# Archer Scripts Date Validator

The Archer Scripts Date Validator tool & utility is a custom object designed to streamline common operational functionality such as date validation within the Archer platform.  This offering helps Archer Users ensure date fields do not exceed a year (or any number of days) from the current date, start dates are never later than end dates, and a suite of other configurable date comparison and validation options. The offering is both environment and field ID agnostic, ensuring a seamless integration experience.

On this page

* [Release history](#Releasehistory)
* [Overview of Archer Scripts Date Validator](#OverviewofArcherScriptsDateValidator)
  + [Key features and benefits](#Keyfeaturesandbenefits)
  + [Prerequisites (ODA and system requirements)](#PrerequisitesODAandsystemrequirements)
  + [Additional resources](#Additionalresources)
* [Archer Scripts Date Validator Tool & Utility components](#X142c494c670faecb5c73df6cf0c8ae06fe1030d)
  + [Architecture diagram](#Architecturediagram)
  + [Applications](#Applications)
* [Installing Archer Scripts Date Validator Tool & Utility](#X34cc6c9938c19154930fa9545d0601280bc3d00)
  + [Installation overview](#Installationoverview)
    - [Step 1: Prepare for the installation](#Step1Preparefortheinstallation)
    - [Step 2: Install the package](#Step2Installthepackage)
    - [Step 3: Test the installation](#Step3Testtheinstallation)
  + [Installing the package](#Installingthepackage)
    - [Task 1: Back up your database (optional)](#Task1Backupyourdatabaseoptional)
    - [Task 2: Import the package](#Task2Importthepackage)
    - [Task 3: Map objects in the package](#Task3Mapobjectsinthepackage)
    - [Task 4: Install the package](#Task4Installthepackage)
    - [Task 5: Review the package installation log](#Task5Reviewthepackageinstallationlog)
* [Configuring the Archer Scripts Date Validator Tool & Utility](#X4e98d03076cd1b2131d4a35a39d1ae57ae16323)
  + [Task 1: Configure the fields to be compared](#Task1Configurethefieldstobecompared)
  + [Task 2: Understanding more scenarios](#Task2Understandingmorescenarios)
* [Using Archer Scripts Date Validator Tool & Utility](#Xd9259446885d20f1cee3ae68f798a2318b491d3)
  + [Task 1: Test script in the Archer Scripts UI Enhancement Toolkit ODA from the package](#Xc5ddea369f1e5f3169db3e912284ffdea5e3211)
  + [Task 2: Using the script in your own applications](#Task2Usingthescriptinyourownapplications)
* [Certification environment](#Certificationenvironment)

## Release history

Last updated: May 2024

## Overview of Archer Scripts Date Validator

### Key features and benefits

The Archer Scripts Date Validator tool & utility will enable organizations to:

* Compare dates to ensure validation logics (i.e. start date cannot be later than end date)
* Notify user and remove invalid entries
* Ensure date validity (i.e. date does not exceed a specified number of days from the current date)
* Define logic through the Archer Administrator
* Configure warning messages

### Prerequisites (ODA and system requirements)

| Components | Prerequisites |
| --- | --- |
| Archer Solution Area(s) | N/A |
| Archer Use Case(s) | Any use cases |
| Archer Applications | N/A |
| Uses Custom Application | Yes.  Script can be applied to any applications |
| Requires On-Demand License | No |
| Archer Requirements | Archer 6.9 and later |
| Partner/Vendor Requirements | Valid License is required, Free of charge |
| Operating System | All |
| Database Server | All |
| Services Server | All |
| Supported Archer Environments | * On-Premises * Archer SaaS |

### Additional resources

The following additional resources are available for this offering:

* Archer Scripts website: <https://www.archerscripts.com/>

## Archer Scripts Date Validator Tool & Utility components

### Architecture diagram

This offering is powered by a custom object, which can be deployed to any application or questionnaire.

### Applications

**Note:** The “Archer Scripts LLC UI Enhancement Toolkit” On-Demand Application (ODA) is ONLY used to host the custom object and provide already configured examples to showcase the functionality. No ODA license is required for this solution.

| Application | Description |
| --- | --- |
| Archer Scripts UI Enhancement Toolkit | This application is used to host the custom object. It has sample fields built which can be used to test the various features provided by the custom object. |

## Installing Archer Scripts Date Validator Tool & Utility

### Installation overview

This offering consists of a custom object, designed for easy integration into existing applications. The Archer Scripts LLC UI Enhancement Toolkit on-demand application is not required to be installed if you wish to apply the custom object directly to your Archer applications.

**Note:** To facilitate a clear understanding of its capabilities, we offer the "Archer Scripts LLC UI Enhancement Toolkit" ODA that serves exclusively as a demonstration tool. This ODA showcases the functionalities through pre-configured examples. Please follow the subsequent steps to install the ODA, which hosts the script for demonstration purposes only, no ODA license is required for this solution.

#### Step 1: Prepare for the installation

1. Ensure that your Archer system meets the following requirements:
   * Archer Platform version 6.9 and later.
2. Obtain the Data Dictionary for the ODA by contacting your Archer Account Representative or calling 1-888-539-EGRC. The Data Dictionary contains the configuration information for the use case.
3. Read and understand the "Packaging Data" section of the Archer Online Documentation.

#### Step 2: Install the package

Installing a package requires that you import the package file, map the objects in the package to objects in the target instance, and then install the package. See Installing the Application Package for complete information.

#### Step 3: Test the installation

Test the application according to your company standards and procedures, to ensure that the use case works with your existing processes.

### Installing the package

#### Task 1: Back up your database (optional)

This step is optional as the ODA is new and is not interacting with any existing use case applications.

There is no Undo function for a package installation. Packaging is a powerful feature that can make significant changes to an instance. Archer strongly recommends backing up the instance database before installing a package. This process enables a full restoration if necessary.

An alternate method for undoing a package installation is to create a package of the affected objects in the target instance before installing the new package. This package provides a snapshot of the instance before the new package is installed, which can be used to help undo the changes made by the package installation. New objects created by the package installation must be manually deleted.

#### Task 2: Import the package

1. Go to the Install Packages page.
   1. From the menu bar, click Admin menu.
   2. Under Application Builder, click Install Packages.
2. In the Available Packages section, click Import.
3. Click Add New, then locate and select the package file that you want to import.
4. Click OK.

The package file is displayed in the Available Packages section and is ready for installation.

#### Task 3: Map objects in the package

**Important:** This step is required only if you are upgrading to a later version of Archer Scripts LLC UI Enhancement Toolkit.

1. In the Available Packages section, select the package you want to map.
2. In the Actions column, click  for that package.

* The analyzer runs and examines the information in the package. The analyzer automatically matches the system IDs of the objects in the package with the objects in the target instances and identifies objects from the package that are successfully mapped to objects in the target instance, objects that are new or exist but are not mapped, and objects that do not exist (the object is in the target but not in the source).
* **Note:** This process can take several minutes or more, especially if the package is large, and may time out after 60 minutes. This time-out setting temporarily overrides any IIS time-out settings set to less than 60 minutes.
* When the analyzer is complete, the Advanced Package Mapping page lists the objects in the package file and corresponding objects in the target instance. The objects are divided into tabs, depending on whether they are found within Applications, Solutions, Access Roles, Groups, Sub-Forms, or Questionnaires.

1. On each tab of the Advanced Mapping Page, review the icons that are displayed next to each object name to determine which objects require you to map them manually.

| Icon | Name | Description |
| --- | --- | --- |
| Awaiting mapping review | Awaiting Mapping Review | Indicates that the system could not automatically match the object or children of the object to a corresponding object in the target instance.  Objects marked with this symbol must be mapped manually through the mapping process.  **Important:** New objects should not be mapped. This icon should remain visible. The mapping process can proceed without mapping all the objects.  **Note:** You can execute the mapping process without mapping all the objects. The Awaiting mapping review icon is for informational purposes only. |
| Checkmark | Mapping  Completed | Indicates that the object and all child objects are mapped to an object in the target instance. Nothing more needs to be done with these objects in Advanced Package Mapping. |
| Missing objects | Do Not  Map | Indicates that the object does not exist in the target instance or the object was not mapped through the Do Not Map option. These objects will not be mapped through Advanced Package Mapping and must be remedied manually. |
|  | Undo | Indicates that a mapped object can be unmapped. This icon is displayed in the Actions column of a mapped object or object flagged as Do Not Map. |

1. For each object that requires remediation, do one of the following:
   * To map each item individually, on the Target column, select the object in the target instance to which you want to map the source object. If an object is new or if you do not want to map an object, select Do Not Map from the drop-down list.
   * **Important:** Ensure that you map all objects to their lowest level. When objects have child or related objects, a drill-down link is provided on the parent object. Child objects must be mapped before parent objects are mapped. For more details, see "Mapping Parent/Child Objects" in the Archer Online Documentation.
   * To automatically map all objects in a tab that have different system IDs but the same object name as an object in the target instance, do the following:
     1. In the toolbar, click Auto Map.
     2. Select an option for mapping objects by name.

| Option | Description |
| --- | --- |
| Ignore case | Select this option to match objects with similar names regardless of the case of the characters in the object names. |
| Ignore spaces | Select this option to match objects with similar names regardless of whether spaces exist in the object names. |

* + 1. Click OK.
    - The Confirmation dialog box opens with the total number of mappings performed. These mappings have not been committed to the database yet and can be modified in the Advanced Package Mapping page.
    1. Click OK.
       - To set all objects in the tab to Do Not Map, in the toolbar, click Do Not Map.
    - **Note:** To undo the mapping settings for any individual object, click  in the Actions column.
    - When all objects are mapped, the Checkmark icon is displayed in the tab title. The Missing objects icon is displayed next to the object to indicate that the object will not be mapped.

1. Verify that all other objects are mapped correctly.
2. (Optional) To save your mapping settings so that you can resume working later, see "Exporting and Importing Mapping Settings" in the Archer Online Documentation.
3. Once you have reviewed and mapped all objects, click .
4. Select I understand the implications of performing this operation and click OK.

The Advanced Package Mapping process updates the system IDs of the objects in the target instance as defined on the Advanced Package Mapping page. When the mapping is complete, the Import and Install Packages page is displayed.

**Important:** Advanced Package Mapping modifies the system IDs in the target instance. Any Data Feeds and Web Service APIs that use these objects will need to be updated with the new system IDs.

#### Task 4: Install the package

All objects from the source instance are installed in the target instance unless the object cannot be found or is flagged to not be installed in the target instance. A list of conditions that may cause objects not to be installed is provided in the Log Messages section. A log entry is displayed in the Package Installation Log section.

1. Go to the Install Packages page.
   1. From the menu bar, click Admin menu.
   2. Under Application Builder, click Install Packages.
2. In the Available Packages section, locate the package file that you want to install, and click Install.
3. In the Configuration section, select the components of the package that you want to install.
   * To select all components, select the top-level checkbox.
   * To install only specific global reports in an already installed application, select the checkbox associated with each report that you want to install.
   * **Note:** Items in the package that do not match an existing item in the target instance are selected by default.
4. In the Configuration section, under Install Method, select an option for each selected component. To use the same Install Method for all selected components, select a method from the top-level drop-down list.

* **Note:** If you have any existing components that you do not want to modify, select Create New Only. You may have to modify those components after installing the package to use the changes made by the package.

1. In the Configuration section, under Install Option, select an option for each selected component. To use the same Install Option for all selected components, select an option from the top-level drop-down list.

* **Note:** If you have any custom fields or formatting in a component that you do not want to lose, select Do not Override Layout. You may have to modify the layout after installing the package to use the changes made by the package.

1. To deactivate target fields and data-driven events that are not in the package, in the Post-Install Actions section, select the Deactivate target fields and data-driven events that are not in the package checkbox.

* To rename the deactivated target fields and data-driven events with a user-defined prefix, select the Apply a prefix to all deactivated objects checkbox, and enter a prefix. This can help you identify any fields or data-driven events that you may want to review for cleanup post-install.

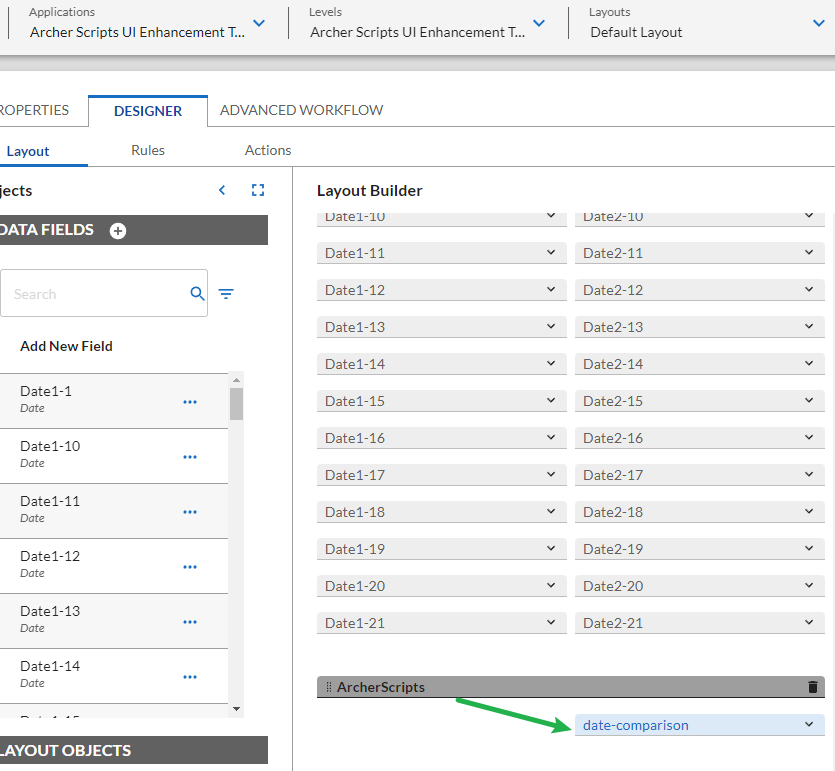
1. Click Install.
2. Click OK.

#### Task 5: Review the package installation log

1. Go to the Package Installation Log tab of the Install Packages page.
   1. From the menu bar, click Admin menu.
   2. Under Application Builder, click Install Packages.
   3. Click the Package Installation Log tab.
2. Click the package that you want to view.
3. In the Package Installation Log page, in the Object Details section, click View All Warnings.

## Configuring the Archer Scripts Date Validator Tool & Utility

### Task 1: Configure the fields to be compared

1. Locate the custom object under DESIGNER.  
   
2. Expand the custom object and review the Code section. The “dateComparison” section is where all the configurations reside. See the definition below:
   * `comparisonFieldName`: name of the field to compare and will be cleared if invalid
   * `baseFieldName`: name of the base field to compare against
   * `logic`: comparison logic
   * `warningMessage`: message to show if the comparison is invalid

* Using the following block as an example to compare two date fields and ensure Date2 must be greater than Date1.  If the value entered in Date2 is less than Date1, it will be cleared and the user will receive a warning message.   A real-world scenario for Date1 is Issue Discovery Date, and Date2 is Issue Resolved Date.
* {
* comparisonFieldName: 'Date2',
* logic: '>',
* baseFieldName: 'Date1',
* warningMessage: 'Invalid entry. Date2 must be later than Date1.',
* },

### Task 2: Understanding more scenarios

1. The following block compares two dates, Date2 has to be greater than or equal to Date1:

* {
* comparisonFieldName: 'Date2-2',
* logic: '>=',
* baseFieldName: 'Date1-2',
* warningMessage: 'Invalid entry. Date2 must be later than or equal to Date1.',
* },

1. The following block ensures Date1 does not equal to Date2:

* {
* comparisonFieldName: 'Date2-6',
* logic: '!=',
* baseFieldName: 'Date1-6',
* warningMessage: 'Invalid entry. Date1 cannot be the same as Date2.',
* },

1. The following block ensures Date2 must not exceed a year from Date1:

* {
* comparisonFieldName: 'Date2-7',
* logic: '<$+1year',
* baseFieldName: 'Date1-7',
* warningMessage: 'Invalid entry. Date2 must not exceed a year from Date1.',
* },

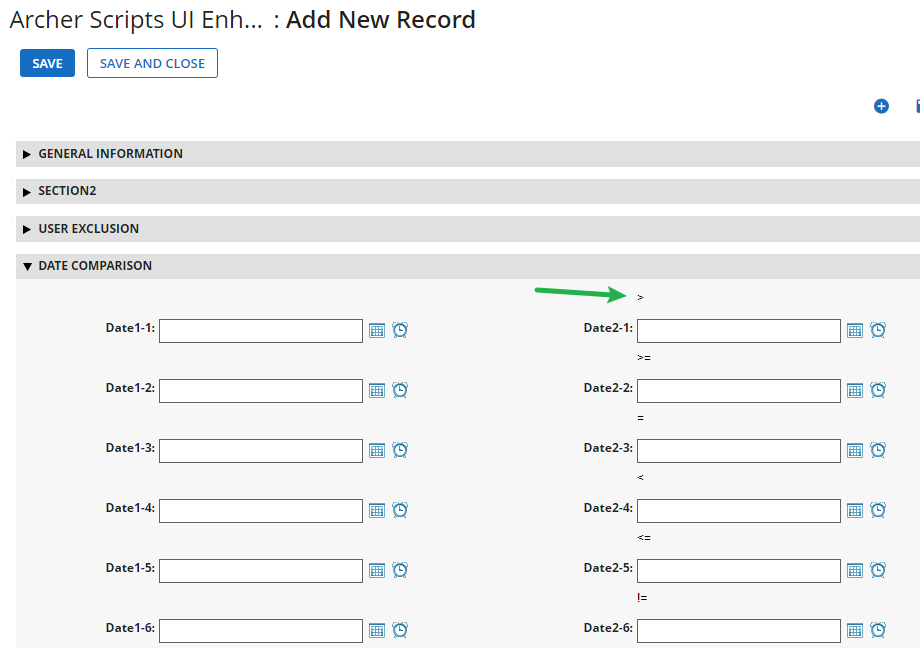
1. More logic is listed in the script. Only copy the scenario that is applicable to your use case for date comparison and validation. Use the ODA to test all the scenarios.

## Using Archer Scripts Date Validator Tool & Utility

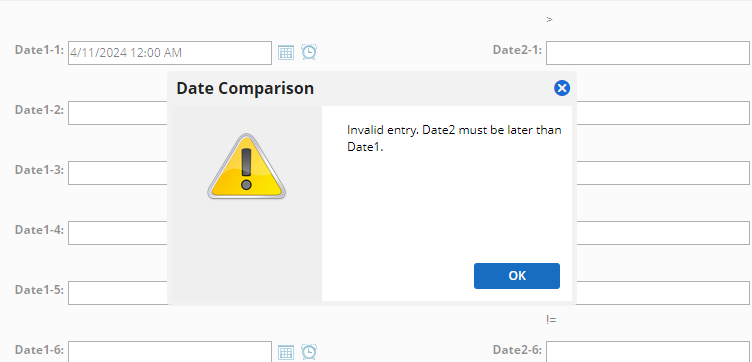
### Task 1: Test script in the Archer Scripts UI Enhancement Toolkit ODA from the package

1. Locate the Archer Scripts UI Enhancement Toolkit ODA.
2. Create a new record.

* **Note:** The logic is displayed as help text ABOVE the field.  See screenshot below, the > sign means Date2-1 field has to be later than Date1-1 field. Enter a date in Date1-1,  then try to enter a date in Date2-1 that’s less than Date1-1, you should see the warning message, and the date entered in Date2-1 will be cleared.



The screenshot below shows the warning message after a date entered in Date2-1 is less than 4/11 which is in Date1-1.



### Task 2: Using the script in your own applications

1. Go to Application Builder and Archer Scripts UI Enhancement Toolkit ODA, locate the custom object and copy the code into your application where you want to deploy the date comparison feature.
2. Support your business use case is that the “Requested Extension Date” cannot exceed a year from the “Initial Due Date”. In this case, you would use the following logic:

* {
* comparisonFieldName: 'Requested Extension Date',
* logic: '<$+1year',
* baseFieldName: 'Initial Due Date',
* warningMessage: 'Invalid entry. Requested Extension Date must not exceed a year from Initial Due Date.',
* },

1. Remove all the other blocks (logic) in the config section. Below is what it looks like after applying only the logic from step 2.  All other sections in the script should be untouched.

* const config = {
* dateComparison: [
* {
* comparisonFieldName: 'Requested Extension Date',
* logic: '<$+1year',
* baseFieldName: 'Initial Due Date',
* warningMessage: 'Invalid entry. Requested Extension Date must not exceed a year from Initial Due Date.',
* },
* ],
* };

1. You can have multiple blocks configured for multiple Date field pairs, similar to the original example. Below is the configuration example for 2 Date comparisons. One from the previous example, and a new logic that ensures Issue Resolved Date is later than Issue Identified Date.

* const config = {
* dateComparison: [
* {
* comparisonFieldName: 'Requested Extension Date',
* logic: '<$+1year',
* baseFieldName: 'Initial Due Date',
* warningMessage: 'Invalid entry. Requested Extension Date must not exceed a year from Initial Due Date.',
* },
* {
* comparisonFieldName: 'Issue Resolved Date',

            logic: '>=',

            baseFieldName: 'Issue Identified Date',

            warningMessage: 'Invalid entry. Issue Resolved Date must be later than or equal to Issue Identified Date.',

},

],

};

## Certification environment

Date tested: April 2024

| Product Name | Version Information | Operating System |
| --- | --- | --- |
| Archer | 6.14 | Windows |