



Assignment Plug-In – User Guide

Contents

Overview.....	3
Using the Assignment Plug-In via Process Builder	4
Immediate Action	4
Scheduled Action	7
Using the Assignment Plug-In via Flow Builder	12
Creating Button to Call Flow Example	15
Using the Assignment Plug-In via Flow (Legacy Flow Builder)	17
FAQ.....	22

Overview

The Assignment Plug-In was designed to solve a few common business use cases for Lead and Case assignment in your Salesforce instance:

1. Inserting or updating Leads and/or Cases from Process Builder whereby you do not have the ability to check the 'Assign using active assignment rules' checkbox.
2. Calling your Lead and/or Case Assignment Rules in a Flow.
3. Updating Lead records in Lightning will not trigger assignment rules - even if they are configured to fire.
 - a. The current workaround for this is to have users switch back to the Classic user interface where this functionality will work as expected (see the Salesforce Knowledge article for reference: <http://sforce.co/2vmzicy>).

The Assignment Plug-In solves all three of these business use cases:

1. The Assignment Plug-In can be used in a Process Builder process by calling a class delivered as a part of this package as an Action step. When that action is called, it will take the record (Lead or Case) and run it through your standard assignment rules.
2. The Assignment Plug-In is delivered with a Flow plug-in that can be used to call your standard assignment rules.
3. The Assignment Plug-In will work in Classic or Lightning – there's no need to toggle between the two user interfaces.

Important Note:

The Assignment Plug-In application, while invoked by Process Builder or Flow, executes in the context of the running user – not in a system context. This measure is a requirement of the AppExchange security review policy. This means that in order for the application to run as expected, the running user must have correct access to the objects, fields and ability to reassign Lead/Case records.

Using the Assignment Plug-In via Process Builder

The Assignment Plug-In delivers four Apex classes that can be used via Process Builder:

Class	Usage
Invocable_StandardAssignment	Simply execute your assignment rules in an immediate action
Invocable_StandardAssignmentV2	Execute your assignment rules in an immediate action, with the ability to trigger email notifications configured within your assignment rules
Invocable_StandardAssignmentFuture	Execute your assignment rules as a scheduled action
Invocable_StandardAssignmentFutureV2	Execute your assignment rules in an scheduled action, with the ability to trigger email notifications configured within your assignment rules

The key variation between the standard and the “V2” versions are the ability to also allow for emails to be sent as part of the reassignment process.

****Important Note**** If you are using the Assignment Plug-In for Case assignment via a Community, be sure to leverage the `Invocable_StandardAssignmentFuture` or `Invocable_StandardAssignmentFutureV2` classes within your process

To use the Assignment Plug-In in your Process Builder process, follow the steps below:

Immediate Action

1. Navigate to Process Builder via the Setup menu (Build → Create → Workflow & Approvals → Process Builder)
2. If there is already an existing process where you want to add the assignment rule call to, skip to step 7, otherwise, click on the New button from the Process Builder screen
3. Enter a name for the process and determine how this process will be called. For this example, we'll create a new process that is invoked when a record is changed

New Process

Process Name * API Name * ⓘ

Assignment Plug-In Demo Assignment_Plug_In_Demo

Description

The process starts when *

A record changes ▼

Cancel Save

4. Click the “+ Add Object” shape and select the object you want to have the Assignment Plug-In run on (Lead or Case)
5. Since the plug-in is designed for updates, select the “when a record is created or edited” option under the “Start the process” section and Save

Choose Object and Specify When to Start the Process

Object *

Lead ▼

Start the process *

☐ only when a record is created

☒ when a record is created or edited

6. Next, click the “+ Add Criteria” shape and enter the criteria for which you want your assignment rules to be run
 - For this example, we are simply looking to have the Assignment rules processed when the Lead.Status value = ‘Working – Contacted’

Define Criteria for this Action Group

Criteria Name * ⓘ

Status = Working - Contacted

Criteria for Executing Actions *

☒ Conditions are met

☐ Formula evaluates to true

☐ No criteria—just execute the actions!

Set Conditions

	Field *	Operator *	Type *	Value *
1	[Lead].Status	Equals	Picklist	Working - Contacte

+ Add Row

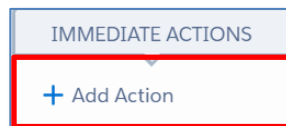
Conditions *

☒ All of the conditions are met (AND)

☐ Any of the conditions are met (OR)

☐ Customize the logic

7. Next, in the Immediate Actions shape, click on “+ Add Action”



8. In the resulting Action screen, select the following and Save:

- **Note:** The example shown here leverages the *Invocable_StandardAssignmentFutureV2* Apex class however the standard *Invocable_StandardAssignmentFuture* could be used here as well
- Action Type = Apex
- Action Name = <Intuitive Name for Action>
- Apex Class = *Invocable_StandardAssignmentFutureV2*
- Set Apex Variables section:
 - record Id
 - Type = Reference
 - Value = Either “Lead ID” or “Case ID”
 - triggerAutoResponseEmail
 - Set to “True” if auto-response rules should be triggered for the record, such as when a Case is created.
 - triggerOtherEmail
 - Set to “True” if email outside the organization should be triggered, for example notifications to Community Users for Case assignments.
 - triggerUserEmail

- Set to “True” if email that is sent to users inside your organization should be triggered. For example, use this setting to ensure that notification emails configured via lead assignment rules are sent upon assignment.

Select and Define Action

Action Type *

Apex

Action Name *

Call Assignment Rules

Apex Class *

Run_Assignment__Invocable_!

Set Apex Variables

Field *	Type *	Value *
record Id	Field Reference	[Lead].Id
triggerAutoResponseEmail	Boolean	False
triggerOtherEmail	Boolean	False
triggerUserEmail	Boolean	True

9. Finally, click the Activate button on your process to enable your new Assignment Rule process

Scheduled Action

1. Navigate to Process Builder via the Setup menu (Build → Create → Workflow & Approvals → Process Builder)
2. If there is already an existing process where you want to add the Assignment rule call to, skip to step 8, otherwise, click on the New button from the Process Builder screen
3. Enter a name for the process and determine how this process will be called. For this example, we'll create a new process that is invoked when a record is changed

New Process

Process Name * API Name * ⓘ

Assignment Plug-In Demo Assignment_Plug_In_Demo

Description

The process starts when *

A record changes ▼

Cancel Save

4. Click the “+ Add Object” shape and select the object you want to have the Assignment Plug-In run on (Lead or Case)
5. Since the plug-in is designed for updates, select the “when a record is created or edited” option under the “Start the process” section and Save

Choose Object and Specify When to Start the Process

Object *

Lead ▼

Start the process *

☐ only when a record is created

☒ when a record is created or edited

6. Next, click the “+ Add Criteria” shape and enter the criteria for which you want your Assignment rules to be run
 - For this example, we are simply looking to have the Assignment rules processed when the Lead.Status value = ‘Working – Contacted’

Define Criteria for this Action Group

Criteria Name * ⓘ

Status = Working - Contacted

Criteria for Executing Actions *

☒ Conditions are met

☐ Formula evaluates to true

☐ No criteria—just execute the actions!

Set Conditions

	Field *	Operator *	Type *	Value *
1	[Lead].Status	Equals	Picklist	Working - Contacte

+ Add Row

Conditions *

☒ All of the conditions are met (AND)

☐ Any of the conditions are met (OR)

☐ Customize the logic

7. Next, be sure to expand the 'Advanced' section and select the option to execute the actions only when specific changes are made to the record and Save
 - This is enabled so that we can run a delayed action on the record

▼ Advanced

Do you want to execute the actions only when specified changes are made to the record? ⓘ

☒ Yes

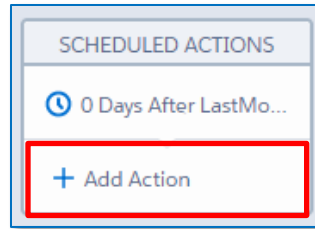
8. Next, skip the Immediate Actions shape and go to the Scheduled Actions shape and click on 'Set Schedule'
9. Set the time to execute the process and Save
 - In this example, since we want the assignment rules to fire as close to real-time as possible, we'll select 0 Days After LastModifiedDate

Set Time for Actions to Execute

☒ 0 Days After LastModifiedDate

☐ Days from now.

10. Under the schedule that was just added, click on "+ Add Action"



11. In the resulting Action screen, select the following and Save:

- **Note:** The example shown here leverages the *Invocable_StandardAssignmentV2* Apex class however the standard *Invocable_StandardAssignment* could be used here as well
- Action Type = Apex
- Action Name = <Intuitive Name for Action>
- Apex Class = *Invocable_StandardAssignmentV2*
- Set Apex Variables section:
 - record Id
 - Type = Reference
 - Value = Either “Lead ID” or “Case ID”
 - triggerAutoResponseEmail
 - Set to “True” if auto-response rules should be triggered for the record, such as when a Case is created.
 - triggerOtherEmail
 - Set to “True” if email outside the organization should be triggered, for example notifications to Community Users for Case assignments.
 - triggerUserEmail
 - Set to “True” if email that is sent to users inside your organization should be triggered. For example, use this setting to ensure that notification emails configured via lead assignment rules are sent upon assignment.

Select and Define Action

Action Type *

Apex

Action Name * ⓘ

Call Assignment Rules

Apex Class * ⓘ

Run_Assignment__Invokable_!

Set Apex Variables

Field *	Type *	Value *
record Id	Field Reference	[Lead].Id
triggerAutoResponseEmail	Boolean	False
triggerOtherEmail	Boolean	False
triggerUserEMail	Boolean	True

12. Finally, click the Activate button on your process to enable your new assignment rule process

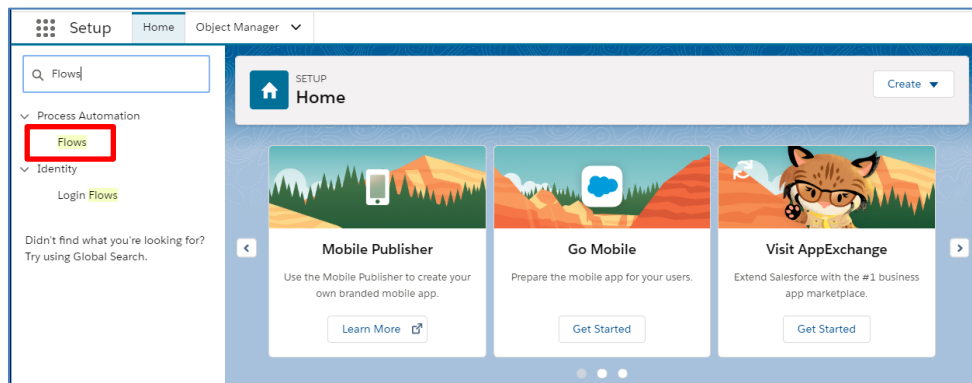
Using the Assignment Plug-In via Flow Builder

The Assignment Plug-In delivers the same four Apex classes available in Process Builder to your Flows:

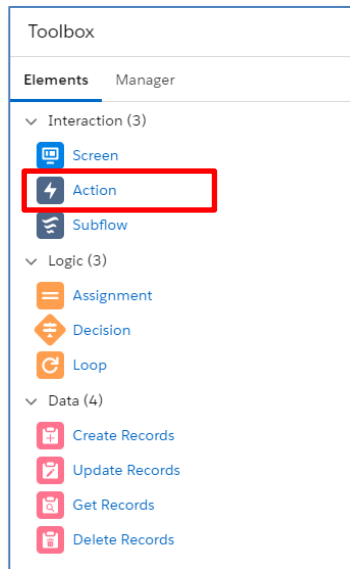
Class	Usage
Invocable_StandardAssignment	Simply execute your assignment rules in an immediate action
Invocable_StandardAssignmentV2	Execute your assignment rules in an immediate action, with the ability to trigger email notifications configured within your assignment rules
Invocable_StandardAssignmentFuture	Execute your assignment rules as a scheduled action
Invocable_StandardAssignmentFutureV2	Execute your assignment rules in a scheduled action, with the ability to trigger email notifications configured within your assignment rules

In order to use the Assignment Plug-In in your Flow, follow the steps below:

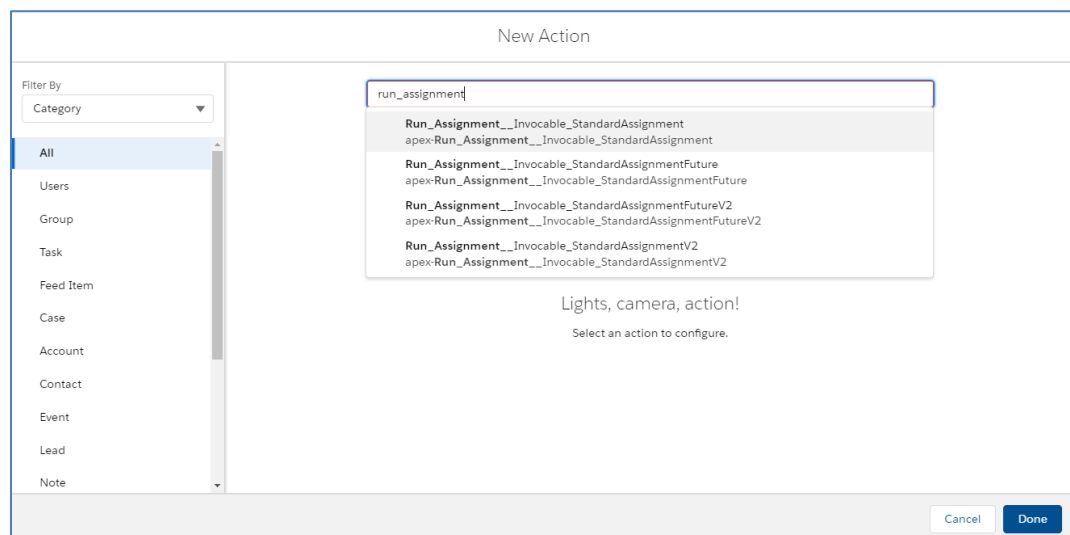
1. Navigate to your Flow via the Setup menu (Process Automation → Flows)



2. If there is already an existing Flow where you want to add the assignment rule call to, simply open that Flow. Otherwise, click on the New button from the Flows screen.
3. From the Toolbox on the left-hand side, find the Action icon and drag that on to the canvas



4. After placing that on the canvas, you'll be presented with a pop-up dialog
5. In the text box, start typing 'run_assignment' and you'll be presented with the four options available in the app



6. Select the Apex Class You'll Be Using
 - See more information on these classes in the above table
 - *For purposes of this example, the 'Run_Assignment__Invocable_StandardAssignmentFutureV2' class has been selected
7. Next, enter a Label & API Name for the Action
8. Next, in the record Id field, click within the text box and do the following:
 - Select 'New Resource' from the dropdown menu
 - From the next screen, select "Variable" as the Resource Type

- Enter an API Name for the variable
 - We'll be setting the Id for a Case record in this field so it will be named CaseId however any intuitive name can be used here
- Optionally enter a Description
- Under Data Type, select 'Text'
- Select both checkboxes for 'Available for input' & 'Available for output'
- Click Done

The screenshot shows the 'Edit Variable' dialog box. The 'API Name' field is filled with 'CaseId'. The 'Data Type' dropdown is set to 'Text'. Below the dropdown, there are two checkboxes: 'Available for input' and 'Available for output', both of which are checked. At the bottom right, there are 'Cancel' and 'Done' buttons. The 'Done' button is highlighted in blue.

9. After clicking Done, you'll be back at the New Action screen within your Flow
10. Click into the Record Id field again and you should now be able to select the variable value you created in the previous steps

The screenshot shows the 'New Action' dialog box. The 'Action Type' is set to 'Apex Action' and the 'Apex Action' is 'Run_Assignment__Invocable_StandardAssignmentFutureV2'. The 'Label' field contains 'Run Assignment Rules' and the 'API Name' field contains 'Run_Assignment_Rules'. The 'Record Id' field is highlighted with a red box and contains the formula '({CaseId})'. The 'Set Input Values' tab is selected.

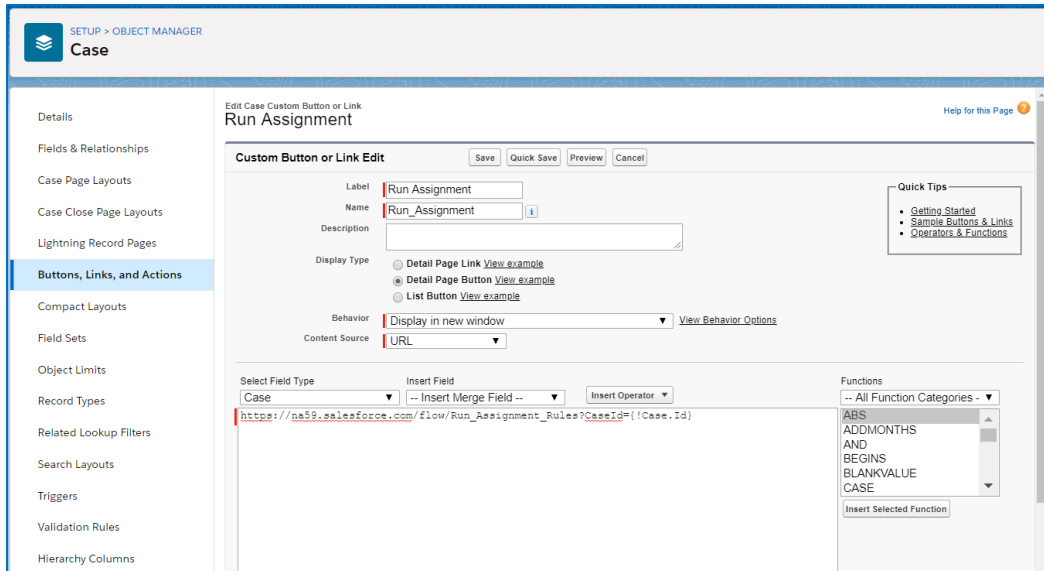
11. Next in the triggerAutoResponseEmail, triggerOtherEmail & triggerUserEmail fields, select either \$GlobalConstant.True or \$GlobalConstant.False

Set Input Values	Store Output Values
A ₃ *record.Id {!CaseId}	
∞ *triggerAutoResponseEmail {!SGlobalConstant.False}	
∞ *triggerOtherEmail {!SGlobalConstant.False}	
∞ *triggerUserEmail {!SGlobalConstant.False}	

12. Once all required fields have been populated, click the Done button
13. You'll be returned to the Flow Builder canvas where you can now connect the newly added Apex Action to the Start shape or embed it into your existing Flow
14. After it has been connected to your Flow, click the Save or Save As button

Creating Button to Call Flow Example

- From the Setup menu,
 - For Classic navigate to Build → Customize → Cases → Buttons, Links, and Actions
 - For Lightning, navigate to Object Manager, Case and then select 'Buttons, Links, and Actions' from the left hand navigation menu
- Click the New Action button
- From the resulting screen, enter the following information:
 - **Label** = <Intuitive Name for Running Flow> (Ex. Run Case Assignment Rules)
 - **Display Type** = Detail Page Button
 - **Behavior** = Display in new window
 - **Content Source** = URL
 - **URL Value** = `https://<X>.salesforce.com/flow/<Y>?<Z>={!Lead.Id}`
 - Where X = your Salesforce instance server, Y = the name of your Flow & Z = the name of your flow variable
 - Example:
`https://na59.salesforce.com/flow/Run_Assignment_Rules?CaseId={!Lead.Id}`



- Last, you'll want to be sure to add the newly created button to any Lead or Case page layouts where this button should be available
 - Note: You'll need a separate button and Flow for Leads and Cases. You won't be able to re-purpose the same assets for each object

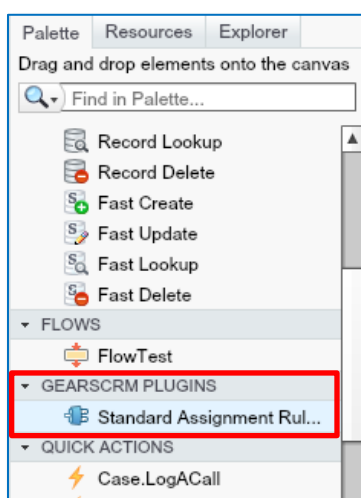
Using the Assignment Plug-In via Flow (Legacy Flow Builder)

If you'd like to incorporate the Assignment Plug-In into your flow, there is a plug-in provided via this managed package.

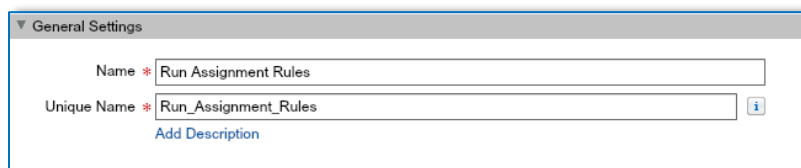
Important Note: This flow plug-in will not trigger additional notification emails that are available with the "V2" classes outlined in the Process Builder documentation.

In order to use the Assignment Plug-In in your Flow, follow the steps below:

1. Navigate to Flow via the Setup menu (Build → Create → Workflow & Approvals → Flows)
2. If there is already an existing Flow where you want to add the assignment rule call to, simply open that Flow, otherwise, click on the New button from the Flows screen
3. From the Palette on the left hand side, find the 'GEARSCRM PLUGINS' header and drag the 'Standard Assignment rules' plug-in onto the canvas



4. In the resulting dialog box enter the following:
 - **General Settings** section
 - Name = <Intuitive Name for the Plug-in>
 - Unique Name = <defaults based on the value entered in Name>



- **Inputs/Outputs** section
 - **Inputs**
 - sObjectId = <Create New Variable>
 - Unique Name = 'LeadId' or 'CaseId'

- Data Type = 'Text'
- Input/Output Type = 'Input and Output'
- Default Value = leave NULL

The image shows a 'Variable' dialog box with the title 'Variable' and a close button 'x'. Below the title is the instruction 'Create updatable values that can be used throughout your flow.' The dialog contains the following fields:

- Unique Name ***: A text box containing 'LeadId' with an information icon 'i' to its right.
- Description**: A large empty text area.
- Data Type**: A dropdown menu with 'Text' selected.
- Input/Output Type**: A dropdown menu with 'Input and Output' selected, with an information icon 'i' to its right.
- Default Value**: A dropdown menu with 'Enter value or select resource' selected.

At the bottom are 'OK' and 'Cancel' buttons.

■ Outputs

- Source = errorMessage
- Target = <Create New Variable>
 - Unique Name = 'ErrorMessage'
 - Data Type = 'Text'
 - Input/Output Type = 'Input and Output'
 - Default Value = leave NULL

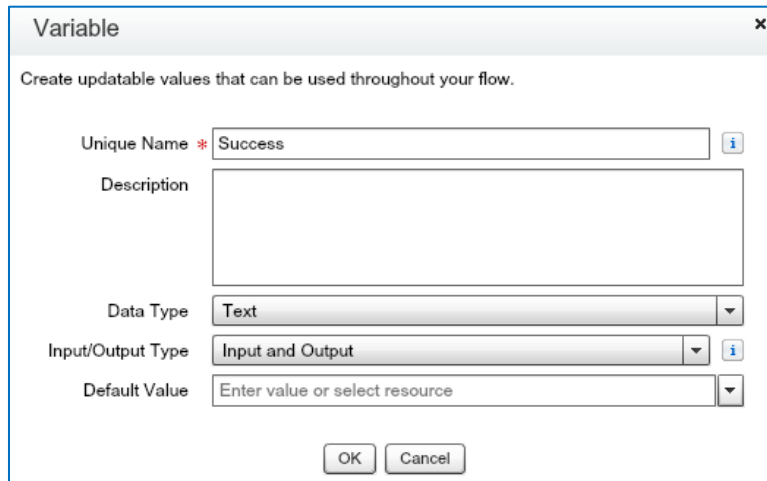
The image shows a 'Variable' dialog box with the title 'Variable' and a close button 'x'. Below the title is the instruction 'Create updatable values that can be used throughout your flow.' The dialog contains the following fields:

- Unique Name ***: A text box containing 'ErrorMessage' with an information icon 'i' to its right.
- Description**: A large empty text area.
- Data Type**: A dropdown menu with 'Text' selected.
- Input/Output Type**: A dropdown menu with 'Input and Output' selected, with an information icon 'i' to its right.
- Default Value**: A dropdown menu with 'Enter value or select resource' selected.

At the bottom are 'OK' and 'Cancel' buttons.

- Click 'Add Row' to add another Output
- Source = success
- Target = <Create New Variable>
 - Unique Name = 'Success'

- Data Type = 'Text'
- Input/Output Type = 'Input and Output'
- Default Value = leave NULL



Variable

Create updatable values that can be used throughout your flow.

Unique Name * Success

Description

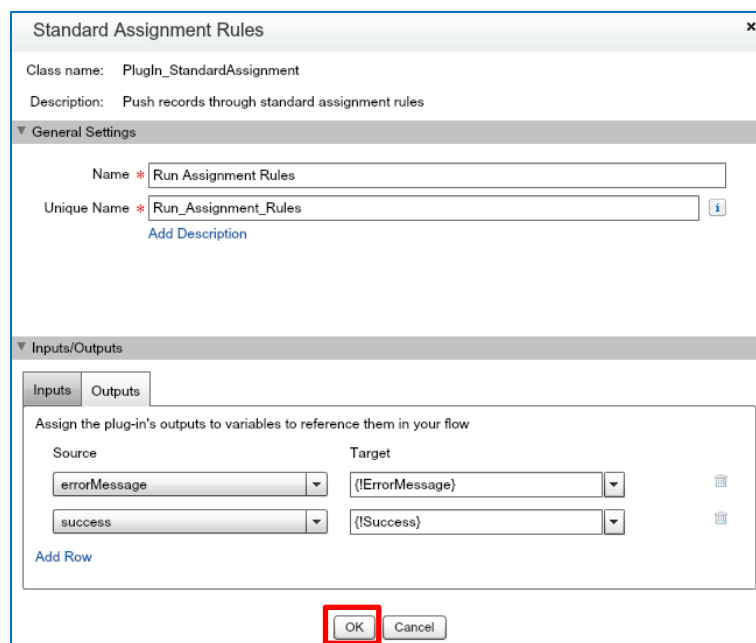
Data Type Text

Input/Output Type Input and Output

Default Value Enter value or select resource

OK Cancel

5. After entering all inputs click the OK button at the bottom of the dialog box



Standard Assignment Rules

Class name: PlugIn_StandardAssignment

Description: Push records through standard assignment rules

General Settings

Name * Run Assignment Rules

Unique Name * Run_Assignment_Rules

Add Description

Inputs/Outputs

Inputs Outputs

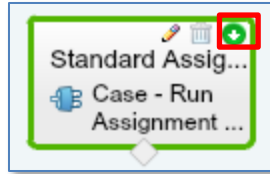
Assign the plug-in's outputs to variables to reference them in your flow

Source	Target
errorMessage	{!ErrorMessage}
success	{!Success}

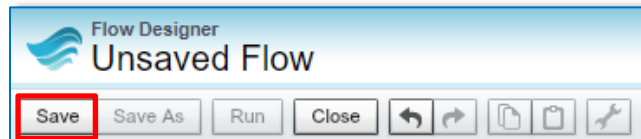
Add Row

OK Cancel

6. Next, click the green down arrow on the Standard Assignment rules plug-in shape to make that the starting element



7. Finally, save your Flow by clicking on the Save button above the Palette



8. From the resulting dialog box, enter an intuitive name for the Flow and leave all other fields as their default values and click OK

9. The Flow to call your Assignment rules has now been created but you'll need to add an action on either the Lead or Case object to call the Flow
 - In this example, we'll create a button on the Lead page layout that can be clicked to invoke the Flow that was just created
10. Navigate to the 'Buttons, Links, and Actions' setup for either the Lead or Case object (Build → Customize → Leads/Cases → Buttons, Links, and Actions)
11. Click on the 'New Button or Link' button at the top of the screen
12. Enter the following information into the new button or link screen and Save:
 - Label = <Intuitive Name for Running Flow> (Ex. Run Lead Assignment rules)
 - Display Type = Detail Page Button
 - Behavior = Display in new window
 - Content Source = URL
 - URL Value = https://<X>.salesforce.com/flow/<Y>?<Z>={!Lead.Id}

- Where X = your Salesforce server, Y = the name of your Flow & Z = the name of your flow variable
- Example:
https://na59.salesforce.com/flow/Run_Assignment_Rules?LeadId={!Lead.Id}

Lead Custom Button or Link
New Button or Link

Custom Button or Link Edit [Save] [Quick Save] [Preview] [Cancel]

Label: Run Lead Assignment Rules

Name: Run_Lead_Assignment_Rule

Description:

Display Type:

☐ Detail Page Link [View example](#)

☒ Detail Page Button [View example](#)

☐ List Button [View example](#)

Behavior: Display in new window [View Behavior Options](#)

Content Source: URL

Select Field Type: Lead Insert Field: -- Insert Merge Field -- Insert Operator:

https://na59.salesforce.com/flow/Run_Assignment_Rules?LeadId={!Lead.Id}

Functions:

-- All Function Categories --

ABS

AND

BEGINS

BLANKVALUE

CASE

CASESAFEID

[Insert Selected Function]

13. Last, you'll want to be sure to add the newly created button to any Lead or Case page layouts where this button should be available
 - Note: You'll need a separate button and Flow for Leads and Cases. You won't be able to re-purpose the same assets for each object

FAQ

Question: Can the Assignment Plug-In be used for both Leads & Cases at the same time?

Answer: Yes, the Assignment Plug-In can be used for both Leads & Cases at the same time. Since both have independent standard assignment rules, there is no risk of conflict.

Question: Can I use the plug-in for both Flow & Process Builder for the same object?

Answer: While this is possible, it is not recommended due to potential processing conflicts. If possible, configure the Assignment Plug-In either in Flow or Process Builder.

Question: For the Process Builder plug-in, why would I use the scheduled action instead of the immediate action?

Answer: In some situations, you may want to have the assignment delayed such that the order of execution is completed fully for a given transaction before you call assignment. In this case, you'd want to use the scheduled action vs. the immediate action plug-in.

Question: I've configured the Assignment Plug-In however it's not working as expected – why?

Answer: The most common reason for it not firing as expected is due to the standard Salesforce order of execution (<http://sforce.co/1hUzTY2> / <http://sforce.co/2um3bWQ>). If key criteria for your assignment rules are not set on the record being assigned prior to the time that the Assignment Plug-In is run, then there's a high likelihood that it will appear that something did not work properly. In this case, change the way in which you are calling the Assignment Plug-In such that it occurs as a scheduled action after the record has fully committed to the database vs. immediately calling the plug-in.

Question: Our business logic to determine when to run our assignment rules is pretty complex and ideally we'd like a solution outside of flow or process builder to call them – is this possible?

Answer: The Assignment Plug-In was designed primarily for use in more simplistic logic scenarios in either Process Builder or Flow. GearsDesign does have other applications available that can assist in these types of scenarios. Please contact support@gearsdesign.com or visit our website at <http://gearsdesign.com/> for more information.
