in the month. In the first field behind the checkbox, select the week of the month the job shall be executed. In the second field behind the checkbox, select the day of the week the job shall be executed. The interval between job executions is the number of weeks specified in the field **month(s)** field of the selected scheduling method.
  - **Once**: The job is only executed once.
- **Range of Recurrence**: This definition is mandatory. Select the start date for the job schedule from the calendar in the **Start Date** field. For the definition of the end of the recurrence period, either select **End after:** and define the maximum number of occurrences in the **Occurrences** field or select **End by:** and select the end date for the job scheduling from the calendar in the field next to the checkbox entry. If you have selected **Once** in the **Recurrence Pattern** field, only define the date of execution via the **Start Date** field.

Execution Info tab:

This tab provides the ability to adapt the job scheduling to environmental conditions, like for example the scheduling of other jobs or maintenance windows during which execution shall be postponed.

- **Maximum Wait Time [min]**: Enter a maximum time in minutes that the job execution can be postponed if one of the following applies:
  - The execution of jobs for the current job schedule depends on the execution of one or multiple other jobs to be finished because the execution result of one job is required to be available to execute this job or because both jobs perform conflicting actions. If a job of the current

schedule shall be executed while a job defined in the **Dependent Jobs** field is still running, the job will not be executed but re-scheduled for execution five minutes later until the dependent jobs are finished.

- A job for the current job schedule cannot be executed because the current execution time lies within a maintenance window. Maintenance windows are time periods that are blocked to avoid interruption of the job by maintenance work. If a job is due within a maintenance window period, the job will not be executed immediately. If the maintenance window is configured to re-schedule jobs, it will be re-scheduled for execution one minute after the end of the maintenance window. If the maintenance window is configured to skip jobs, the job execution during the maintenance window will be skipped and the next execution of the job is scheduled according to the settings in the **Schedule** tab of the job schedule editor.

  Maintenance windows must have been configured for the Alfabet components in order to adapt job scheduling to the maintenance window via the **Maximum Wait Time [min]** setting. For information about the definition of maintenance windows, see *Defining Maintenance Windows* in the reference manual *Configuring Alfabet with Alfabet Expand* or *Defining Maintenance Windows for Scheduled Jobs* in the reference manual *System Administration*.

If the time between the originally due job execution and the end of the maintenance window or the end of execution of a dependent job exceeds the maximum wait time, job execution will be cancelled and an error message is written into the log file. The next execution of the job is scheduled according to the settings in the **Schedule** tab of the job schedule editor.

- **Dependent Jobs**: If the execution of jobs for the current job schedule shall be postponed if another job is still running, click **New** > **Create New Job Dependency**. Select the job schedule the current job schedule depends on in the list and click **OK**. If a job is running for one of the dependent job schedules, the execution of the job for the current job schedule will be shifted to the scheduled time plus minutes until the execution of all dependent jobs is finished. If the maximum wait time defined with **Maximum Wait Time [min]** is exceeded while dependent jobs are still running, the execution of the current job is cancelled and the next execution scheduled according to the settings in the **Schedule** tab of the job schedule editor.

- **Expected Execution Time [min]**: If a maximum time for execution of the job is defined in minutes in this field, a warning is written into the log file available via the **Show Log** button if the execution of a job for the current schedule exceeds the maximum wait time. The job is nevertheless further executed.

- **Executing User**: Select a user for which the wakeup events of the job schedule shall be executed. If no user is specified, the user editing the job schedule will be the execution user for the job schedule. Each time a scheduled job has been executed, the executing user will see a slide-in **Event Feedback** message window informing him/her whether the current execution was successful. If a result is available for download, a download link is displayed. By default, the slide-in message will be displayed to the user that created the job schedule when he/she is logged in with the same user profile as on creation of the job schedule. This default behavior can be changed. Select a user from the drop-down list to show the **Event Feedback** message to that user instead. Please note that the selected user must be logged in with the user profile defined in the **Executing User Profile** attribute to receive the **Event Feedback** messages for this job schedule.

- **Executing User Profile**: The user profile that the user defined with the **Executing User** attribute must be logged in with to receive **Event Feedback** messages for execution of this job schedule. By default, a message will be displayed to the user that created the job schedule when he/she is logged in with the same user profile as on creation of the job schedule. This attribute is mandatory if **Executing User** is defined.

- **On Completion Jobs**: You can configure jobs to be started after execution of the current job. This required configuration in both jobs.
  - Configure the indirectly triggered job without an own job schedule. In the **Basic Data** tab of this job, select the **Indirectly Triggered** checkbox.
  - Click the **New** button above the **On Completion Jobs** table of the current job and select **Add**. Select the job that shall be started on completion of the current job.

4) Click **OK** to save your changes.

After having created the job schedule, you can use it as a template for creating new job schedules. Select the job schedule in the table and click **New** > **Create Job Schedule As Copy** from the toolbar.

Checking Success of a Scheduled Job Execution

A Job Schedule Report functionality is available that give an overview as well as details

the Job Schedule Report view shows four sections:

- A filter area
- A tabular overview of all scheduled jobs matching the current filter criteria
- A chart area on the right of the table that displays a bar chart of all executions of the job schedule currently selected in the tabular overview.
- A field beneath the chart for display of the log file for a selected result from the bar chart.

You can do the following to check the success of job schedule execution:

- Limiting the Information via Filter Settings

Limiting the Information via Filter Settings

To limit the results displayed in the **Job Schedule Report** to a subset of available job schedule executions, set the following filters and click the **Update** button:

- **Stereotype**: Select one or multiple job schedule stereotypes to limit the results to executions of job schedules with the selected stereotypes.
- **Name**: Enter a string to view only job schedule stereotypes with a name including the selected string.
- **User**: Select a user to limit the results to job schedules owned by the selected user. A job schedule is owned by the person creating the job schedule, if not otherwise defined in the job schedule with the **Executing User** attribute in the **Execution Info** tab of the job schedule editor.
- **Time Filter**: You can select to view results for the last 24 hours, the last week or the last month.

Chapter 27: Managing Data Quality

Data quality includes more than a check for availability of a setting for an attribute. An attribute value might need to have a defined structure for different domains in the company, like for example a prefix for the name of projects to map them to different departments. Other attributes only need to be set if other attributes are already set, for example the attribute storing information about the cloud service provider must only be set if the application is marked as cloud relevant via another attribute.

Data quality rules provide the ability to define rules which shall apply for data entry individually according to the demands of the customer. Each data quality rule also specifies the severity level for rule violation and information about the required fix for the rule violation.

Users can be informed directly on the user interface about the overall correctness of the data for an object and the severity of data rule violation. A color code with yellow for hints, orange for warnings, and red for errors highlights the information. For each data quality violation, the user can be informed about the action required to enhance data quality and can be guided to the relevant editor or view to correct the data.

Users can optionally also be informed via email about open data quality rule violations that they need to fix.

Implementation of data quality rules requires the following:

- Solution designers must define the customization of object cockpits and the availability of configured reports in the tool Alfabet Expand to provide the information about the data quality to the end users. The reference manual *Configuring Alfabet with Alfabet Expand* describes different ways to add information about the data quality to users on the user interface. In data workbenches, default data quality score and maximum data quality violation severity are displayed to the users without prior configuration required.
- Data quality rules must be defined in the **Data Quality Rule Configuration** functionality.
- A private ADIF scheme must be executed in regular intervals to calculate data completeness and correctness. The ADIF scheme can be executed to calculate the data quality either for all objects, a single object, or all objects of a defined object class. Regular execution of the ADIF scheme can be scheduled via the **Job Scheduler** functionality.

  In addition, the calculation of the rules can be triggered on close of an editor or wizard step to recalculate the score directly after data entry. This configuration is done by solution designers in the tool Alfabet Expand.

- A private ADIF scheme must be executed in regular intervals to send the emails out for data quality notifications. Regular execution of the ADIF scheme can be scheduled via the **Job Scheduler** functionality.

The following sections inform about the definition of data quality rules and the execution of the rules via the **Job Scheduler** functionality:

Defining Data Quality Rules

The **Data Quality Rule Configuration** functionality allows data quality rule groups and data qualities to be defined.

Data quality rule groups structure data quality rules for better maintenance. In addition, the grouping of data quality rules can be used to limit display of data quality rule violations in a configured report to the rules of a defined group. For example, data quality rules defining the data quality of security related attribute settings for applications can be stored in a separate group. A configured report which only informs about data violations of data quality rules in the security group can then be defined for security officers.

Data quality rules groups can be hierarchically structured.

Data quality rules can only be created in a data quality rule group. A data quality rule can be assigned to multiple data quality rule groups.

Definition of data quality rule groups and data quality rules are described in the following sections:

- Creating Data Quality Rule Group
- Changing the Structure of the Data Quality Rule Group Hierarchy
- Creating Data Quality Rules
- Managing Data Quality Rules in the Data Quality Group Hierarchy

Creating Data Quality Rule Group

To create a data quality rule group:

- To open a view to create a data quality rule, either click on the root node of the explorer or click on an existing data quality rule group and open the **Subordinate Data Quality Rule Groups** page view of the data quality rule group.
- In the toolbar, select **New** > **Create New Data Quality Rule**.
- In the editor that opens, define a name and description for the data quality rule.

Changing the Structure of the Data Quality Rule Group Hierarchy

Data quality rule groups can be hierarchically structured. You can change the existing hierarchy of data quality rules with the following actions:

- **To move a data quality rule group that exists on any level of the hierarchy as child to another data quality rule group**:
  1) Open the **Subordinate Data Quality Rule Groups** page view of the intended superordinate data quality rule group.
  2) In the toolbar, select **New** > **Move Existing Data Quality Rule Group**.
  3) Select the data quality rule group that shall be moved from the selector.
- **To move a data quality rule group that is subordinate of another data quality rule group to the root level**:

1) Open the **Subordinate Data Quality Rule Groups** page view of the superordinate data quality rule group.
2) Select the data quality rule group in the table.
3) In the toolbar, select the **Detach** button.

- **To delete a data quality rule group**:
  1) Click the root node of the explorer to delete a root data quality rule group or open the **Subordinate Data Quality Rule Groups** page view of the superordinate data quality rule group.
  2) Select the data quality rule group in the table.
  3) In the toolbar, click the **Delete** button.
  4) A warning will be displayed if the delete action will also delete data quality rule groups and data quality rules exclusively subordinate to the data quality rule group that shall be deleted. Data quality rules which are not only assigned to the data quality rule group which shall be deleted but also to other data quality rules groups will not be deleted. They will still exist and be children of the other data quality rule groups.

     If you confirm the warning, the data quality rule group and all subordinate content exclusively assigned to the data quality rule group will be irrevocably deleted from the Alfabet database.

Creating Data Quality Rules

Data quality rules are created in the **Data Quality Rules Configuration** functionality in the **Data Quality Rules** page view of a data quality rule group.

To create a data quality rule:

1) In the toolbar, select **New** > **Create a Data Quality Rule**
2) In the editor that opens, define the following attributes for the data quality rule:

   Mandatory Properties:

   - **Name**: Enter a meaningful name for the data quality rule. The name of the data quality rule shall be formulated to give a short hint to the user about the required data input.
   - **Severity**: Select a severity from the drop-down list. The severity of the data quality rule is coupled with a color code used to display the severity of a data quality rule violation on the user interface. Hints are displayed in blue, warnings in yellow and errors in red.
   - **Data Quality Rule Type**: Select one of the following data quality rule types from the drop-down list:
     - **Value Exists**: Checks the availability of any input for a defined object class property of a defined object class/object class stereotype.
     - **Regular Expression**: Checks the composition of a string for a defined object class property of a defined object class/object class stereotype via a regular expression. For example, a rule can check whether a string starts with a specific letter or contains no special characters. This type of data quality rule enables naming conventions to be checked for objects.
     - **SQL Query**: Checks the input of an object class property based on the definition of other object class properties. For example, a rule can check whether business support is defined for

an application. Or a rule could check only that a custom attribute is set to define the cloud service provider for applications specified as cloud relevant.

- **Target Classes**: Select the object classes for which the data quality rule shall be evaluated.
- **Is Active**: Select the checkbox after complete configuration of the data quality rule. Only data quality rules for which **Is Active** is set will be evaluated.

Attributes mandatory for the **Value Exists** data quality rule type:

- **Target Properties**: Select the object class property that shall be mandatory.

Attributes relevant for the **SQL Query** data quality rule type:

- **Expression**: Define an SQL query that returns a result if the rule is fulfilled. The query must refer to the current object via the Alfabet query language parameter `@BASE`.

For example, to ensure that the application is assigned to at least one business support, define the following query:

```
SELECT app.REFSTR
FROM APPLICATION app
WHERE app.REFSTR = @BASE
AND NOT EXISTS(SELECT * FROM BUSINESSSUPPORT bss WHERE
bss.OBJECT = app.REFSTR)
```

- **Complex SQL Query**: Select the checkbox if the query is complex, for example containing a `WITH` statement. By default, the evaluation mechanism can only read simple SQL queries. For performance reasons, the checkbox should only be selected if applicable. You can use the test functionality in the **Test Data Quality Rule Definition** tab to see whether the SQL query is successfully evaluated. If the SQL query is not passing the test, select the checkbox and test again.
- **Target Properties**: This setting is optional and only required to enable the navigation from a configured report about data quality issues to a target view to fix the issue. If the problem defined in the query can be fixed via setting one or multiple object class property values in an editor, select the properties from the drop-down list.

Attributes mandatory for the **Regular Expression** data quality rule type:

- **Target Properties**: Select one or multiple object class properties of the type `String` for which the rule shall apply.
- **Expression**: Define a regular expression that expresses how a string entered into the selected object class property or object class properties shall be defined.

Standard regular expressions for defining a sequence of strings are accepted. If you are not familiar with regular expressions for defining a sequence of strings, you can consult descriptions of regular expression syntax in the internet. An overview is for example provided at: https://learn.microsoft.com/en-us/dotnet/standard/base-types/regular-expression-language-quick-reference

For example, the following regular expression enforces that the data input is either a number or the string "n.a.":

```
^n\.a\.$|^[0-9]
```


Attributes to inform the user how to solve the issue:

- **Resolution**: Enter a short explanation about the action required to solve the data quality issue.
- Configured reports about data quality issues found during evaluation of the data quality rule can be configured to provide navigation to the resolution target. By default, the editor opens for setting the object class properties defined in the **Target Properties**. Alternatively, a standard page view or configured report can open. This requires the setting of the following fields:
  - **Resolution View Type**: Select **Standard View** to provide navigation to a standard Alfabet view. Select **Configured Report** to provide navigation to a configured report.
  - **Resolution View Name**: Select the view that shall open to resolve the data quality issue from the drop-down list.
- **Resolution View Type**:

Optional attributes that limit the validity of the rule:

- **Valid From**: If a date is selected in this field, evaluation of the data quality rule will start at the selected date.
- **Valid To**: If a date is selected in this field, evaluation of the data quality rule will stop at the selected date.

Attributes for administration of the rule:

- **Description**: The description field is for the data quality rule administrator only to describe the relevance of the data quality rule. It shall not be included into configured reports or object views to inform the user about the data quality rule.

3) Change to the **Test Data Quality Rule Definition** tab of the editor.
4) Select the number of objects per applicable object class for that a test evaluation of the data quality rule shall be performed.
5) Click the **Test** button. The rule is evaluated, and a message is displayed that informs about success or issues during execution and the duration of execution. Below the message, the objects violating the rule that are found during the test run are displayed in the editor tab.
6) Optionally, configure the sending of notifications via email for unresolved data quality issues. Emails will be sent out to users in a grouped way. On each run of the ADIF job executing email sending, a user will receive one email only containing all data quality violation issues for all data quality rules configured to send notification emails.
   - **Send Email Notifications**: Select the checkbox to activate sending of emails.
   - **Notify Responsible User**: Select the checkbox to send notification emails to the authorized user of the object for which a data quality violation has been detected.
   - **Notify Roles**: Select the checkbox of one or multiple roles listed in the field to send notification emails to the users with the selected role for the object for which a data quality violation has been detected.

Managing Data Quality Rules in the Data Quality Group Hierarchy

The **Data Quality Rules** page view of a data quality rule group provides a number of functions to manage data quality rules:

- **To move a data quality rule that already exists in another data quality rule group to the current data quality rule group**:

  1) In the toolbar, select **New** > **Move Existing Data Quality Rule**.
  2) Select the data quality rule that shall be moved from the selector.

  The data quality rule is removed from all other data quality rule groups it was assigned to and is assigned to the current data quality rule group.

- **To add a data quality rule that already exists in another data quality rule group to the current data quality rule group**:

  1) In the toolbar, select **New** > **Add Existing Data Quality Rule**.
  2) Select the data quality rule that shall be added from the selector.

  The data quality rule is added to the current data quality rule group. It also remains in all other data quality rule groups it was assigned to.

- **To detach a data quality rule from a data quality rule group**:

  - Select the data quality rule in the table.
  - In the toolbar, select the **Detach**  button.

  If a data quality rule is not assigned to any data quality rule group, it is moved to the explorer folder **Unassigned Data Quality Rules**. Expand the folder node in the explorer to view the folder content. If unassigned data quality rules exist, they should be assigned to a data quality rule group. Navigate to the **Data Quality Rules** page view of a data quality rule group and re-assign the data quality rule using the **New** > **Move Existing Data Quality Rule** toolbar option.

- **To delete a data quality rule**:

  1) Select the data quality rule in the table.
  2) In the toolbar, click the **Delete** button.

  If you confirm the warning, the data quality rule will be irrevocably deleted from the Alfabet database.

Calculating Data Quality Rules and Sending out Notification Emails

Data quality rules are calculated via a private ADIF scheme. The ADIF scheme can calculate the data quality for all objects, a single object, or objects of a defined object class. The calculation results in the following:

- For each data quality rule plus object whose data violates the rule, an object of the new **Data Quality Violation** object class is created. The data quality violation stores the information about the issue. Configured reports can be created for this object class to provide information about data quality violations, the proposed fixes, and navigation to the views or editors relevant for fixing the issues. For more information about the configuration of data quality evaluation reports, see *Configuring the Data Quality Rules Functionality* in the reference manual *Configuring Alfabet with Alfabet Expand*.

- A data quality score is calculated for each object. The results of the data quality score calculations are stored in the new object class properties **Data Quality Score** and **Data Quality Severity**. These object class properties are available for all object classes that capture information about the IT and business infrastructure.

To start the evaluation of data quality from the Alfabet user interface you must run an ADIF job based on the ADIF scheme **RunRescanDataQualityForObject**.

To trigger sending of email notifications, you must run an ADIF job based on the ADIF scheme **Notify-DataQualityRuleViolations**. The ADIF job can be run with different settings for the minimum severity a data quality violation must have to be included in the email. Only one email containing all data quality rule violations relevant for a user is sent to a user during a single run of the ADIF job.

Run the ADIF jobs with one of the following methods:

- To start the ADIF job once, for example for test purposes, the ADIF job can be started via the **ADIF Jobs Administration** (`ADMIN_AdifJobs`) functionality. For more information, see Executing and Controlling ADIF Jobs.
- To batch calculate the data quality score regularly, the ADIF job can be started via the **Job Scheduler** (`JobSchedule`) functionality on the Alfabet user interface. For more information about the **Job Scheduler** functionality, see *Activating the Job Schedule Functionality* and the section Scheduling ADIF Jobs and Batch Jobs via the Job Schedule Functionality.

Solution designers with access to Alfabet Expand can define automatic execution of the ADIF job after the close of a wizard or workflow step. For more information, see *Configuring the Data Quality Rules Functionality* in the reference manual *Configuring Alfabet with Alfabet Expand*.

Chapter 28: Reviewing and Correcting the Original Vocabularies

The **Review Vocabulary Strings** and **Implement Changed Vocabulary Strings** functionalities allow you to review the custom strings created in the context of the enterprise's solution configuration. This includes, for example, strings created for configuration objects in the context of Alfabet Expand or for navigation pages in the context of the Guide Pages Designer. The new functionalities **Review Vocabularies** (`ADMIN_Vocabulary_Reviewer`) and **Implement Changed Vocabulary Strings** (`ADMIN_VocabularyImplementer`) are accessible in the Alfabet user interface.

The following information is available:

Reviewing the Vocabulary Strings

The **Review Vocabulary Strings** functionality allows a language reviewer to review all public strings in the database and propose changes to the strings as needed. Information regarding the configuration object or associated code where the string occurs is included in order to provide the language reviewer with information about the string's context. Strings can be filtered based on search words, the vocabulary source (`METAMODEL`, `GUIDEPAGES`, `ITPlan`, etc.), and creation date. A string that requires no changes can be marked as reviewed and saved with the review date and name of the reviewer. Reviewed strings as well as strings that have proposed changes will be automatically removed from the **Review Vocabularies** dataset. However, reviewed strings can be included in the dataset by selecting the **Include Reviewed Strings** checkbox by means of search criteria. The strings with proposed changes will be moved to the **Implement Changed Vocabulary Strings** functionality, where it can be reviewed by the solution designer or other relevant user.

The following filters are available:

- **Source**: Select one or more vocabulary sources to display strings stored in the specified source. Strings from your solution configuration will typically be in the (`METAMODEL` or `GUIDEPAGES` vocabulary source although you may also modify protected strings in the (`ITPlan`, `Platform`, and `Extension` vocabulary sources.
- **Original**: Enter the string to display the string in the dataset. Use the wildcard * before and/or after the string if only a part of the string is entered.
- **Created Before**: Select a date to display all strings reviewed on and after the specified date.
- **Created After**: Select a date to display all strings created on and before the specified date.

- **Accessibility**: Select Protected to display standard strings or select Public to select custom strings.
- **Configuration Culture**: Select the language used to configure the original strings.
- **Include Obsolete Strings**: Select the checkbox to display strings that have become obsolete due to a proposed change being implemented.
- **Include Reviewed Strings**: Select the checkbox to display strings that have been marked as reviewed.

The following columns are displayed:

- **Source**: Displays the source of the vocabulary.
- **Original String**: Displays the string to be reviewed.
- **Accessibility**: Displays whether the string is a protected string created by Alfabet or a public string created in the context of your solution.
- **Length**: Displays the number of characters in the string. Strings may not be longer than 600 characters. If a text is longer than 600 characters, it will NOT be stored in the Alfabet database.

  If the string shall be translated, you should consider what the maximum length of the original language string should be in order to compensate for the translation. For example, a string in German is typically 6% longer than the same string in English.

- **Trailing Blank**: Displays a checkmark if a blank character exists at the end of the string.
- **Proposed String**: Displays the string that has been proposed to replace the original string.
- **Change Requested**: Displays a checkmark if the string has been reviewed and a change proposal has been made.
- **Occurrences**: Displays the configuration object where the string is used. If more than one occurrence exists, the number of occurrences will be displayed. Click the **Show Occurrences** button to view a dialog showing all occurrences of the string.
- **Create Date**: Displays the date that the original string was created.
- **Configuration Culture**: Displays the culture that the string was originally created for.

The following columns will be appended to the dataset if the **Include Reviewed Strings** checkbox is selected:

1) **Reviewer**: Displays the user name of the user who reviewed the string.
2) **Review Date**: Displays the date that the original string was reviewed.
3) **Review Comment**: Displays the comment made by the reviewer about the proposed change.
4) **Implementer**: Displays the user name of the user who implemented the string.
5) **Implemented String**: Displays the changed string that replaced the original string.
6) **Implementation Comment**: Displays the comment made by the implementer about the change that was made to the original string.
7) **Replaces**: Displays the string that was replaced by the implemented string.

The following columns will be appended to the dataset if the **Include Obsolete Strings** checkbox is selected:

1) **Obsolete**: Displays a checkmark if the original string is obsolete and no longer implemented for the configuration object.

2) **Obsolete Date**: Displays the date that the original string was made obsolete because the implementer changed the original string.

The following information is available:

- Understanding Where the String Occurs in the Solution Configuration
- Proposing a Change for a Single String
- Proposing Changes for Multiple Strings Simultaneously
- Marking Strings as Reviewed

Understanding Where the String Occurs in the Solution Configuration

In order to understand where a string is used in the solution configuration, select the relevant row and click **Show Occurrences of String**. A dialog will open. The **Context Column** will display the type of configuration object followed by the name of the configuration object. The **Technical Information** columns displays additional information about the type of interface control where the string can be found. Click **OK** to close the dialog.

Proposing a Change for a Single String

You can propose a change for any string displayed in the dataset. The string must be changed in the **Proposed Change** field in the **Review String** editor in order to mark the string as reviewed and move it to the **Implement Changed Vocabulary Strings** functionality.

To propose a change to a selected string:

1) Select the row with the relevant string that you want to change and click the **Review String** button in the toolbar. The **Review String** editor opens.
2) Define the following fields, as needed.
   - **Original**: Displays the original string.
   - **Source**: Displays the source of the vocabulary.
   - **Obsolete Date**: If relevant, displays the date that the string was made obsolete in the vocabularies.
   - **Obsolete**: Displays a checkmark if the string is obsolete.
   - **Proposed Change**: The original string is displayed per default in this field. Modify the text as needed. The string must be modified to be marked as reviewed and displayed in the **Implement Changed Vocabulary Strings** functionality.
   - **Review Comment**: Optional. Provide a comment regarding the proposed change. Please note that the string will not be marked as reviewed and displayed in the **Implement Changed Vocabulary Strings** functionality if only a comment is provided but the string is not changed in the **Proposed Change** field.
3) Click **OK** to save your changes.

Proposing Changes for Multiple Strings Simultaneously

You can simultaneously review multiple strings at once. All selected strings will be displayed in the **Multi-Editor to Review Vocabulary Strings** editor. A string must be changed in the **Proposed Change** field in the **Review String** editor to mark the string as reviewed and move it to the **Implement Changed Vocabulary Strings** functionality.

To propose changes for multiple strings:

1) Select a row and hold the CTRL key as you select the rows with the relevant strings that you want to change and the **Review Selected Strings** button in the toolbar. The **Multi-Editor to Review Vocabulary Strings** editor opens.
2) Define the following columns, as needed.
   - **Original String**: Displays the original string.
   - **Source**: Displays the source vocabulary.
   - **Trailing Blank**: Displays a checkmark if a blank character exists at the end of the string.
   - **Proposed Change**: The original string is displayed per default in this field. Modify the text as needed by changing the text in the cell or clicking the **Edit** button to expand the field in which you can edit the string. The string must be modified in order to be marked as reviewed and displayed in the **Implement Changed Vocabulary Strings** functionality.
   - **Review Comment**: Optional. Provide a comment regarding the proposed change. Please note that the string will not be marked as reviewed and displayed in the **Implement Changed Vocabulary Strings** functionality if only a comment is provided but the string is not changed in the **Proposed Change** field.
3) Click **OK** to save your changes:

Marking Strings as Reviewed

If a string requires no changes, you can mark it as reviewed. In this way, it will be removed from the dataset. It will not be moved to the **Implement Changed Vocabulary Strings** functionality. The string can be viewed in the **Review Vocabularies** dataset by selecting the **Implement Reviewed Strings** checkbox.

To mark a string as reviewed, select the string in the dataset. You may select multiple strings by selecting a row and holding the CTRL key as you select the rows with the relevant strings that you want to mark as reviewed. Click the **Mark As Reviewed** button in the toolbar.

Implementing the Change Proposals in the Vocabulary

The **Implement Changed Vocabulary Strings** functionality allows a solution designer to view all strings with proposed changes that were made in the **Review Vocabulary Strings** functionality. The solution designer can explicitly accept the proposal or can correct the proposal as needed. Please note that the solution designer must manually correct the string in the solution configuration based on the information about where the string occurs. Marking a string as changed is purely for documentation purposes and as a means to communicate about and track the changes to the string.

The following filters are available:

- **Source**: Select one or more vocabulary sources to display strings stored in the specified source. Strings from your solution configuration will typically be in the (`METAMODEL` or `GUIDEPAGES` vocabulary source although you may also modify protected strings in the (`ITPlan`, `Platform`, and `Extension` vocabulary sources.
- **Original**: Enter the string to display the string in the dataset. Use the wildcard * before and/or after the string if only a part of the string is entered.
- **Created Before**: Select a date to display all strings reviewed on and after the specified date.
- **Created After**: Select a date to display all strings created on and before the specified date.
- **Accessibility**: Select Protected to display standard strings or select Public to select custom strings.
- **Configuration Culture**: Select the language used to configure the original strings.
- **Include Obsolete Strings**: Select the checkbox to display strings that have become obsolete due to a proposed change being implemented.
- **Include Implemented Strings**: Select the checkbox to display strings that have been marked as implemented.

The following columns are displayed:

- **Source**: Displays the source of the vocabulary.
- **Accessibility**: Displays whether the string is a protected string created by Alfabet or a public string created in the context of your solution.
- **Original String**: Displays the string to be reviewed.
- **Length**: Displays the number of characters in the string. Strings may not be longer than 600 characters. If a text is longer than 600 characters, it will NOT be stored in the Alfabet database.

  If the string shall be translated, you should consider what the maximum length of the original language string should be in order to compensate for the translation. For example, a string in German is typically 6% longer than the same string in English.

- **Trailing Blank**: Displays a checkmark if a blank character exists at the end of the string.
- **Proposed String**: Displays the string that has been proposed to replace the original string.
- **Change Requested**: Displays a checkmark if the string has been reviewed and a change proposal has been made.
- **Occurrences**: Displays the configuration object where the string is used. If more than one occurrence exists, the number of occurrences will be displayed. Click the **Show Occurrences** button to view a dialog showing all occurrences of the string.
- **Creation Date**: Displays the date that the original string was created.
- **Reviewer**: Displays the user name of the user who reviewed the string.
- **Review Date**: Displays the date that the original string was reviewed.
- **Review Comment**: Displays the comment made by the reviewer about the proposed change.
- **Configuration Culture**: Displays the culture that the string was originally created for.

The following columns will be appended to the dataset if the **Include Implemented Strings** checkbox is selected:

1) **Implementer**: Displays the user name of the user who implemented the string.
2) **Implemented String**: Displays the changed string that replaced the original string.
3) **Implementation Comment**: Displays the comment made by the implementer about the change that was made to the original string.
4) **Replaces**: Displays the string that was replaced by the implemented string.

Understanding Where the String Occurs in the Solution Configuration

In order to understand where a string is used in the solution configuration, select the relevant row and click **Show Occurrences of String**. A dialog will open. The **Context Column** will display the type of configuration object followed by the name of the configuration object. The **Technical Information** columns displays additional information about the type of interface control where the string can be found. Click **OK** to close the dialog.

Implementing a Change for a Single String

You can implement a change for any string displayed in the dataset. The string proposed by the reviewer will be automatically displayed in the **Implemented Change** field in the **Implement Change for String** editor. If no further change is required, the string can be confirmed as implemented by clicking the **OK** button. If the proposed string needs to be changed, it can be changed in the **Implemented Change** field. The reviewer that proposed the change will be able to see the change that was implemented for the original string in the **Review Vocabulary String** functionality.

To implement a change to a selected string:

1) Select the row with the relevant string that you want to change and click the **Implement Change for String** button in the toolbar. The **Implement Change for String** editor opens.
2) Define the following fields, as needed.
   - **Original**: Displays the original string.
   - **Source**: Displays the source of the vocabulary.
   - **Obsolete Date**: If relevant, displays the date that the string was made obsolete in the vocabularies.
   - **Obsolete**: Displays a checkmark if the string is obsolete.
   - **Reviewer**: Displays the user name of the user who reviewed the string.
   - **Review Date**: Displays the date that the original string was reviewed.
   - **Proposed Change**: Displays the proposed change to the string.
   - **Review Comment**: Displays the comment provided by the reviewer about the change.
   - **Implemented Change**: The proposed change is displayed per default in this field. Modify the text as needed.

- **Implementation Comment**: If necessary, enter a comment about why the proposed comment was not accepted.

3) Click **OK** to save your changes:

Implementing Changes for Multiple Strings Simultaneously

You can simultaneously implement changes for multiple strings at once. You can implement a change for any string displayed in the dataset. The string proposed by the reviewer will be automatically displayed in the **Implemented Change** field in the **Implement Change for String** editor. If no further change is required, the string can be confirmed as implemented by clicking the **OK** button. If the proposed string needs to be changed, it can be changed in the **Implemented Change** field. The reviewer that proposed the change will be able to see the change that was implemented for the original string in the **Review Vocabulary String** functionality.

To implement changes to multiple strings:

1) Select a row and hold the CTRL key as you select the rows with the relevant strings that you want to change and the **Implement Changes for Selected Strings** button in the toolbar. The editor opens.
2) Define the following columns, as needed.
   - **Original String**: Displays the original string.
   - **Source**: Displays the source vocabulary.
   - **Trailing Blank**: Displays a checkmark if a blank character exists at the end of the string.
   - **Implemented String**: The proposed change is displayed per default in this field. Modify the text as needed.
   - **Implementation Comment**: If necessary, enter a comment about why the proposed comment was not accepted.
3) Click **OK** to save your changes:

Index

