Bizzdesign

Alfabet 11.12 – Platform Release Notes

Contents

Release Notes 11.12.0 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.

Smart AI-powered content and translation

Alfabet introduces advanced AI capabilities designed to enhance productivity by enabling faster comprehension of key information and more intuitive interaction with content. By combining intelligent text generation with secure, multilingual translation, this feature allows users to create, understand, and refine information with greater speed, clarity, and consistency. Users can generate text in content areas and guided data views in the user interface language that they are working in as well as translate texts to any of the other languages supported in Alfabet. Built with security at its core, all translations are processed through protected, compliant mechanisms, and the data in your Alfabet repository is never used to train AI services, ensuring privacy at all times. This feature is available on request. AI service hosted on Amazon Bedrock® is provided for this feature at no extra cost for Alfabet customers, or you can integrate with other AI services to meet your specific requirements.

AI-powered text generation for key fields. AI-driven text generation is available to generate high-quality content on an object-by-object basis for specified text fields such as descriptions, demand benefits, and other textual details. The AI analyzes relevant context such as titles, metadata, and configuration comments, to produce clear, relevant, and stylistically consistent text in the user's preferred language. The solution designer can go to Alfabet Expand to activate custom or protected properties of the type **Text** by setting the new **Local Settings** > **AI Text Generation** attribute to **True**. In the **AI Text Generation Related Properties** field, the solution designer can specify which other properties shall be used as context for generating the text. For example, to support the generation of text for the **Description** property of the class **Application**, the solution designer might specify the application name, architecture type, and development type to be used as context to generate an application's description. The specification of AI text generation can also be refined on the level of class settings to ensure that the relevant context is used for a class stereotype, for example.

For enabled properties, the **Generate with AI** functionality will be available in the 3-dots menu of the text field in content areas and guided data views. Once the text

generation is triggered, the AI service will generate the text in the language of the user interface. Users may modify the generated text and save the text or retrigger the text generation again based on the edited text.

Secure AI-Driven Translation for Text Fields The new AI-translation capability allows users to work with content in their native language thus removing language barriers, clearer communication and better decision-making occurs across global teams. Properties of the type **Text** that have been enabled for text generation can be automatically translated in any supported language via the **Translate Text with AI** that is available in the 3-dots menu of the text field. Translations are handled through a secure, compliant process that protects sensitive information at all times. This ensures consistent understanding and adoption of content worldwide while maintaining trust and data security.

Any field for which AI text generation and translation is enabled must reside in its own dedicated property group containing no other properties.

Compact mode for enhanced visibility

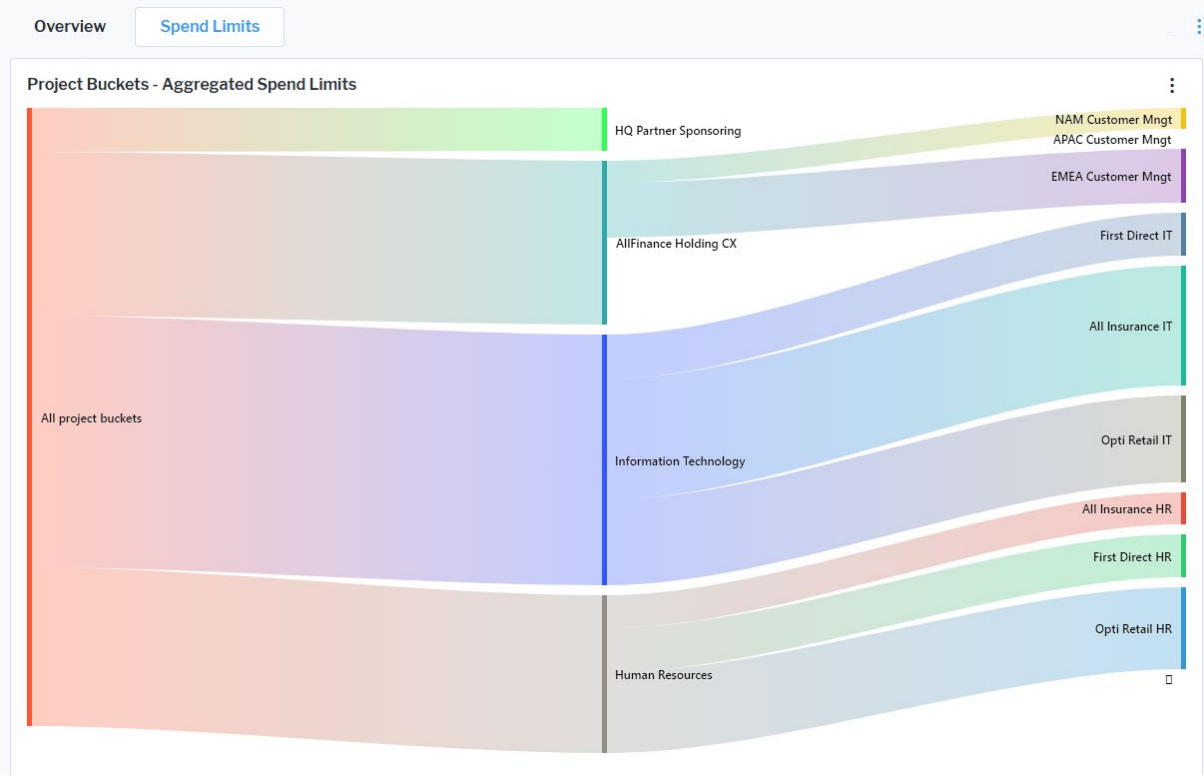
A new compact mode rendering is introduced to improve screen real estate, especially for large datasets and low-resolution displays. Designed especially for users with data-heavy roles, the compact mode delivers a sleek, zoom-like experience that fits 30–100% more information on screen, minimizing scrolling and maximizing productivity.

Compact mode streamlines the masthead to maximize usable space and brings a crisp, high-density experience across the interface. Standard datasets and expandable tables are fully optimized with slimmer headers, tighter rows, narrower columns, and refined font scaling. Content areas display more information at a glance with reduced spacing and padding, and customized content area layouts remain fully intact. All inline editing functionalities are preserved in datasets, content areas, and guided data views.

Each user can decide for themselves whether to use the compact mode by enabling the new **Compact Mode** checkbox in their **User Settings**. The **Compact Mode Level** field lets users choose between the default compact mode rendering or an even higher level to further increase density. The compact mode setting is saved to the user context settings and persist across user sessions.

Sankey diagram report type

A Sankey diagram provides a clear and intuitive way to visualize how resources such as IT budgets, capacity, or data flow across systems, projects, or processes. By representing flow magnitude through varying connection widths, it enables stakeholders to immediately understand how resources are allocated, uncover dependencies between elements, and identify inefficiencies or imbalances that may otherwise remain hidden in traditional reports.



The new configurable Sankey diagram report template allows solution designers with access to Alfabet Expand to configure Sankey diagrams based on queries and share them with the user community. Nodes and flow weights are derived from a query, and the report can visualize a single level or multiple levels of nodes. The diagram's appearance can be customized via the report assistant, and when nodes represent objects, optional navigation can be enabled to allow users to drill down directly into the underlying data.

Sankey diagram visualization in dark mode is not fully supported. The diagram will be visible, but with low contrast between background and items.

Share and embed views with secure links

Shareable links make it easy for users to quickly distribute and access views, improving collaboration and transparency. A new **Share Link** option has been added to the three-dot menu of views, enabling users to generate links that can be shared via email. Depending on the server alias security configuration, these shareable links can be accessed either by anonymous visitors or restricted to named users.

In addition, embeddable links can be created for use in iframes within third-party components or external websites, allowing views to be seamlessly integrated into other contexts. These embedded views always display up-to-date information retrieved at runtime, while preserving the filter settings applied at the time the link was created. The view is presented in read-only mode without masthead or navigation, and access from external sites is limited to explicitly allowed URLs defined in the alfasettings.json security configuration.

For Alfabet Cloud customers: This feature is available on request. On-premise customers need to activate the feature in the server alias settings of the Alfabet Web Application via the new **Allow Shareable Links** checkbox in the **Server Settings** > **Security** tab of the server alias editor.

Display of CUI texts

Customers who are required to label content as CUI (Controlled Unclassified Information) or display similar security notices can configure user profiles to show persistent messages in the user interface. These messages can be positioned as a header or footer, appearing above or below the Alfabet user interface to ensure visibility.

The header or footer is displayed automatically when a text is defined in the user profile settings and the user logs in with that user profile, ensuring consistent and prominent communication of security requirements.

Migration to Microsoft .NET 10

Microsoft® support for .NET Framework 8 ends on November 10, 2026. Alfabet 11.12.0 has therefore been migrated to .NET Framework 10. This change also enhances overall system performance.

All embedded third-party components were updated to the latest versions supporting .NET Framework 10. The list of of third-party components provided under https://bizzdesign.com/policy/licenses has been updated accordingly.

The following changes apply to the installation of Alfabet 11.12.0 compared to previous releases:

- .NET Framework 10.0 and ASP .NET Core 10.0 must be installed on the server host prior to installation of the Alfabet Server and Alfabet Web Application. If the installer is started on a host without the required components, a prompt will be shown which provides you with a download link for one of the required components. After installation, restart of the installer will show a prompt for installation of the second component.
- To align with ASP.NET Core's built-in support for environment variables, the `alfa-web.config` configuration file containing variable definitions has been added to the configuration infrastructure of the Alfabet Web Application. The configuration file is mandatory and has been added to all relevant sub-folders and the **`Examples`** folder. The file must be copied to the **`config`** folder.

Improvements

The following enhancements to the Alfabet platform are available with this release.

Usability enhancements for notifications

The notification pane has been updated with a refreshed look and feel, enhancing the design and overall user experience through redesigned tabs within the notification pane. A new **Mentions** tab makes it easy to stay on top of conversations by highlighting notifications whenever users are invited via @mentions through the **User Comments** option in the three-dots menu. Additionally, tabs for **Data Capture**, **ADIF**, and **Job Schedule** notifications are now displayed only for users logged in with an administrative profile, ensuring a more relevant and tailored experience.

The visualization and behavior of the notification bell have also been improved to provide clearer, more intuitive feedback. Users are alerted to new notifications directly in the masthead, with a new blue indicator dot appearing on the notification bell when unread items are available. Once a notification is marked as read or the user navigates to the related object (for example, via an @mention), the blue indicator dot automatically disappears, keeping the experience clean and up-to-date.

The Notification Center, where users can access the full notification history, has also been aligned for a more consistent user experience.

Content pages with related object information

Content areas that display information about an object can now be configured to also include data about related objects. This enhancement allows, for example, data from the base object of a workflow step to be shown directly in a separate content page in the workflow step's content area. Users can edit the base object through inline editing, eliminating the need to switch between different content areas and improving efficiency.

Solution designers with access to Alfabet Expand can change the base object for fetching the information on the level of the content page or individual content items. The new base object is defined via a query that finds a single object in the Alfabet database.

Improved layout of data

Field alignment within attribute groups has been reworked to improve clarity and consistency. All fields are now left-aligned, ensuring that values are displayed in clean, consistent columns within the attribute group, rather than appearing differently distributed across rows.

In addition, diagram items that display multiple indicator icons now include padding between the icons. This improves the visual appearance of the diagram and makes the displayed information easier to read and interpret.

Streamlined Excel export

For exporting data to Microsoft Excel®, new formats **XSL (No Icons)** and **XSLX (No Icons)** are now available. These options allow users to export data without indicator icons and without a legend, providing a cleaner and simplified output. This feature is a resolution for ticket ASD-7841.

Multi-language support for Alfabet AI Assistant

The AI assistant for answering questions about Alfabet based on the existing documentation has been enhanced to understand and respond in the language currently used in the user interface.

Calculated properties in editors

Calculated properties can be added to object inspection windows, runtime editors, and custom editors in the same way as object class properties. These properties are intended for informational purposes and are always displayed as read-only, static values.

If a user updates an underlying value that affects the calculation, the calculated property will not automatically refresh within the current view. Additionally, there is a limitation for calculated object class properties that return references or arrays of references, and these values will not be displayed in runtime editors. This improvement is a resolution for ticket ASD-10210.

Generic operation to copy ICT objects

A new standard generic operation has been added to the ICT object class. When an existing ICT object is selected, this operation enables the creation of a new ICT object based on the selected one, streamlining reuse and ensuring consistency. This improvement is a resolution for ticket ASD-9908.

Automatic recalculation of indicators

Solution designers with access to Alfabet Expand can configure content items that display indicator cards, evaluation cards, or object class property groups containing calculated indicators to automatically recalculate indicators without requiring user interaction.

To enable this functionality, the new **Recalculate Indicators on Load** attribute must be set to **True**. This ensures that indicators are recalculated when the base object changes or when the content area or guided data view is reloaded. Note that

this setting only triggers the recalculation itself. To display the updated values, the **Load on Update** property of the content item must also be set to **True**. Recalculation is always performed before the content item is refreshed.

Base object for lookahead queries

The Alfabet parameter `@BASE` can be used in lookahead queries defined for custom selectors. With this change, lookahead queries can refer to the base object of the current view to filter out options which a user shall not select based on the relation of the selector objects with the current base object. This improvement is a resolution for ticket ASD-9846.

Streamlined external source configuration

The dedicated integration with Amazon Web Services® is no longer supported. All features can be implemented with the generic OAS-based interface with third-party RESTful services. This approach provides greater flexibility and can be easily adapted to changes in external APIs without requiring product-specific updates or waiting for reimplementation of dedicated connectors.

As a result, the XML object `AmazonWebServicesConfig` is obsolete and has been removed from the **XML Objects** folder in Alfabet Expand.

Easier file upload via Alfabet RESTful services

The `idocupload` endpoint of the Alfabet RESTful services has been enhanced to simplify file uploads. The JSON specification for the upload can now be provided directly in a request header named `IDOC-Info`.

For backward compatibility, it remains possible to send the JSON specification as a separate file alongside the file being uploaded, as supported in previous releases.

Logging during start-up

The logging implementation using NLog has been enhanced to support logging during the startup phase of Alfabet components. This improvement ensures that initialization processes are fully traceable, enabling better diagnostics and troubleshooting from the very beginning of execution. As a result, updates to the NLog configuration files are required. The configuration examples provided with the release have been revised accordingly to reflect the new setup.

Failover scenario with multiple running Alfabet Servers

High availability for service execution has been improved by introducing failover connection management for the Alfabet Server. It is no longer necessary to rely on a single Alfabet Server instance to handle tasks such as email delivery or event execution for a given database.

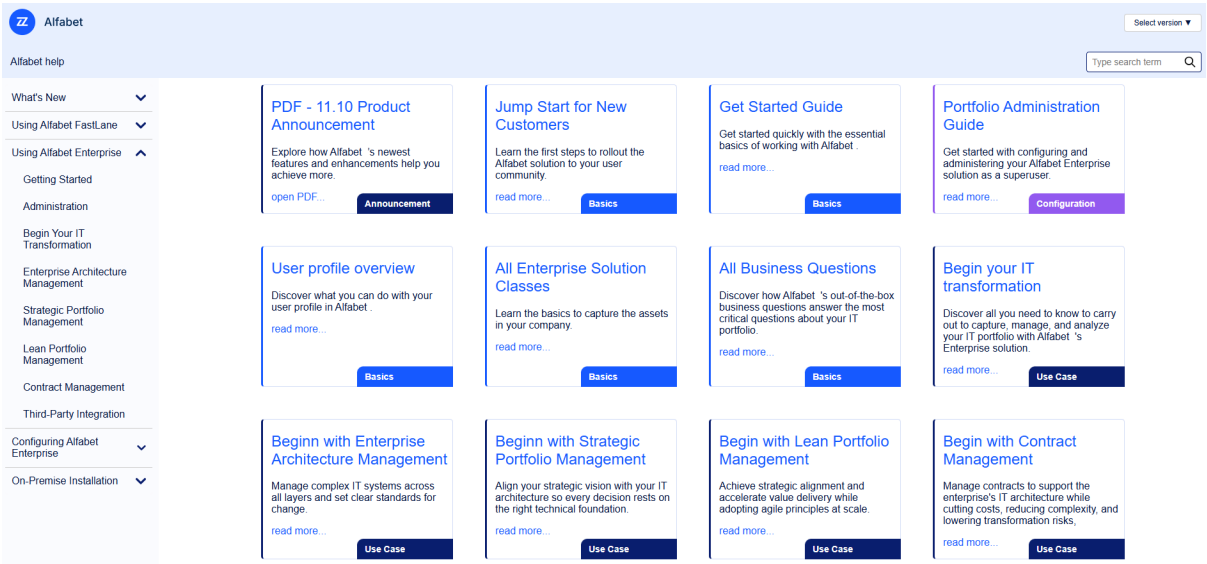
Multiple Alfabet Server instances (running as services or applications) can now connect to the same database. However, only the first connected Alfabet Server instance remains active and executes services, while all others stay in standby mode. If the active Alfabet Server becomes unavailable or is shut down, one of the standby instances automatically takes over. This ensures uninterrupted and reliable execution of services and events.

Archiving of databases with anonymous data via command line

Databases can now be archived via a command line tool to a file while optionally anonymizing sensitive data, without affecting the original source database. A new command-line parameter has been added to the database archive command of AlfaAdministratorConsole.exe, enabling the creation of archive files in ADBZ format. When the parameter `-AnonymizeData true` is specified, the data in the generated ADBZ file is anonymized. By default, this setting is set to `false`, meaning data is archived in its original form unless anonymization is explicitly requested.

New Help Start Page

The Alfabet online help start page has been redesigned to improve usability and reduce the need for scrolling. A new left-hand table of contents panel enables quicker navigation to relevant information. Users can click a link to view related topics in a tile layout for easier browsing. The layout is fully responsive, ensuring optimal readability on tablets and mobile devices. The start page is also Section 508 compliant and supports full keyboard navigation.



Fixes

- ICT objects and business functions have been added to the range of object classes that can be added as object to a service item. They were available for service items in Alfabet 10 releases and were missing in previous Alfabet 11 releases. This fix is a resolution for ticket ASD-10455.
- If a user set a global filter in a content area, the items in the content area were not updated according to the filter setting unless the user clicked the browser refresh button. This has been corrected and visible content will be updated with the global filter settings. This fix is a resolution for ticket ASD-9817.
- Empty values were not filtered if a data workbench column filter was set to return all values for an object class property based on an enumeration. The filter has been changed to explicitly contain an option for empty values.
- Export of content areas caused the user interface to become unresponsive, preventing the export process from completing.
- Favorites for workflow steps could not be opened.
- Workflows configured to send an email when the workflow was finished terminated with an error message and the workflow was never marked as finished. The fix of this issue is a resolution for ticket ASD-10310.
- The **Use Recipient's User Profile for External Links** setting in the server alias of the Alfabet Web Application was not honored when the user opened the user interface from a link in an email sent in the context of a workflow. The user profile opened with the sender's user profile and an error was displayed if the recipient did not have the user profile assigned. The fix of this issue is a resolution for ticket ASD-9767.
- If a user selects objects in a data workbench or tabular view and then changes the filter settings, any selected objects that are no longer visible will be automatically deselected. In a hierarchical data workbench, if a user detaches a selected object from its parent, the selection is not transferred to the next object in the list. Any selection will be cleared.
- If a user updates data in either the **Basic Data** or **Object Details** attribute groups in a content area or guided data view, and the updated data is part of the breadcrumb information, the object data in the breadcrumb will reflect the changed data.

- The storage of workflow step status and comments was incorrect if saving of recent objects was activated in the server alias of the Alfabet Web Application. This issue has been resolved and workflow activities are now saved correctly.
- Changes performed via button interactions on workflow steps did not immediately take effect because the button interaction did not update the content area. The fix of this issue is a resolution for ticket ASD-10416.
- Users could not switch between the **Analysis View** and **Data Entry View** in the data workbench if the object was locked and the user had no edit permissions. Users were thus not able to change to the the **Analysis View** if the **Data Entry View** was used as the first entry point to the object. The switch between both views has also been enabled for locked objects. Users can access both views but will not have edit permissions in any of the views.
- Strings in **Categorized Visual Items** were not consistently translated when the UI was rendered in a language other than English.
- Category headings in the drop-down list of results for the global search were not translated when the UI was rendered in a language other than English. The fix of this issue is a resolution for ticket ASD-10707.
- The user permissions for deleting content pages from content areas were ignored when the content area was opened from an explorer and the user had edited the object. The fix of this issue is a resolution for ticket ASD-10622.
- An error occurred when creating a data capture template for the class stereotype **AI Feature** if the property **Name** was included in the data capture template.
- If data was updated from a data capture template and extended attributes were set only for a subset of updated objects, the empty values were filled with values set for other imported objects. The fix of this issue is a resolution for ticket ASD-10351.
- If the **Enable for Data Capture Templates** attribute was not specified for a class setting, the system did not fall back to the **Enable for Data Capture Templates** setting in the class model. This has been fixed and the **Enable for Data Capture Templates** attribute specified for the object class property will be used if the **Enable for Data Capture Templates** attribute is empty for the corresponding property on the class setting.
- An error occurred when a user tried to equally distribute benefits in the **Benefits Tracking** view for projects. The fix of this issue is a resolution for ticket ASD-10490.

- When the width of a standard graphic view or configured report was modified, it was not possible to revert to the default layout. A new **Reset to Default** option is now available in the three-dot menu for standard graphic views, configured reports, and business functions that embed graphic views. This option restores the view to its original default settings. When selected, a confirmation message is displayed indicating that filters and column structures will be reset to their original configuration.
- When a user resized a wizard window and then changed to another wizard step, the wizard returned to the default size.
- The 50-object limit in data workbench visualizations was implemented even if the graphics showed aggregated values with only a small number of visual elements. This has been corrected and the number of objects displayed will only be reduced if the visualization would result in an exceedingly high number of graphic elements.
- If the database contained indicators without assigned values for an object (for example, due to data imports via data capture templates), setting the indicator in the data workbench resulted in a duplicate entry being created. The fix of this issue is a resolution for ticket ASD-10884.
- In relation selectors that allow linking a base object to different target object classes, changing the target class removed the base object context. The fix of this issue is a resolution for ticket ASD-10956.
- If a selector other than the standard user selector was implemented via the class settings for the object class Person, the **Users Administration** (`ADMIN_UsersOverview`) view could not be opened. The fix of this issue is a resolution for ticket ASD-10105.
- The **Service Product Items** view did not show any objects when rendered in a language other than the default language. The fix of this issue is a resolution for ticket ASD-10454.
- A check for unallowed special characters was missing for user profile and object class property group definitions. The resulting names led to rendering issues and error messages. New objects are now scanned for special characters like '&', and new objects cannot be created if the name contains a special character. Existing objects are not checked. Customers are advised to check their configuration for no-permissible characters in technical names of user profiles and object class property groups. On migration to Alfabet release 11.12, customers can check whether issues with duplicate user profile names exist in their database. User profile name duplications will be listed in the Microsoft

Excel® log file created during update of the meta-model. This fix is a resolution for ticket ASD-10553.

- If the **Responsibilities** view was embedded in a wizard, closing the error message about incorrect role setting erroneously closed the entire wizard. The fix of this issue is a resolution for ticket ASD-10568.
- The **Reports Administration** view could not be opened when filters had been set. The fix of this issue is a resolution for ticket ASD-9669.
- The **Role Types** view of the **Class Configuration** explorer included a non-functional **Allow Duplicate Roles** setting. This setting has no effect and was removed from the view.
- When an error occurred during authentication via an IDP service provider, the error message was misleading. It informed the user about a wrong configuration instead of an issue with the IDP service provider. The content of the error message has changed. It is not possible for the system to distinguish at the time the error occurs whether the error is caused by a false configuration or a general system issue. Therefore, the error message now lists both as potential reasons for the failure of authentication. This fix is a resolution for ticket ASD-8836.
- Alfabet sessions were terminated although the user was active. The fix of this issue is a resolution for ticket ASD-8459.
- Performance issues caused by recalculating all data quality rules affecting a view's base object have been resolved. Recalculation is now limited to data quality rules where the view the user is currently working with is the resolution view defined in the data quality rule.
- The **Data Quality Score** (`DataQualityScore`) and **Data Quality Severity** (`DataQualityMaxSeverity`) object class properties which are available for all relevant object classes were also added to the **Data Quality Violation** (`ObjectDataQualityFinding`) object class. This object class is a technical object class for storing the findings resulting from data quality violations. It is not subject to data quality checks and the two object class properties for storing data quality violations have therefore been removed.
- Performance issues with Gantt charts showing milestones have been resolved.
- The display of custom editors in wizards has been fixed to honor the tab index and tab order in the custom editor definition. This fix is a resolution for ticket ASD-9996.

- For some object classes, configuration of class settings to use runtime editors for an object class resulted in errors when the user tried to open an object of the object class. The fix of this issue is a resolution for tickets ASD-10706 and ASD-10955.
- Setting a visibility condition on a group box caused the custom editor tab to load endlessly. The fix of this issue is a resolution for ticket ASD-10441.
- List box controls were never supported for data entry in custom editors or custom filters. The list box control icon has therefore been removed from the editor for customer editor or configured report filter design.
- A range query has been added to the `CostType` object class property of the `SkillRequest` object class. At the same time, the accessibility of the object class property was set from **Protected** to **Private** and cannot be changed by customers. The accessibility of the `Auto` object class property for the `CostType` object class which specifies whether the cost type is to be available for skill requests in the context of project management has been changed from **Private** to **Protected**. Customers can change this object class property to add or remove cost types without changing the range query for the skill request.
- Tags set in Alfabet Expand for object class property groups were ignored. This has been corrected and object class property groups can be included in AMM files based on tagging. This fix is a resolution for ticket ASD-10641.
- Diagronal label orientation in configured bar chart reports did not functioning correctly.
- ADIF import from CSV files has been fixed to honor the setting of the **Import Column Binding** attribute of the import entry. If the **Import Column Binding** attribute is set to `ByName` columns are not imported in the order specified in the import file but according to the column name in the attribute definitions for the ADIF entry.
- IDOC downloads via ADIF were not logged.
- If the number of requests to a third-party API via the generic API integration interface exceeded 100, only the most recent 100 requests were processed. Responses from earlier requests were ignored, leading to incomplete data handling. This issue has been resolved. The system now correctly processes and stores all API requests, regardless of volume. For example, payload reports containing large numbers of calls (e.g., 2,000 or more) will now have all corresponding data accurately persisted in the temporary tables.

- Update of the meta-model including role type configurations merged the role types to the target database, but completely overwrote the class configuration for role types instead of merging changes to the existing class configuration. The fix of this issue is a resolution for ticket ASD-9158.

Forthcoming changes

While it is not possible to manually create two user profiles with the same name, user profile name validation was missing for some mechanisms importing user profile data. A key has been introduced in the database to further protect user profile data integrity. With Alfabet release 11.13, user profiles with the same name will be forcefully removed from the database. On migration to Alfabet release 11.12, customers can check whether issues with duplicate user profile names exist in their database. User profile name duplications will be listed in the Microsoft Excel® log file created during update of the meta-model. It is recommended that these names are corrected before migrating to Alfabet release 11.13.