Bizzdesign

Alfabet 11.13 - Platform Release Notes

Contents

Release Notes 11.13.0 Platform Changes

The Alfabet platform has been updated to include many new capabilities and enhancements to the functional scope of Alfabet.

New

The following is introduced to the Alfabet platform this release.

Next-generation data workbench v. 2 with better performance

Faster data handling. The data workbench implementation has been completely redesigned, delivering significant performance improvements across the application. Data loads are faster, and updates to data are processed more efficiently, resulting in a more responsive user experience.

Enhanced bulk update processing. The bulk update functionality has been refined to improve performance and scalability when working with large object selections:

- If 40 objects or less are selected, the dedicated bulk update editor will open and the uses can make the needed changes.
- If 41 to 300 objects are selected, a message is displayed describing the number of objects and extent of the changes that will be executed asynchronously. Users can continue working without waiting for all background activity to complete.
- If more than 300 objects are selected, bulk update in the user interface is not available. In such cases, a warning message informs the user that the maximum number of objects allowed for a simultaneous update has been exceeded and update via ADIF is recommended.

Targeted focus on new objects. When a new object is created, focus in the data workbench automatically shifts to make the object visible in the data workbench. Newly created objects are no longer displayed at the top of the visible object list.

Note: Following an update of the meta-model to Alfabet 11.13.0, the new data workbench implementation is enabled by default and becomes the standard data workbench for all Cloud installations. If issues are encountered with the new implementation, it is possible to revert to the previous data workbench version through a setting in the

`SolutionOptions` XML object available in the Alfabet Expand and Alfabet Administrator.

To simplify system administration and troubleshooting, information about the active data workbench implementation has been added to the **About Alfabet** dialog. The previous implementation is identified as **Data Workbench Version: 1**, while the new implementation is identified as **Data Workbench Version: 2**. This allows administrators and support teams to quickly verify which version is currently in use.

MCP server for AI-driven access to enterprise architecture data

The new Alfabet MCP Server provides secure AI-driven access to enterprise architecture data through natural language, eliminating the need for specialized query knowledge and reducing dependency on architecture teams. Alfabet's MCP Server translates AI requests into structured REST API calls and provides flexible, on-demand reporting across applications, technologies, projects, business capabilities, and related data.

Built-in authentication ensures secure access, supporting both username/password and OAuth authentication. The MCP Server is read-only and prevents any modifications to repository data. Access is restricted to users with explicitly configured permissions for the relevant REST services.

The MCP Server is currently available as a beta feature for administrative users only. User interface visibility rules are not yet enforced, which may allow users to access information about objects that are not visible in the standard UI. Access should therefore be granted only to trusted users with appropriate permissions.

Data-triggered actions for user-initiated changes

Data-triggered action rules for user interactions in standard views.

Stereotypes have been introduced for data-triggered action rules. The **View-Based Data-Triggered Action Rule** rule type enables the execution of data-triggered actions in standard views in response to button and inline editing interactions. For button-based actions, the data-triggered action is executed in the context of the object on which the button operates. For inline editing scenarios, the action is executed in the context of the base object of the class-based content area in which the

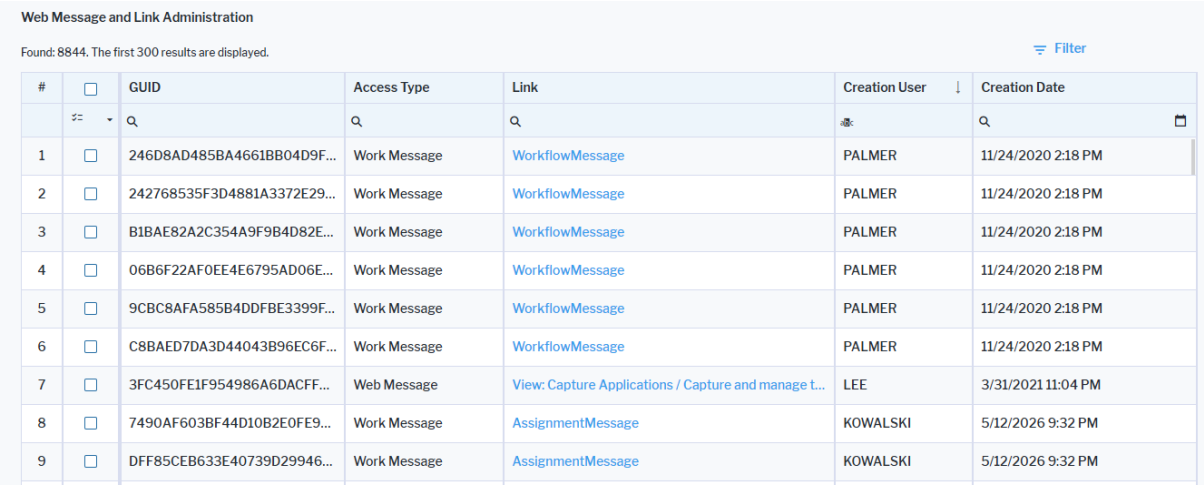
view is embedded. This enhancement enables automated processing and validation logic to be applied directly during user-driven data changes.

The **Operation-Based Data-Triggered Action Rule** stereotype has been introduced for existing data-triggered action rules targeting data workbenches and content areas. On migration to Alfabet 11.13, all existing data-triggered action rules will be automatically migrated to the **Operation-Based Data-Triggered Action Rule** stereotype.

Enhanced usability of the data-triggered action rule editor. The data-triggered action rule editor now provides additional context for generic class operations, including button captions and base objects. This makes it easier to distinguish between similar operations, such as different relation-creation actions. For example, users can now more easily identify and select the correct relation-creation operation when multiple operations exist for adding relations to different target object types.

Restrict access to shared links

Links to the user interface are included in email notifications when tasks are assigned to users. Users can also generate links to specific views for sharing with colleagues or embedding in third-party applications. To enable access, these links rely not only on information contained in the URL but also on data stored in the database. As a result, links do not expire by default.



#	GUID	Access Type	Link	Creation User	Creation Date
1	246D8AD485BA4661BB04D9F...	Work Message	WorkflowMessage	PALMER	11/24/2020 2:18 PM
2	242768535F3D4881A3372E29...	Work Message	WorkflowMessage	PALMER	11/24/2020 2:18 PM
3	B1BAE82A2C354A9F9B4D82E...	Work Message	WorkflowMessage	PALMER	11/24/2020 2:18 PM
4	06B6F22AF0EE4E6795AD06E...	Work Message	WorkflowMessage	PALMER	11/24/2020 2:18 PM
5	9CBC8AFA585B4DDFBE3399F...	Work Message	WorkflowMessage	PALMER	11/24/2020 2:18 PM
6	C8BAED7DA3D44043B96EC6F...	Work Message	WorkflowMessage	PALMER	11/24/2020 2:18 PM
7	3FC450FE1F954986A6DACFF...	Web Message	View: Capture Applications / Capture and manage t...	LEE	3/31/2021 11:04 PM
8	7490AF603BF44D10B2E0FE9...	Work Message	AssignmentMessage	KOWALSKI	5/12/2026 9:32 PM
9	DFF85CEB633E40739D29946...	Work Message	AssignmentMessage	KOWALSKI	5/12/2026 9:32 PM

To support security and compliance requirements, access via existing links can now be revoked. A new **Web Message and Link Administration** (`Admin_AlfaWebMessages`) view allows administrators to deactivate previously generated links, preventing access through

outdated or no longer authorized links. This provides greater control over shared content and helps customers enforce link access policies.

Enhanced role type configuration and access management

Role type configuration has been enhanced to provide greater flexibility and control at the object class level. Administrators can now override role type settings through class settings, allowing role behavior to be tailored for specific combinations of object classes and user profiles. This makes it possible to define how roles are assigned and managed on a per-class basis rather than relying solely on global role type definitions.

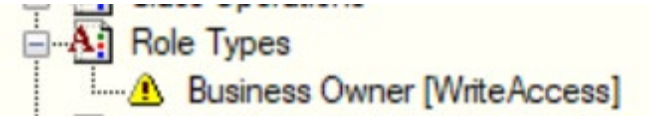
Greater flexibility in access management. Access management settings support more advanced configuration options. Alongside general editability and visibility settings, administrators can define conditions that determine whether a role is editable or visible. These conditions can be based on property values of the object for which the role is defined or on characteristics of individual users, enabling more granular access control.

Class setting-based role type assignment. Role type assignments to property groups can be configured via class settings. When a property group assignment is defined both in the **Role Type** and in class settings, the class-specific configuration takes precedence. Configuring property group assignments in class settings provides the additional benefit that the assigned role types participate in the property group ordering defined within the same class configuration.

Further customization options have been added for role types:

- Modify the caption and hint text displayed for a role type.
- Enable or disable the **Advanced Search** functionality in the selector to select users or organizations for a role. The underlying selector type for **Advanced Search** can be changed for role types defined to create roles for organizations only or users only.

A new consistency check has been added to the class settings configuration to simplify the management of role type overwrites. The validation automatically detects whether role types configured in the class settings are no longer available because they have been deleted or removed from the object class. When such a configuration issue is detected, a warning icon is displayed on the affected role type node, helping administrators quickly identify and correct invalid configurations before they cause unexpected behavior.



In addition, a new **Check Consistency** action is available on the class settings node. The consistency check validates the role type configuration and displays all detected inconsistencies in a validation dialog, enabling administrators to identify and resolve configuration issues more efficiently.

Control property editability using conditions

The editability of object class properties can now be controlled through conditions. Users can be explicitly granted or denied permission to edit data based on criteria such as their role for an object or their membership in a user group. Conditions can also be used to make a property read-only depending on the value of another object class property. Editability conditions are configured in the same way as the existing visibility conditions and can be defined using the new **Editability Conditions** attribute in the object class property configuration within the class settings.

Editability conditions cannot grant edit access to object class properties if **Read Access** or **No Access** is specified in the **Access in User Interface** attribute of the property. They can only further restrict editing of properties configured if the attribute is set to **Write Access**.

Extensions to generic operations

Configurable copy, version, and variant creation of objects. A new generic operation type `Create As Copy` enables customers with access to Alfabet Expand to configure buttons for the user interface to create copies, versions, and variants of objects for any object class.

The generic operation determines which object properties and relationships are transferred from the source object to the newly created object. During the copy process, property values can also be updated automatically from the target object back to the source object. For example, when creating a new version of an object, a property on the original object can be updated to reference the newly created successor version.

The operation supports flexible handling of relationships to other objects:

- Copy relationships only by creating equivalent references from the new object to the existing related objects.
- Copy related objects as well and automatically establish relationships between the copied objects and the newly created objects. In addition, you can configure which object class properties of related objects are copied when new related objects are created.

Enhanced configuration of affected architecture relationships. The new generic operation types `Add Complex Relation`, `Create and Add Complex Relation`, and `Delete Complex Relation` can be specified to maintain objects that are affected by or relevant to a project, demand, value node, IT policy, or measure type. Business data usage and project bucket allocation can also be managed via the new generic operations.

Customers with access to Alfabet Expand can configure buttons that allow users to create architecture relationships to objects of the class Project, Demand, Value Node, IT Policy, and Measure Type. Buttons can be specified to create new or add existing objects with the architecture relationship or remove an object from architecture relationship. Architecture specifications for these object classes are implemented through an intermediate object class that maintains the relationship between the selected object and the associated architecture object. This intermediate relationship object is not exposed in the user interface.

Improved control of button visibility. In the specification of the `Create and Add Relations` generic operation, evaluation of the object class to which the new object is added required both the specification of the **Base Object Class** attribute and the enablement of the **Object-Based Operation Context**. This has been changed and the base object is evaluated independent of the setting for **Object-Based Operation Context**. The button can now be configured to be displayed in data workbenches only.

Amazon Bedrock Guardrails support for connections to the text generation AI

Amazon Bedrock Guardrails® blocks harmful content and protects private data in generative AI applications. The AI text generation feature available in Alfabet has been amended with the optional ability to protect requests to the AI service via Guardrails. Responses are directly routed back to the Alfabet application. Customers using the AI service provided

by Alfabet on premise must update the server variable configuration with a new server variable provided by Bizzdesign B.V.

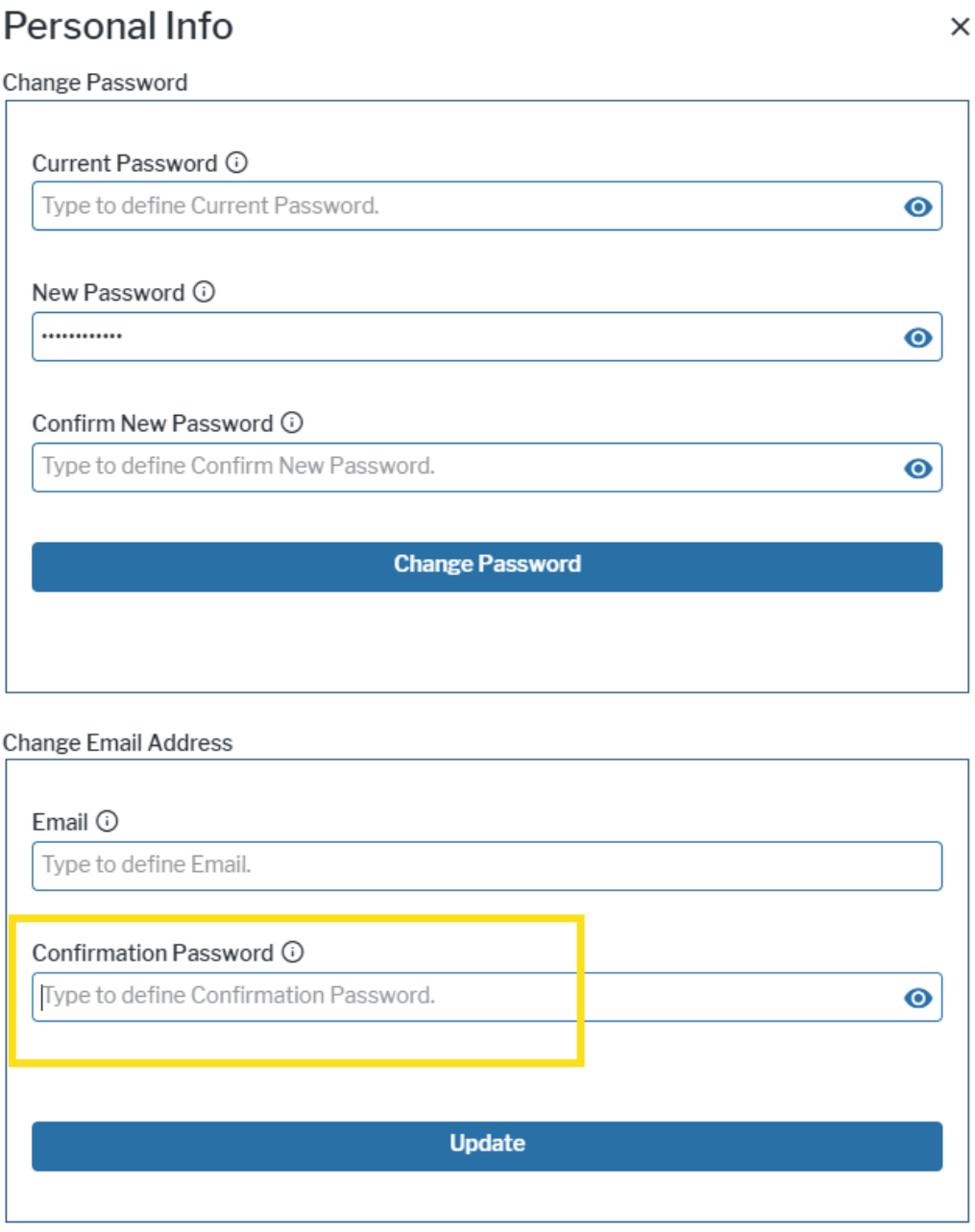
Customers using a connection to their own AI service account on Amazon Bedrock can access Alfabet Expand to configure the XML object `AIServicesConfig` to use Guardrails. Customers must have a Guardrails configuration in Amazon Bedrock for the AI service they want to use.

Improvements

The following enhancements to the Alfabet platform with this release.

Password-protected email address changes

Users who sign in to Alfabet with a user name and password can request a password reset if they forget their credentials. Temporary login credentials are then sent to the email address associated with their account. To help prevent unauthorized changes to this critical contact information, additional security measures have been introduced.



For Alfabet-managed users authenticated through user name and password, changes to the registered email address now require password verification. Before an email address can be updated in the user menu, the currently signed-in user must re-enter their password in the new **Confirmation Password** field in the **Personal Info** dialog. This additional authentication step helps protect user accounts by ensuring that only authorized users can modify the email address used for password recovery and account-related communications.

Better usability in data workbenches

The following usability enhancements are available in both data workbenches version 1 and verion 2.

Simpler redesign of data set columns. Reorder columns in the data workbench via the new drag-and-drop functionality, eliminating the need to use the **Structure** editor.

Improved readability in boards. A new zoom toolbar in board visualizations enhances readability by allowing users to focus on specific board elements.

Easy restructure of Gantt columns. Reorder columns in Gantt visualizations in data workbenches via the new drag-and-drop functionality. Users can move the columns included in the **Object Attributes to Display** definition of the Gantt visualization. The changed column order is saved in the user context settings as well as when the visualization is shared with other users.

Expanded lock indicator for read-only objects

The  lock icon is now also displayed in tabular views to indicate that an object is read-only and cannot be edited when opened. Previously, this indicator was only available in data workbenches.

Consistent enforcement of mandatory properties

In the standard class model, object class properties that are required to satisfy standard Alfabet class model constraints, such as class keys, are designated as mandatory. In the Meta-Model tab explorer in Alfabet Expand, these properties are identified by a yellow background.

Mandatory properties defined in the meta-model are now consistently enforced as non-nullable across all editing interfaces, with the exception

of custom editors. Users can no longer clear or set required property values to null, ensuring that mandatory data is always maintained. This enhancement improves data integrity, supports compliance with model constraints, and provides consistent behavior throughout the application. For custom editors, a configuration option for specifying properties as mandatory already exists and remains unchanged. Fields in custom editors enforce strict null handling if the **Presence** attribute of the field is set to **Always**.

Workflow-specific edit permissions

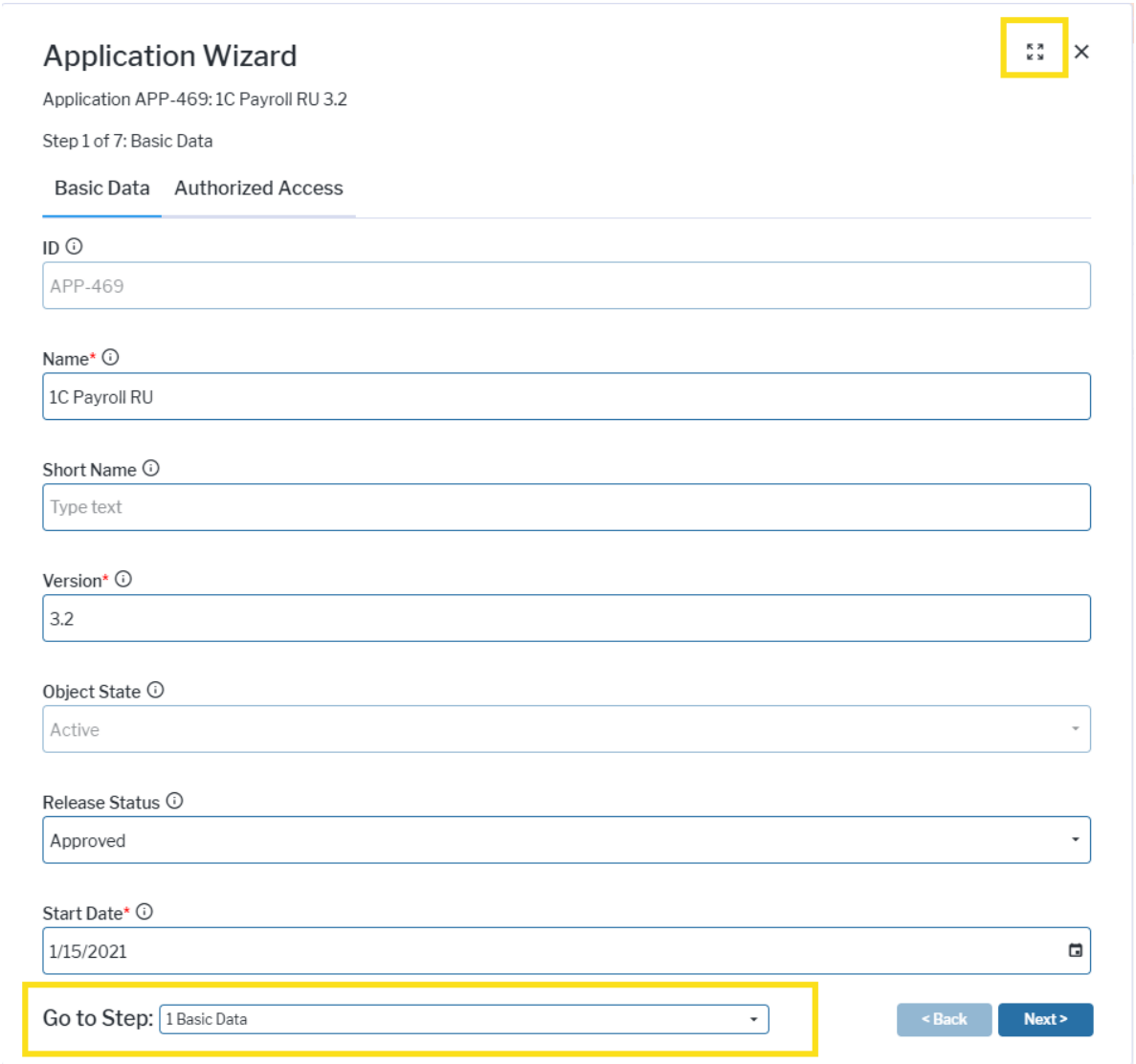
The write permission concept for workflow steps has been enhanced to better support approval and review processes. Users assigned as responsible users for a workflow step can now edit the object associated with that step, even if they do not otherwise have write access to the object. This allows workflow participants to perform actions such as approving, rejecting, or updating workflow-related information without requiring broader object maintenance permissions. Standard object access permissions continue to apply outside the workflow context.

Note: Standard and custom editors maintain their own permission model and therefore remain unavailable within workflow steps. Any data that needs to be edited as part of a workflow must be made available through a property group in a guided data view or content area.

User-friendly wizards

Flexible resizing of wizard. A  resize button has been added to the upper right of wizard windows. Clicking the button allow users to quickly enlarge the wizard for a better overview of the views and editors in the wizard.

Improved readability of next wizard steps. The size of the **Go to Step** field in the wizard has been made extended so that long wizard step captions are easier to read.



Immediate refresh of data quality information

Data quality information is now updated automatically when a user resolves a data quality issue in a content item. The updates are reflected immediately in both the data quality window and the header, eliminating the need to refresh the page manually.

Note: This does not apply to property groups that are configured to require a manual property group update for data quality rule evaluation.

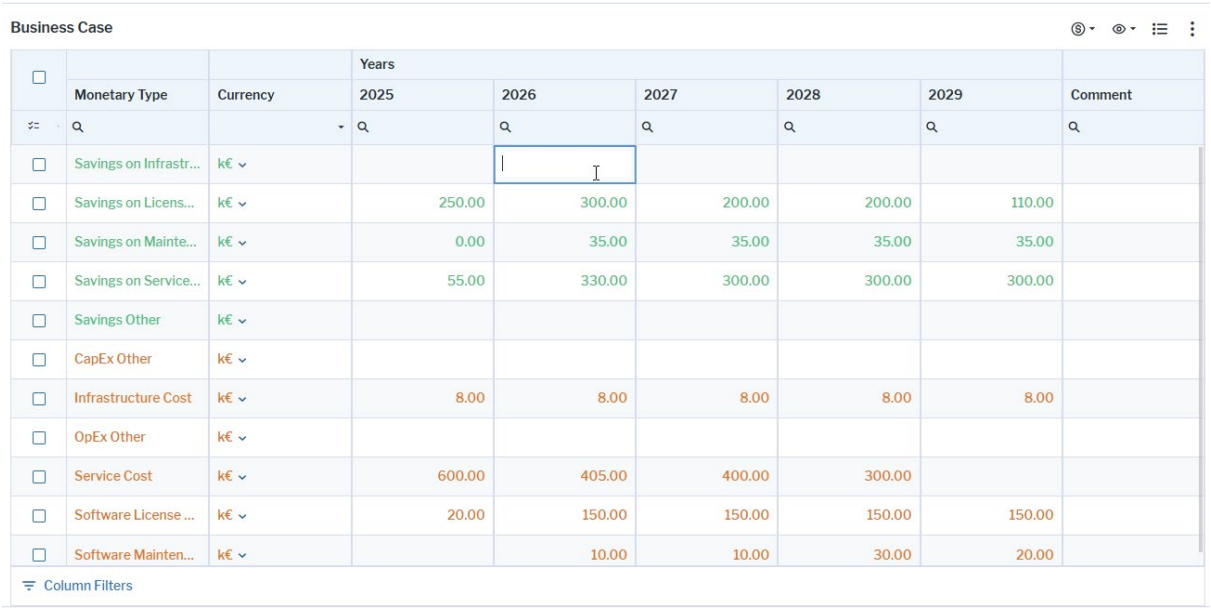
Enhanced user comments

The **User Comments** functionality now allows users to edit their own comments after posting. When a user hovers over one of their comments, an edit button appears in the upper-right corner next to the date and

time. Selecting the edit button enables the user to modify the comment and save the changes.

Improved usability for cost maintenance

Optimized default column width in cost reports. The default column width in cost reports has been reduced from 200 pixels to 100 pixels. This change minimizes unnecessary white space and improves data visibility, allowing users to view more information on screen without excessive horizontal scrolling. Users can still resize columns manually by dragging the column border. Any adjustments are automatically saved in the user's personal settings and will be retained the next time they open the cost report.



Inline editing available for project costs. The following views have been enhanced:

- **Business Case** (`PRJ_BusinessCase` and `PRJ_BusinessCaseSimple`) views. Users can edit cost values directly in the dataset if the **View** > **Aggregate Costs Along Project Hierarchy** option for the view is not activated.
- **Benefit Tracking** (`PRJ_BenefitTracking`) view.
- **Benefit Realization/Cashout Planning** (`PRJ_Cashout`) view.

Better filtering for cost views. The filter behavior in the **Project Costs** view for projects and the **Cost Accrual** views for applications and service products has been improved. When the **Start Year** or **End Year** filters are

left blank, the views now default to displaying the respective start/end year of the selected project, application, or service product. In addition, **Start Year** and **End Year** filter settings are now maintained independently for each cost report instance, preventing filter values from being shared across cost views for other assets of the same class and enabling asset-specific reporting based on individual timelines.

Usability enhancements for project resource planning

Project resource planning views have been enhanced to improve usability and clarity. Column names and actions have been renamed to better reflect their purpose, additional resource information is now available directly in the views, and error messages have been improved.

- **Work Breakdown Structure** (`PRJ_WorkBreakdownStructure`) view: The **Time Schedule** column has been renamed to **Work Breakdown**, and the **Resource** column is now called **Provider**. In the toolbar, **Create Skill Request** has been renamed to **Request Skills**.
- **Project Resource Planning** (`PRJ_ResourcePlanning`) view: Two new columns, **Provider** and **Count and Daily Rate**, have been added to provide additional resource information. The **Task Schedule** column has been renamed to **Resource Breakdown**. In the toolbar, **Reset Project Resource Plan** is now called **Revert to Work Breakdown**, and **Transfer Schedule from Project to Resource Plan** has been renamed to **Transfer Skill Requests from Work Breakdown**.

Support of dark mode in AI help assistant

The AI-based Help Assistant has been updated to provide full support for dark mode. All content, controls, and visual elements have been updated to maintain appropriate contrast and readability, helping ensure a consistent and accessible user experience across light and dark themes.

Improved document management capabilities

Users can add information about a document in a new **Description** field when documents are uploaded via the **Attachments** (`Object Attachments`) or **Internal Documents** (`IDOC_MANAGER`) views.

A new **Description** (`DESCRIPTION`) object class property has been added to the **Document** (`ALFA_IDOCUMENT`) object class and the standard editor to edit documents. All properties of the **Document** (ALFA_IDOCUMENT) object class are now exposed for use in editability

and visibility rules. Administrators can leverage these properties to selectively grant or restrict document upload and management capabilities, including through button overrides defined in the object class settings. This change is an enhancement implemented for ticket ASD-9976.

Extended drill-down navigation from indicator cards

Indicator cards now support drill-down navigation to content areas in addition to standard views, configured reports, and data workbenches. The **Indicator Type** editor has been extended to allow content areas as drill-down targets.

When a class-based content area is selected, the originating object is automatically passed as context, enabling embedded views, workbenches, and reports to display object-specific information. This enhancement provides greater navigation flexibility and delivers more relevant, context-aware insights throughout Alfabet.

Enhanced control of diagram visibility

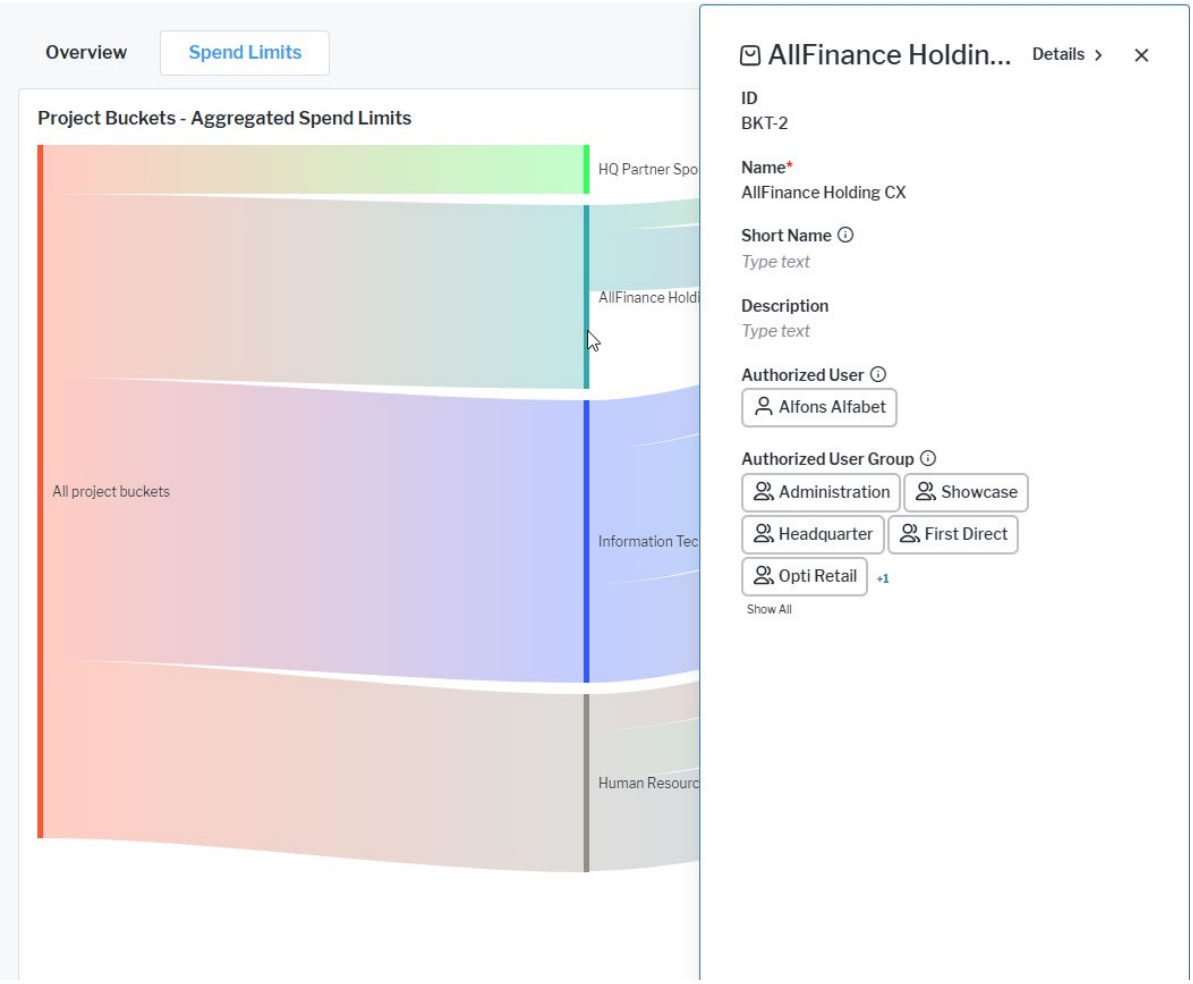
The **Diagram Designer** now includes a **View Settings** editor that helps make large and complex diagrams easier to understand by displaying only the information relevant to a specific audience.

The **View Settings** editor lists all object classes and relationships that can be shown or hidden in a diagram. Relationships between object classes are displayed using arrow notation, such as Application → Application or Application → Peripheral. By default, all available object classes and relationships are visible. The user designing the diagram can selectively hide specific object classes or relationship types, such as information flows or local components, to reduce visual complexity and focus the diagram on the information that matters most.

Richer data analysis in Sankey diagrams

Object inspection supported in Sankey diagrams. Users can open the object inspection panel by single-clicking on any object node in the Sankey diagram. This enhancement aligns the interaction model with other report types, providing a more consistent and intuitive user experience when inspecting objects. This change is a resolution for ticket ASD-11201.

Configurable drill-down navigation in Sankey diagrams. Sankey diagrams now support configurable drill-down navigation to designated views, enabling users to navigate directly from source nodes, target nodes, and links to relevant data views. New configuration options in the report assistant allow administrators to define and customize the navigation behavior for each diagram element, creating a more tailored and efficient analysis experience. These enhancements provide greater flexibility when analyzing relationships and exploring underlying data.



.

Manage unassigned objects in Kanban reports

Configured Kanban reports can now display objects that are not assigned to a row or column because the object property used for the row or column definition has no value. These objects can be shown in dedicated **Unassigned** rows and/or columns, making it easier to identify and manage incomplete assignments.

Administrators can configure unassigned rows and columns to appear either at the beginning or the end of the Kanban board. Objects can also be moved from an unassigned row or column to a regular one. When this occurs, the corresponding object property is automatically updated to reflect the new assignment.

A new **Unassigned Row/Column** section has been added to the Kanban report assistant settings:

- **Enable Unassigned Row** and **Enable Unassigned Column** attributes control whether dedicated rows or columns for unassigned objects are displayed.
- **Unassigned Row Position** and **Unassigned Column Position** attributes determine whether unassigned objects appear at the start or end of the board.
- **Unassigned Row Caption** and **Unassigned Column Caption** attributes allow administrators to define the labels shown for unassigned rows and columns.

Richer context for AI text generation

AI Text Generation now includes object class properties of the type `Reference` as well as calculated properties for the context used to generate text. This enhancement results in richer, more accurate generated content. The **AI Text Generation Related Properties** attribute is available in Alfabet Expand for both the object class as well as class settings, offering greater flexibility and control over which properties contribute to AI-generated text.

Enhanced visibility condition handling

Real-time update of object inspection. Visibility conditions defined for object class properties are now applied dynamically in object inspection windows. For example, if a user modifies data in a way that causes the visibility condition for a property to no longer be met, the property is automatically removed from the inspection window immediately. This update is reflected in real time, eliminating the need to close and reopen the inspection window to refresh the displayed properties.

Easier query validation. Queries used in visibility conditions can only be processed if they meet certain requirements: they must be non-complex, and each column in the result set must have a unique name. If a query does not meet these requirements and cannot be processed, the entire

column controlled by the visibility condition is removed from data workbenches.

To simplify troubleshooting, logging has been added for queries used in visibility conditions. When the logging level is set to Warning or higher, a warning message is written to the configured NLog system log target whenever a query cannot be processed.

In addition, the .log file generated during a meta-model update from an earlier Alfabet release to Alfabet 11.13.0 now includes warnings for all invalid visibility condition queries detected in the meta-model configuration. This enhancement helps administrators identify and correct configuration issues more efficiently.

Enhanced usability in runtime editors

Runtime editor property sorting. A new **Sort Runtime Editor Properties** attribute has been added to the **Properties** node of class settings. It enables sorting of object class properties displayed in runtime editors, including properties exposed through class setting property overwrites. This improvement is a fix for ticket ASD-10452.

Display calculated properties in runtime editors. Calculated object class properties can now be included in runtime editors and object inspection windows for informational purposes. Because these properties are derived automatically, they remain read-only and cannot be edited by users. However, displaying calculated values alongside editable properties provides additional context and can help users make more informed decisions when entering or updating data.

When calculated object class properties are enabled for runtime editors, they can also be reordered using the new sorting capability available in the **Properties** node of the class settings configuration, allowing administrators to optimize the presentation of property information.

Meta-model updates for risk management classes

- Standard object class properties defined in the Alfabet meta-model have been added to the `Risk`, `RiskManagementTemplate`, and `RiskSet` object classes. These properties are `ID`, `Status`, `Documents`, `ResponsibleOrganization`, `CREATION_DATE`, `CREATION_USER`, `LAST_UPDATE_USER`, `LAST_UPDATE`, `ICON`, and `ALFA_LICENSE_POLICY`. By making them available out-of-the-box, this enhancement simplifies object class customization and enables

the use of these attributes in integration scenarios, such as ADIF imports and RESTful service-based data exchange, without requiring the creation of custom object class properties.

- For the `Risk`, `RiskManagementTemplate`, and `RiskMitigation` object classes, the `ProbabilityValue` and `DamageValue` properties have been changed from `Private` to `Protected`, enabling range definitions and enhanced configuration options.
- The `Risk`, `RiskGroup`, `RiskManagementTemplate`, `RiskMitigation`, `RiskMitigationCategory`, and `RiskMitigationTemplate` object classes can be enabled for data capture templates. This change is a resolution for ticket ASD-11606.

Enhanced data capture for standard platforms

Several standard Alfabet meta-model settings have been updated to support the creation of data capture templates that include a broader range of platform-related properties.

The **Enable for Data Capture Templates** attribute has been set to **True** for the **Embedded Standard Platforms** (`ConsistsOf`) and **Associated Master Platforms** (`MasterPlatforms`) object class properties. In addition, the **Access in User Interface** attribute for these properties has been changed to `WriteAccess`, enabling users to view and maintain these relationships directly within data capture templates.

The **Diagrams** (`Diagrams`) object class property has also been updated, with its **Access in User Interface**attribute set to `ReadAccess`.

Alfabet Expand usability enhancements

- A text editor has been implemented for the **Description** attribute of categorized visual items, providing a more user-friendly experience when editing long descriptions.
- The private ADIF jobs for execution of data import and export in the scope of the interface with ITPedia® have been hidden from the **ADIF Schemes** explorer in Alfabet Expand to enhance the clarity of the explorer. The ITPedia-related ADIF schemes were not editable by customers and only executable via Alfabetinternal code.

Clean-up of deprecated functionalities

- Deprecated cyclic monitoring events and their associated private management events have been removed from Alfabet Expand. This streamlines event selection lists and improves usability when configuring data-triggered actions and other event-based configurations.

Improved AMM file generation

The AMM file generation editor has been enhanced by removing deprecated features and adding support for new configuration options.

- The **Business Question** and **Business Question Group** object classes have been removed from the **Reference Data** tab in the AMM file editor. These object classes are deprecated and are no longer supported by the updated meta-model configuration. Removing them simplifies the editor, reduces potential confusion, and ensures alignment with current configuration standards and best practices.
- A new **Replace Target Help Context Content** checkbox has been added to the **Meta-Model Content**tab of the AMM file generation editor. When selected, this option removes the existing context-sensitive help content from the target database before importing the content defined in the AMM file. This option is currently applicable only for AMM files used to configure predefined solutions for Alfabet IT Transformation Server licenses. Creating and managing custom context-sensitive help in AMM files is not currently supported for customer-defined configurations.

Fixes

- Resolved an issue where the **Add Document** option was incorrectly displayed in the **Attachments** view after closing the **Create Web Link** editor, despite being disabled in the **Configuration** editor. This fix is a resolution for ticket ASD-11027.
- It was not possible to change the default role type that was automatically assigned to a person when a new object was created. To address this, the access rights for the object class properties for the class **ClassConfiguration** have been updated. The **Access in User Interface** attribute has been set to **ReadAccess** for the properties **ClassName**, **Stereotype**, and **ShowName**. All other properties are assigned the value **WriteAccess** so that reference data such as evaluation types, cost types, and default role types can be configured by an administrative user in the Alfabet user interface. This fix is a resolution for ticket ASD-10855.
- If **View** > **Show Embedded Standard Platforms** was selected in the **Platform Architecture** or **Standard Platform Architecture** view, the legend included the color coding for the base standard platform. This has been corrected. When **Show Embedded Standard Platforms** is selected, the **Base Standard Platform** section is no longer displayed in the legend. Instead, an **Embedded Standard Platforms** section is shown, displaying the appropriate color coding for the embedded standard platforms.
- Copy and paste has been enabled for customer-defined generic operations to easy the definition of multiple generic operations with similar settings.
- Fixed the display of breadcrumbs to show the active view in a darker color. This fix is a resolution for ticket ASD-11568.
- Resolved incorrect rendering of configured bar chart reports. Reports are now displayed correctly without numbers between the bars. This fix is a resolution for ticket ASD-11742.
- The export of data workbench data has been corrected to preserve the object order as displayed in the workbench. In addition, exported data is now generated in the language currently selected for the user interface, ensuring that localized content is exported consistently with the user's active language settings. This fix is a resolution for ticket ASD-11671.

- Issue resolved with performance issues caused by the resizing of the Alfabet navigation panel or the complete browser window. This fix is a resolution for ticket ASD-11626.
- Fixed an issue that could cause column header widths to be calculated incorrectly when grouped configured reports were resized and scrollbars appeared. This fix is a resolution for ticket ASD-11545.
- Fixed issues with the expand and collapse functionality in grouped datasets. Action icons are now displayed consistently, and grouped sections behave as expected. This fix is a resolution for ticket ASD-11547.
- User names are now displayed consistently across all **Responsibilities** views in the format First Name Last Name, replacing the previously inconsistent comma-separated format. This fix is a resolution for ticket ASD-11142.
- If a role type was configured to allow one person and/or one organization to be specified for a role, it was not possible to change the user or organization without first explicitly deleting the existing role. The assignment of roles has been changed to allow the direct overwrite of an existing role. The user must confirm a warning that the existing setting will be overwritten.
- Fixed the generic class operation for the start of a workflow to hand over the current object to the workflow as workflow base object.
- Fixed an issue where links in workflow notification emails opened the workflow step content area instead of the configured guided data view. Workflow steps now open the correct guided data view when accessed from email links.
- Fixed an issue where the **Validate** context menu action for workflow template configuration in Alfabet Expand did not fully validate workflow steps that had a guided data view or content area assigned. Validation now correctly checks the configuration of these workflow steps and reports any detected issues. This fix is a resolution for ticket ASD-11182.
- Fixed the display of buttons for the AI text generation and translation functionalities to show up on **Description** fields in guided data views for workflow steps.

- Fixed the issue with an empty subject line in emails sent for consolidated workflows. If the **Notification Subject** attribute of the workflow does not specify a subject string, the text `New Workflow Activity from Alfabet` will be shown in the subject line for emails sent for consolidated workflows. This fix is a resolution for ticket ASD-10558.
- Fixed the incorrect display of value streams in the **Value Stream Groups** explorer. This fix is a resolution of ticket ASD-11586.
- Resolved an error that occurred when detaching architecture elements in the affected architecture of projects, demands, and strategy. This fix is a resolution for ticket ASD-11546.
- Fixed an error that occurred when users tried to set the column filters in the **Operational Expenses** view. This fix is a resolution for ticket ASD-11021.
- Fixed performance issues that occurred when the business question **What is our target architecture?** was rendered. To address this issue, the rendering behavior for large matrix reports has been optimized across the platform. To ensure consistent performance and responsiveness, a maximum limit has been introduced for the number of matrix cells that can be rendered. When a report exceeds this limit, users are informed through a message that the report size exceeds the supported rendering threshold. This fix is a resolution for ticket ASD-11181.
- Resolved an issue where links to other diagrams were not persistently stored when added to object nodes in the Alfabet diagram designer.
- Duplicate entries for class configurations were written to the `CLASSCONFIGURATIONS` table because of insufficient unique key configuration. Unique key configuration now checks for name and stereotype name. During the migration to Alfabet 11.13, the `CLASSCONFIGURATIONS` table will be scanned for duplicates and the older entries will be removed. The respective entries in related configuration object tables, like the `ROLETYPECONFIG` table storing the role type configuration for the class settings, will also be removed. Warnings are written to the meta-model update log file.
- Resolved an issue where report button overrides configured through the **Configure Buttons** panel did not persist reliably. To address this, the storage location for configured report button overrides has been

moved from the **Presentation Objects** node to the **Reports** node of the class settings. Existing report button overrides have been automatically migrated to the new location to ensure consistency and preserve all existing configurations. The **Configure Buttons** functionality has also been updated to store and maintain button overrides in the **Reports** section, ensuring that changes made through the user interface persist correctly.

- While it has not been possible to manually create multiple user profiles with the same name, validation for duplicate user profile names was missing in certain import mechanisms. A database-level uniqueness constraint has been introduced for user profile names to strengthen data integrity. During the update to Alfabet 11.13, any duplicate user profile names detected are automatically renamed with the suffix `_dup`. All affected profiles are documented in the meta-model update log.
- Fixed an issue where the configured default value for the `ObjectState` property of the `Application` object class was ignored during object creation. The default value is now applied correctly by the editor. This fix is a resolution for ticket ASD-11075.
- Corrected the execution messages for ADIF export jobs to include the name of the executed ADIF job. This fix is a resolution for ticket ASD-11501.
- Corrected the console help for the ADIFConsole.exe application to display only currently valid command line options. The `-synchronous`, `-heartbeat`, `-recipientmail`, and `-sendermail` command line parameters are obsolete and have been removed from the command line help. This fix is a resolution for ticket ASD-11335.
- In the `alfasettings.json` example configuration files for the Alfabet Web Application, `font-src 'self' data:;` has been added to the `ContentSecurityPolicy` specification to avoid issues with fonts in the new Gantt chart component. This is currently only relevant for Cloud-based Alfabet FastLane instances.
- Resolved an issue where translated strings for new objects were not transferred to a target database when the AMM file was configured to include translations. This fix is a resolution for ticket ASD-11666.

Forthcoming Changes

- The `EditableClassView` report type template will be deprecated and removed in Alfabet release 11.14. Its functionality is now provided through data workbenches. To ensure a smooth upgrade to Alfabet release 11.14, review any existing reports based on the `EditableClassView` template and replace them with equivalent data workbenches before upgrading. This will prepare you for uninterrupted access to the same functionality in Alfabet 14.0.
- The integration interface with CentraSite® and WebMethods® API Portal and WebMethods API Gateway will be deprecated and removed in Alfabet release 11.14.