

Clarif FileMaker Pro 26.0.1 Release Notes

June 2026

This software updates FileMaker Pro to version 26 (26.0.1) and addresses issues. If you are unable to successfully update to FileMaker Pro 26, search the Knowledge Base for installation information.

New features and enhancements

Version number

In FileMaker 2026, the major version number has been advanced to 26 to correspond with the year. The app icon has also been updated to include the same number.



Artificial intelligence

New AI script steps:

- [Insert Image Caption](#): Sends an image to an image captioning model and inserts the returned caption into a field or variable.
- [Insert Image Captions in Found Set](#): For every record in the found set, sends an image from a source field to an image captioning model and inserts the returned caption into a target field.

AI script step enhancements:

- The [Generate Response from Model](#) script step now includes the **Include tool calls and tool results** option to include more information in the saved message history.
- The [Insert Embedding](#) and [Perform Semantic Find](#) script steps now include a **Parameters** option, allowing provider-specific parameters such as dimension to be passed to the AI model. The [Insert Embedding in Found Set](#) script step's **Parameters** option adds support for provider-specific parameters to its existing support for FileMaker-specific parameters.
- The [Insert Embedding](#) and [Insert Embedding in Found Set](#) script steps and the [GetEmbedding](#) function now support image embedding with the model provider Cohere. See [Clarif FileMaker 2026 Technical Specifications](#).
- In the **Parameters** option for these script steps, you can now set a value for the CURLOPT_TIMEOUT key, which sets the maximum time (in seconds) for the AI model request to complete: [Generate Response from Model](#), [Perform SQL Query by Natural Language](#), [Perform Find by Natural Language](#), and [Perform RAG Action](#).
- The [Perform RAG Action](#) script step now:
 - Returns the document ID assigned to a text or PDF document when it is added to a RAG space. You can store this ID for use when removing the document later, without needing to parse the results of the GetRAGSpaceInfo function to locate it.
 - Supports optional per-request values for Similarity Threshold and Number of Top-Ranked Results in the **Parameters** option, allowing these values to be set dynamically for individual requests rather than relying solely on the global server settings.

- Lets you configure the chunk size used for each document when adding content to a RAG space using the new **Tokens per Text Chunk** option. This allows you to set an appropriate chunk size based on content type or language, rather than relying on a single fixed value.

For more RAG enhancements in Claris AI Model Server, see [FileMaker Server Release Notes](#).

AI enhancements:

- Google Gemini is now supported as an AI model provider for text generation and embedding operations by specifying an account in the [Configure AI Account](#) script step. For recommended models, see [Claris FileMaker 2026 Technical Specifications](#).
- The new *field annotation* lets you describe a field specifically for an AI model, without interfering with your existing field comments. This description is sent to an AI model when you use script steps like [Perform SQL Query by Natural Language](#) and [Perform Find by Natural Language](#). To set a field annotation, in the Advanced Options for Field dialog, enter a description for **Add annotation in Data Definition Language (DDL)**. See [Defining advanced field options](#).

Script steps

New script steps:

- [Flush Web Viewer Cookies](#): Immediately flushes all web viewer cookies used in the FileMaker client. This allows developers to ensure that session information from one user isn't accessible to a subsequent user on the same device.
- [Configure Persistent Data](#): Sets or deletes an entry in the persistent data store.
- The new PDF script steps let you build workflows that combine data and pages from multiple sources into PDF documents:
 - [Create PDF](#): Creates an empty PDF file in memory. Add pages using script steps such as [Append PDF](#), then close and save the file using the [Close PDF](#) script step.
 - [Open PDF](#): Opens a PDF file. Add pages using script steps such as [Append PDF](#), then close and save the file using the [Close PDF](#) script step.
 - [Append PDF](#): Appends PDF pages to the file opened by the [Open PDF](#) script step or created by the [Create PDF](#) script step.
 - [Close PDF](#): Closes and saves the open PDF file to a file path, variable, or container field.
 - [Cancel PDF](#): Closes the open PDF in memory without saving it.
 - [Print PDF](#): Prints a PDF file from the specified file path, container field, or variable.

These script steps appear in the "PDF Files" category in the Script Workspace.

Script step enhancements:

- The [Save Records as PDF](#) script step now includes the **Save to** option, which lets you save the PDF to a file path, container field, variable, or append it to the currently open PDF file. This script step now appears in the "PDF Files" category in the Script Workspace.
- The [Export Field Contents](#) script step is now supported in scripts run by FileMaker Server, FileMaker Data API, and the OData API.
- macOS: The [Get Folder Path](#) script step now uses the value in the **Dialog title** option when displaying the folder selection dialog.
- The [Insert from Device](#) script step's barcode and camera options now include a **Flash** setting with On, Off, and Auto values when the back camera is selected, allowing you to control the device flash for use in low-light environments.
- For the [Insert from URL](#) script step:

- The --proxy-negotiate cURL option now enables Negotiate (SPNEGO) proxy authentication for environments that require it.
- The --proxy-ntlm cURL option now enables NTLM proxy authentication for environments that require it when connecting to external services.
- When retrieving a response with a MIME type of application/json and storing the result in a variable, the JSON is now automatically parsed and cached, improving the performance of subsequent JSON operations on that variable.

See [Insert from URL](#).

- The [Re-Login](#) script step now includes an option to specify which external data source to re-authenticate with, making it possible to re-trigger the sign-in dialog for a related file without closing and reopening all files in a multifile custom app.
- The [Save a Copy as XML](#) script step includes the same new options as in the Save a Copy as XML dialog:
 - **Save each layout object's binary data under its node** stores every layout object's binary data (for example, an image) under the layout object's node in the XML file.
 - **Specify options as JSON** lets you specify most of the options for this script step as a JSON object, making it easier to configure this script step with calculations. You can specify which catalogs to include, whether to include details for analysis tools, whether to create a separate XML file for each catalog, and whether to save object binary data under the object's node.
- The [Select Window](#), [Close Window](#), [Move/Resize Window](#), and [Set Window Title](#) script steps now support selecting a window by UUID, making it possible to uniquely identify a window even if multiple windows have the same name or if the window's name has changed. You can use the new `Get(WindowUUID)` function to get the UUID of the currently active window.
- The [Set Zoom Level](#) script step now has a **Custom** option, allowing you to specify any zoom percentage within the available range rather than being limited to predefined levels.
- The [Show Custom Dialog](#) script step now:
 - Lets you specify the dialog's size and position.
 - Displays a scrollbar when the message content exceeds the visible area of the dialog.
- When using the [Import Records](#) or [Execute SQL](#) script steps, ODBC data source credentials can now be specified by a calculation.

Functions

New functions:

- **FieldAnnotation**: Returns the specified field's annotation used in Data Definition Language (DDL). This is the value set for the field in the Advanced Options for Field dialog.
- **BaseTableComment**: Returns the specified base table's comment as set in the Manage Database dialog.
- **FieldDisplayNames**: Returns a field's display names as JSON data. This is the value set for the field in the Advanced Options for Field dialog. If you set custom key-value pairs in that dialog, use this function to read them.
- **Get(GuidedAccessState)**: In FileMaker Go, returns 1 (true) if Guided Access is turned on; otherwise, returns 0 (false). If you set up FileMaker Go as the single app that has your file open—for example, to run as a kiosk—use this function to detect that state.
- **GetPersistentData**: Returns the persistent data value specified by name and instance ID. These values are stored by the Configure Persistent Data script step in the persistent data store.

- [ListPersistentDataIDs](#): Returns a list of instance IDs for the entries with the specified name in the persistent data store. Use this function to discover what instances exist for a given name.
- [Get\(AccountPasswordDaysRemaining\)](#): Returns the number of days remaining before the current account's password must be changed.

Function enhancements:

- The [GetFieldsOnLayout](#) function now returns the new field annotations defined in the Advanced Options for Field dialog or the existing [LLM] tag in field comments. Field annotations allow you to provide field descriptions to AI models without modifying field comments.
- The [GetRecordIDsFromFoundSet](#) function now accepts an optional parameter that specifies a table occurrence or portal object name, returning the record IDs from the related record set or filtered portal rather than the current found set.
- macOS: The [GetTextFromPDF](#) function now supports scanned or image-based PDF files, extracting text that isn't embedded as selectable text in the PDF.
- In FileMaker Server, the [Get\(PersistentID\)](#) function now returns a stable identifier that persists across server process restarts, reboots, and upgrades. The identifier is derived from an OS-level machine identifier at install time and stored in the server's configuration file, so it remains consistent as long as FileMaker Server is reinstalled over the existing installation.

Persistent data store

- The new *persistent data store* is a set of named values saved as part of the schema in a FileMaker Pro file, not as record data. You can create entries to contain any data your scripts and calculations need to use from any context in the current file. For example, the version number of your custom app, JavaScript libraries for web viewers, fixed prompts to send to AI models, configuration for each instance of an add-on, or any fixed data to share among scripts. See [About the persistent data store](#).

Fields and field options

- Container fields now support WebP images. You can insert, display, and work with WebP image files in container fields in all FileMaker clients without needing to convert or rename the files.
- You can now customize *field display names* in certain dialogs so they make sense to users without having to change your underlying field names. In the Advanced Options for Field dialog, for **Customize field display names**, you can set display names to use in the Specify Field Order for Export dialog and in exported Excel files, in the Sort Records dialog, and in the column header in Table View. See [Defining advanced field options](#).
- When saved as part of an add-on and installed in another file, the state of a field's auto-enter lookup option **Don't copy contents if empty** is now preserved.
- You can now control whether default fields are added to newly defined tables. In the File Options dialog, in the new Schema tab, **Add default fields to newly defined tables** specifies the option for the current file. In the Settings dialog, in the Schema tab (previously named Layout), the **Add default fields to newly defined tables in new files** option controls whether the setting in File Options is selected when you create a file. See [Setting file options](#) and [Changing schema settings](#).
- In the Manage Database dialog, when the field list is set to display Advanced in the third column, double-clicking a field opens the new Advanced Options for Field dialog directly.

Script Workspace

- In the Script Workspace, consecutive comments and disabled script steps can now be collapsed and expanded, making it easier to navigate long scripts.

Layouts

- macOS: The new Inspector pane in Layout mode adapts to show you only the settings that apply to the selected object, reducing the amount of information displayed at once. It also now has a simpler two-tab interface—Appearance and Data—that consolidates the previous four tabs (Position, Style, Appearance, and Data). The existing Inspector window is still available. See [Using the Inspector to format objects](#).
- You can now control entry into field objects by calculation rather than by a fixed setting. Your calculation should return 1 to allow full editing, 2 to allow selection and copying but not editing (read-only), or any other value to prevent entry and allow viewing only. See [Allowing or preventing entry into fields](#).

SQL

- Double-quoted system column names "ROWID" and "ROWMODID" are now supported in FileMaker SQL queries, consistent with how other SQL systems handle quoted system column identifiers. ROWID and ROWMODID are also now available as named constants, equivalent to Get(RecordID) and Get(RecordModificationCount), to use in calculations. See [Named constants and other special keywords](#).
- FileMaker SQL now supports FOREIGN KEY syntax in CREATE TABLE and ALTER TABLE statements, allowing tables with relationships to be created or modified using SQL. See [CREATE TABLE statement](#) and [ALTER TABLE statement](#) in FileMaker SQL Reference.

Save a Copy as XML

- The new Save a Copy as XML dialog now lets you save multiple FileMaker Pro files as XML at once, select individual catalogs to include, create a separate XML file for each catalog, store every layout object's binary data under the object's node, and automatically open a summary XML file that lists all generated output. The **Include details for analysis tools** option from the previous dialog is also available. See [Documenting database schemas](#).
- When a FileMaker Pro file is saved using Save a Copy as XML:
 - Table View column information is now included in the XML output, including each column's field reference, visibility, and width.
 - The XML is now encoded in UTF-8 instead of UTF-16 LE.
 - The View By sort order setting is now included for both tables and fields as specified in the Manage Database dialog, including the field order when View By is set to Custom.
 - The Summary.xml file is also generated when multiple FileMaker Pro files are saved at once. The summary includes source file information and creation details, making it possible to identify which source database produced each output XML file when multiple files share the same name.
 - The XML now includes an attribute in the root element indicating whether catalogs were split into separate files, making it possible for analysis tools to determine whether the export represents a complete database definition.
 - In the XML for the Configure Machine Learning Model script step, the Operation value was placed directly inside the ConfigureCoreML element as a text node rather than in its own Operation child element, making it difficult to parse programmatically. The XML structure now uses a proper Operation element.

Add-ons

- Add-ons created with this version of FileMaker Pro now include a creation build ID in their info.json file. When installing add-ons, FileMaker Pro won't display add-ons that were created with a newer version of FileMaker Pro, preventing potential compatibility issues. Add-ons created with older versions of FileMaker Pro continue to work as before. Add-on developers who need to install a newer add-on in an older version of FileMaker Pro can remove the Creation_BuildID attribute from the add-on's info.json file.

Assisted install

For administrators, the Assisted Install.txt file supports these new properties:

- **AI_MAJORUPGRADENOTIFY** lets administrators control whether FileMaker Pro notifies users of major version upgrades via the Software Update dialog. Setting the property to notify of minor updates only prevents users from being prompted to upgrade to a new major version, which is useful when the FileMaker host they use hasn't yet been upgraded.
- **AI_DISABLEPRODUCTANNOUNCEMENTS** disables product announcements in the Home window and hides the **Show product announcements in the Home window** option in the Settings dialog. See [Changing general settings](#).

See [Setting personalization properties in Assisted Install.txt](#) in FileMaker Pro Network Install Setup Guide.

General

- The **Manage database, data sources, containers, and custom functions** privilege now allows accounts with this privilege to use the Data Viewer while debugging scripts for which they have the **Modifiable** privilege. See [Editing other privileges](#).
- The Manage Custom Functions dialog now lets you sort functions by name or by creation order even when functions are organized into folders.
- The Replace Field Contents dialog now includes a **Perform auto-enter options for fields** option when the current account uses the Full Access privilege set. Deselecting this option lets you perform replacements without updating other fields with auto-enter options.
- The Sort Records dialog now includes an option to control whether records with blank values in the sort field appear at the top or bottom of the sorted results.
- The Specify Import Order dialog and the Convert File dialog now allow you to select the **Preserve numbers as exact text in text fields** option when importing data, preventing values that resemble scientific notation from being incorrectly interpreted as numbers during import. See [Setting the import action and mapping fields during import](#).
- The dialog that appears when switching between Form, List, and Table View in a hosted file now uses clearer wording to explain that choosing **Yes** changes the default view for all users, and choosing **No** changes only the current user's view. A **Do not show this message again** option is also available—which saves your choice in the Settings dialog. See [Changing schema settings](#).
- In the Settings dialog, you can now turn off product announcements displayed in the Home window. See [Changing general settings](#).
- [FileMaker features not compatible with previous versions](#) has been restored in FileMaker Pro Help.

Updated libraries and packages

The following third-party libraries have been updated to the indicated versions:

- libcurl 8.18.0
- OpenSSL 3.5.5 LTS
- PDF-Writer 4.8.1
- omniORB 4.3.4
- jsoncons 1.5.0
- zlib 1.3.1

Removed features

- FileMaker Pro 2026 is not supported on macOS 14 Sonoma and, in this release, will no longer install on Ventura. If you're a Ventura user, you must upgrade to a later version of macOS before

you can install this version of FileMaker Pro. For the supported versions of macOS, see [Claris FileMaker 2026 Technical Specifications](#).

- The ability to create a FileMaker Pro file directly on your FileMaker Cloud host has been removed. Instead, you can create the file locally, then upload it to the host.

Addressed issues

Artificial intelligence

- FileMaker Pro quit unexpectedly when the GetRAGSpaceInfo function used a calculation for its parameter instead of a string value.
- FileMaker Pro quit unexpectedly when performing the Generate Response from Model script step with both **Stream** option and **Perform JavaScript in Web Viewer** options turned on.
- When performing the Generate Response from Model script step with the **Stream** option turned off, the message history variable wasn't populated after the script step completed.
- The Perform Find by Natural Language script step returned an error when the AI model returned the omit value as a Boolean instead of a string in the find request JSON.
- The Perform Semantic Find script step returned an error about a missing AI account when the **Query by** option was set to **Vector data**, even though no AI account is required for that option.
- The default JSON prompt generated by the Configure Prompt Template script step for the Find Request template contained invalid JSON.

Calculations and functions

- The ExecuteSQL function returned ? if a JOIN in the FROM clause was enclosed in parentheses.
- FileMaker SQL queries using multiple cross joins returned ? on macOS or caused FileMaker Pro to quit unexpectedly in Windows.
- The GetAsNumber function incorrectly returned 0 when applied to a numeric JSON value retrieved using JSONGetElement.
- The GetLiveText function returned inaccurate or incomplete text for some of the non-English languages.
- The GetRecordIDsFromFoundSet function using type0 or 3 returned a number instead of text, causing GetAsNumber equality comparisons to incorrectly return 1.
- The JSON functions converted small decimal numbers between 0 and 1 with four or more leading zeros to scientific notation (for example, returning 1e-05 instead of 0.00001).
- Using the GetLiveTextAsJSON function on a multi-page PDF file produced invalid JSON syntax by returning comma-separated arrays for each page instead of a single valid JSON array.
- Converting Greek uppercase text to lowercase using Format > Style > Lowercase or the Lower() function incorrectly changed the character "Σ" at the end of a word to "σ" instead of "ς".

Layouts

- The Self function didn't work correctly when used for hiding layout objects on merge fields or merge variables.
- In Layout mode, removing a button from a button bar that was part of a grouped object didn't alert the user that objects can't be removed from a grouped object if they're in a button bar that's in the same grouped object.
- In Layout mode, right-clicking a layout calculation object that contained broken field references didn't open the Specify Calculation dialog, preventing users from fixing the broken references.
- In Layout mode, when the Fields tab of the Objects pane and the Drag Preferences area were both visible and the window height was too short, the Drag Preferences area overlapped other controls.
- Switching from Table View to Form View removed the left padding set for checkbox and radio button fields.
- In Layout mode, the Format > Portal Setup menu was dimmed even when a portal was selected.

- When an add-on containing a button bar with icons positioned to the left or right of the label was installed in another file, the icon or label was partially hidden.
- When switching between Table View and another view in Browse mode in a hosted file, FileMaker Pro displayed a misleading dialog that could cause the layout's **Default view** setting to be changed unintentionally.
- After deleting the table occurrence associated with an add-on's table from the relationships graph, the add-on panel on the layout no longer displayed its content.
- A summary field placed inside a layout calculation object in a sub-summary part showed the total for the entire found set instead of the total for the current sub-summary group.
- Windows: In some cases, the ruler didn't display units of measurement.
- macOS Tahoe: In the color palette in the Inspector, colors in the first three rows couldn't be selected by clicking.
- Windows: FileMaker Pro quit unexpectedly when clicking an item in the Objects tab while the Set Tab Order dialog was open and the Objects pane had been hidden and then shown again.

Web viewers

- macOS: JavaScript running in a web viewer that attempted to start a file download using a binary large object (BLOB) URL displayed an error message instead of saving the file to the Downloads folder.
- Windows: Text was duplicated when typing in Find mode immediately after a FileMaker script performed from a web viewer changed the layout.
- Windows: Pressing the Esc (Escape) key within a web viewer incorrectly displayed a FileMaker Pro alert.

Script Workspace

- Windows: Shift and arrow key combinations in the Script Workspace incorrectly deselected previously highlighted lines when modifying a block selection.
- macOS: When multiple databases were open across different macOS spaces and each had a Script Workspace window open, switching spaces moved all open Script Workspace windows to the active space.
- macOS: Script Workspace windows unexpectedly changed desktops when switching between applications.
- When the **Script** option was unselected in the Configure Local Notification script step, the script name was removed but the script parameter value remained.
- macOS: In the options dialog for the Perform JavaScript in Web Viewer script step, the edit icon didn't appear when hovering over a parameter list item, making it difficult to discover how to edit an existing parameter.

Script steps

- If the Insert File script step was used with a file path that contained a null character, FileMaker Pro stopped responding and required a force quit.
- In some cases, the Go to List of Records script step caused FileMaker Pro to quit unexpectedly when the provided list contained an empty string value.
- In some cases, when the Go to List of Records script step was set to go to the current layout, it navigated to a different layout.
- Opening a snapshot link generated after using the Go to List of Records script step failed to retain the found set and sort order.
- Using the Constrain Found Set script step with the **Find without indexes** option incorrectly created an index.
- macOS: When system notifications were disabled for FileMaker Pro, the Configure Local Notification script step displayed an error dialog even when Set Error Capture was turned on.

- When the version key was set in the Execute FileMaker Data API script step, the response was returned using the v1 format instead of the specified version.
- When using the Replace Field Contents script step with the **Perform auto-enter options for fields** option unselected, modification timestamp fields were still updated.
- Windows: The Install Plug-In File script step failed with a "file locked or in use" error when a duplicate copy of the plug-in existed in the plug-ins folder.

Script Debugger and Data Viewer

- Editing and saving a script from another client while it was paused in the Script Debugger led to inconsistent script step execution.
- In the Data Viewer, switching from the Watch tab to the Current tab caused the parsed state of JSON variables to reset to 0.

Save a Copy as XML



When a FileMaker Pro file was saved using Save a Copy as XML:

- Credentials stored in the Import Records script step for ODBC data sources were output as plain text in the XML.
- With the system locale set to a non-English language, function names in the output were localized to the system language instead of being output in English.

General

- FileMaker Pro quit unexpectedly under certain conditions when performing repeated find operations in a database.
- Windows: When upgrading to the next major version, in some cases, the updater didn't ask whether to keep the previous version and removed the older version.
- The Fields tab in the Manage Database window didn't retain the **View By** setting when navigating between tables and after closing and reopening the file.
- FileMaker Pro quit unexpectedly when editing a global field using a pop-up value list while the found set was empty and another client was simultaneously creating records.
- FileMaker Pro stopped responding when it tried to delete records after a script had already deleted all records within a transaction.
- PDF files generated by FileMaker Pro that contained certain rectangle layout objects didn't open correctly in Adobe Acrobat and caused Acrobat to become unresponsive or quit unexpectedly.
- Saving a self-contained copy of a database failed to embed container objects that were stored as file references.
- When a database had no Full Access accounts and the **Manage accounts that don't have Full Access** privilege was configured, attempting to save changes in the Manage Security dialog incorrectly displayed an error.
- An error message appeared when opening a database with an external file reference if the external data source relied on a global variable that had not yet been set by the startup script.
- When opening a hosted file that referenced an external file and the sign-in dialog for the external file was cancelled, the original file was unexpectedly disconnected and closed.
- When saving records as a PDF file, certain characters such as the euro symbol (€) were rendered as squares when using the Helvetica font.
- Windows: In some cases, copied custom menu sets couldn't be pasted in the Manage Custom Menus dialog.
- Windows: FileMaker Pro displayed intermittent error messages about not being able to find or load the dictionary during standard database interactions, including text entry and entering Find mode.

- Windows: In List view, PDF files stored in container fields within a popover stopped displaying when the popover was opened for a different record.
- Windows: The file cache size set using the AI_SET_FILE_CACHE property in Assisted Install.txt was reset to the default value after opening and closing the Settings dialog.
- Windows: When a dialog window was displayed by using the New Window script step, windows from other applications such as Adobe Acrobat, Word, and Excel couldn't be brought to the front on Windows 11.
- macOS: FileMaker Pro quit unexpectedly when changing a text field to a calculation field while the keyboard was set to the US International PC input source.
- macOS: When replacing a license certificate, the License Confirmed dialog incorrectly displayed the expiration date of the previously installed license.

Platform	Product	Updates Version	Download	Languages
Windows	FileMaker Pro 26	26.0.1		English French Arab Hebrew Indian Thai
macOS	FileMaker Pro 26	26.0.1		

APIs, technologies, or features to be deprecated

As FileMaker Pro evolves, the list of supported technologies, APIs, and features will change. As part of this evolution, certain operating systems versions, hardware, and features may be deprecated in favor of newer ones. Although deprecation does not mean the immediate deletion of an item, you should migrate your solution away from deprecated technologies, because these technologies may be removed in a future version of the product.

For the latest information about deprecated APIs, technologies, and features, search the Knowledge Base.