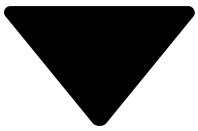
1. Workflow Rule Migration



2. Map Your Workflow Criteria to Process Criteria

Map Your Workflow Criteria to Process Criteria

Learning Objectives

After completing this unit, you'll be able to:

- Describe how the components of a workflow rule's criteria translate into process criteria.
- Combine multiple workflow rules into one process.
- Create a plan for converting workflow criteria into process criteria.

Important

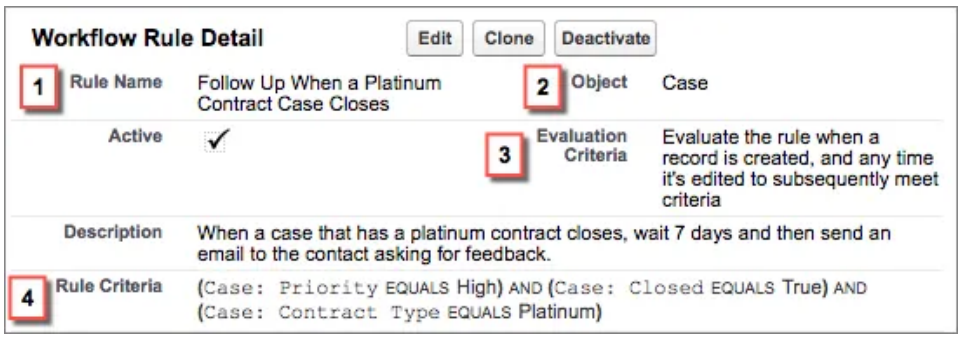
The examples described in this module include custom fields and email templates that don't exist in your Trailhead Playground. This module doesn't show you how to create those fields or templates, because our focus is on the concepts and best practices for migrating workflow rules. We recommend that you read along to understand the concepts, but don't try to follow the exercise steps in your playground. You apply your conceptual knowledge to complete the challenges at the end of each unit. Don't worry: you don't need to create any custom fields or email templates to complete any of the challenges.

Get the Lay of the Land

Before we begin analyzing your workflow rules and figuring out how they map into a process, let's get acquainted with the landscape.

Anatomy of a Workflow Rule

First up, a tour of the anatomy of a workflow rule's criteria. (We look at the actions later.) Along the way, we review how those pieces translate into the wide world in Process Builder.



Rule Name (1)
The names of workflow rules make nice names for criteria nodes, as long as the naming convention works for you.

> **Tip**
>
> You get a visual representation of the actions in Process Builder. So for criteria node names, we recommend summarizing only the criteria. However, if your org already has a naming convention, keep that momentum going. We heart consistency.

Object (2)
The object is a freebie; it's always the same for a process as it was for the corresponding workflow rule.

Evaluation Criteria (3)
Evaluation criteria is where it gets little bit complicated. But just a little! As a refresher, here are the evaluation criteria options for a workflow rule.

- created
- created or edited
- created or edited to subsequently meet criteria

The difference in Process Builder? The first two options are available when you select the object and specify when the process starts. The third option is an advanced setting in each criteria node.

Evaluation Criteria in Workflow	Values in Process Builder	
	Object node: Start the process	Criteria node: Advanced option
created	only when a record is created	n/a
created or edited	when a record is created or edited	Not selected
created or edited to subsequently meet criteria	when a record is created or edited	Selected

Rule Criteria (4)
Rule criteria map closely, if not exactly, from your workflow rule to your process. Just like Workflow, Process Builder lets you set filter conditions or use a formula.

One Process to Rule Them All

Workflow rules are always *nondeterministic*. In regular-person language: Salesforce can't guarantee which workflow rule is evaluated first or second or seventh. To make sure that workflow rules are evaluated in the right order, you have to get creative. Some customers add a field that tracks which workflow rule's turn it is. Yikes. That sounds like a lot of work, doesn't it?

Process Builder takes those workarounds and puts them in a box marked "Not Needed Anymore". As long as you automate everything in a single process, you explicitly set the order. The first criteria node is evaluated first, the second is evaluated second, and so on. So it's *very* important to order your criteria nodes correctly. In a later unit, we cover ordering in depth. Rest easy knowing that you can rearrange your criteria nodes.

If you create multiple processes for a given object, you risk the same problem as you had with workflow rules. Salesforce can't guarantee which process to evaluate first. So we recommend automating everything in one process, whenever possible. Meet a couple features within Process Builder that ease your path to one master process.

`ISNEW()`

Some of your automation applies only to created records. The rest applies to created and edited records. How could you possibly combine all of them into one process? Meet the formula function that detects whether the record being evaluated was just created: `ISNEW()`.

To add a create-only workflow rule to a process that starts when a record is created or edited, convert the associated rule's conditions to a formula. Then add `&& ISNEW()` to your formula.

Invocable Processes

Just like you can launch flows from a process, you can invoke other processes. Basically, invocable processes are modular processes that start only when another process tells them to. Think of invocable processes as an organizational tool to keep you and other admins sane.

Here are some example situations where invocable processes are useful.

- **Partially matching criteria nodes**

  For example, several criteria nodes evaluate whether an account is high value in addition to other conditions. You build an invocable process to handle all those criteria nodes, without the common condition. In your top-level process, if the account is high value you invoke the "Top Account" process.

- **Common action across multiple action groups**

  For example, your process notifies the account owner about the status of a case for many different statuses. Rather than building the same action for each criteria node, build it once with an invocable process. Doing so also reduces the maintenance cost.

Map Your Workflow Criteria into Process Criteria

Now, taking all that info into account, let's build a migration plan for the workflow rules' criteria.

Determine a Process Name

Your process spans the equivalent of multiple workflow rules, so any old rule name won't do. Since you're combining all of the case workflow rules into one process, call this one "Case Management".

Map the Object and Rule Criteria

The object and rule criteria map exactly.

Map the Evaluation Criteria

Here's how the evaluation criteria and rule names for the workflow rules translate into process criteria. The names now summarize the criteria rather than the criteria and actions. To convert workflow rules that evaluate when records are created or edited to subsequently meet criteria, mark the advanced option in the criteria node as selected.

Workflow Rules		**Process Builder**		
Rule Name	**Evaluation Criteria**	**Object Node:** `Start process when`	**Criteria Node:** `Criteria Name`	**Criteria Node: Advanced option**
Escalate Based on Keywords in Subject or Description	created or edited to subsequently meet criteria	when a record is created or edited	Escalation Keywords	Selected
Follow Up When a Platinum Contract Case Closes	created or edited to subsequently meet criteria	when a record is created or edited	Platinum Contract Case Closes	Selected
Notify Sales VP About Cases Filed for High-Value Accounts	created	only when a record is created	Top Account	n/a
Set Resolution Date for Basic Support	created	only when a record is created	Basic Support	n/a
Set Resolution Date for Premium Support	created	only when a record is created	Premium Support	n/a
Set Resolution Date for Standard Support	created	only when a record is created	Standard Support	n/a

The criteria nodes have differing evaluation criteria. Four of them are applicable only when a record is created, and two are applicable when a record is created or edited. To combine them all into one process, use the ISNEW() function.

For each criteria node that applies only to created records:

1. Convert the filter conditions to formulas.
2. Add the `ISNEW()` function to the formula.
3. Mark the Advanced option "Not selected".

With those changes in place, the criteria nodes for all those workflow rules fit in one process. Huzzah!

Process "Case Management"	
Object Node	**Criteria Nodes**

Process “Case Management”			
Object Node	**Criteria Nodes**	**Criteria Name**	**Criteria for Executing Actions**	**Criteria Node: Advanced option**

`Object`: Case
 	
`Start the process`: when a record is created or edited
	Escalation Keywords	`Contains(LOWER( Subject ),"urgent") ||`
		`Contains(LOWER( Subject ),"password") ||`
		`Contains(LOWER( Subject ),"down") ||`
		`Contains(LOWER( Subject ),"emergency") ||`
		`Contains(LOWER( Subject ),"internal server error") ||`
		`Contains(LOWER( Description ),"urgent") ||`
		`Contains(LOWER( Description ),"password") ||`
		`Contains(LOWER( Description ),"down") ||`
		`Contains(LOWER( Description ),"emergency") ||`
		`Contains(LOWER( Description ),"internal server error")`
		
		Copy
			Selected

	Platinum Contract Case Closes	• (Case: Priority equals High) and • (Case: Closed equals True) and • (Case: Contract Type equals Platinum)	Selected
	Top Account	`(Account: Top Account equals true) && ISNEW()` Copy	n/a
	Basic Support	`(Case: Support Plan equals Basic) && ISNEW()` Copy	n/a
	Premium Support	`(Case: Support Plan equals Premium) && ISNEW()` Copy	n/a
	Standard Support	`(Case: Support Plan equals Standard) && ISNEW()` Copy	n/a

Candidates for an Invocable Process

Now let's consider whether to combine any of the criteria nodes into an invocable process.

Remember, one of the main use cases for invocable processes is when multiple criteria nodes share a condition. Well, the `ISNEW()` function you just added counts as a shared condition. Let's put all the "create only" criteria nodes into an invocable process.

The shared condition (`ISNEW()`) is evaluated by a criteria node in the top-level process. So you can go back to using filter conditions for these criteria nodes.

All Together Now: The Final Criteria Plan

Here are the object and criteria settings for the invocable process. In invocable processes, the object node doesn't include the Start the process field and the criteria node doesn't include the advanced setting, so no need to track that information.

Invocable Process "New Cases"		
Object Node	**Criteria Nodes**	
	Criteria Name	**Criteria for Executing Actions**
Object: Case	Top Account	Account: Top Account equals true
	Basic Support	Support Plan equals Basic
	Premium Support	Support Plan equals Premium
	Standard Support	Support Plan equals Standard

Update the plan for the top-level process again. Replace the four criteria nodes that you moved to the invocable process with one criteria node (New Case). The new criteria node uses a formula (`ISNEW()`) to check whether the record was just created.

Process "Case Management"			
Object Node	**Criteria Nodes**		
	Criteria Name	**Criteria for Executing Actions**	**Criteria Node: Advanced Option**

Process "Case Management"			
Object Node	**Criteria Nodes**		
	Criteria Name	**Criteria for Executing Actions**	**Criteria Node: Advanced Option**

`Object`: Case
 	
Start the process: when a record is created or edited
	Escalation Keywords	`Contains(LOWER( Subject ),"urgent") ||`
		`Contains(LOWER( Subject ),"password") ||`
		`Contains(LOWER( Subject ),"down") ||`
		`Contains(LOWER( Subject ),"emergency") ||`
		`Contains(LOWER( Subject ),"internal server error") ||`
		`Contains(LOWER( Description ),"urgent") ||`
		`Contains(LOWER( Description ),"password") ||`
		`Contains(LOWER( Description ),"down") ||`
		`Contains(LOWER( Description ),"emergency") ||`
		`Contains(LOWER( Description ),"internal server error")`
		
		Copy
			Selected

	Platinum Contract Case Closes	• (Case: Priority equals High) and • (Case: Closed equals True) and • (Case: Contract Type equals Platinum)	Selected
	New Case	`&& ISNEW()` Copy	**Not selected**

That's it for the criteria plan!

Implement the Criteria in Your Process

Now let's put that plan into action. Remember, you separated some of the criteria nodes into an invocable process, so you're actually creating two processes. The plans include all the information that you need to create each process and configure the object and criteria nodes.

We could create either the top-level process or invocable process first. The order doesn't really matter when you're configuring criteria. So let's just pick one to do first: the invocable process.

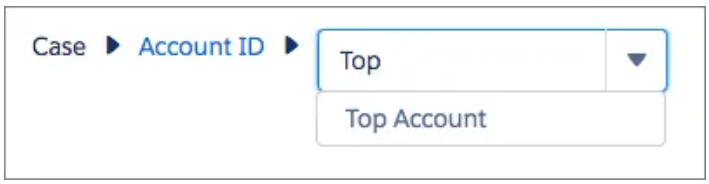
Criteria in the Invocable Process

Invocable Process "New Cases"	
Object Node	**Criteria Nodes**

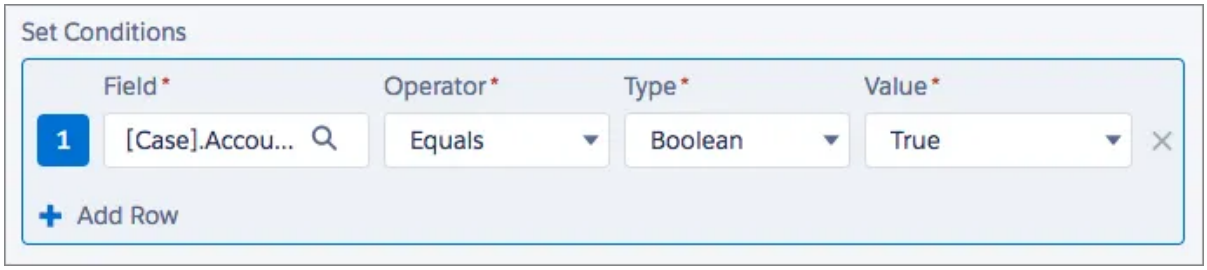
Invocable Process "New Cases"	**Criteria Name**	**Criteria for Executing Actions**

Object Node	**Criteria Nodes**	
	Criteria Name	**Criteria for Executing Actions**
Object: Case	Top Account	Account: Top Account equals true
	Basic Support	Support Plan equals Basic
	Premium Support	Support Plan equals Premium
	Standard Support	Support Plan equals Standard

1. From Setup, enter `Builder` in the Quick Find box, click **Process Builder**, and click **New**.
2. Configure the process' properties.
    - Name: New Cases
    - Description: Automation for all cases on creation
    - The process starts when: It's invoked by another process
3. Configure the object node.
    a. Click **Add Object**.
    b. Select **Case**, and save the object node.
4. Click **Add Criteria**, and configure the Top Account criteria node.
    a. Name the criteria "Top Account".
    b. For Field, select Top Account from the associated account.
        a. Click **Find a field…**
        b. Click **Account ID >** to select from the associated account's fields.
        c. Enter `Top` , select **Top Account**, then click **Choose**.



    c. For Value, select **True**.



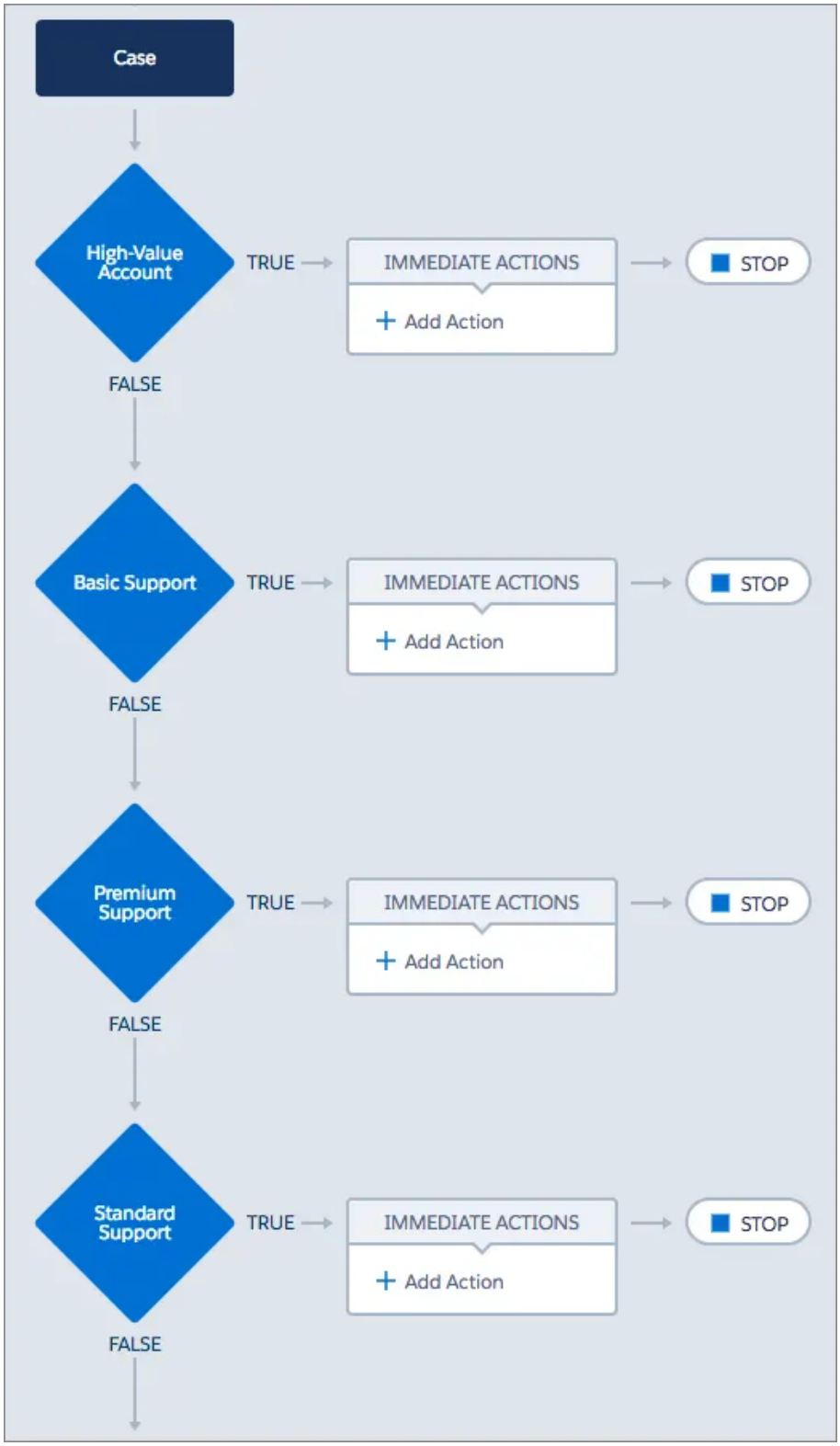
    d. Save the criteria node.
5. Click **Add Criteria**, and configure the Basic Support criteria node.

a. Name the criteria “Basic Support”.
b. For Field, select Support Plan.
    a. Click **Find a Field...**.
    b. Enter `Support` , select **Support Plan**, then click **Choose**.

c. For Value, select **Basic**.

d. Save the criteria node.

6. Click **Add Criteria**, and configure the Premium Support criteria node. Do the same thing that you did for Basic Support, except name the criteria node “Premium Support” and select **Premium** for the filter condition’s value.
7. Click **Add Criteria**, and configure the Standard Support criteria node. Do the same thing that you did for Basic Support, except name the criteria node “Standard Support” and select **Standard** for the filter condition’s value.



There you go—you've configured all the criteria nodes for the invocable process. Now on to the top-level process.

Criteria in the Top-Level Process

Process "Case Management"	
Object Node	**Criteria Nodes**

	Criteria Name	Criteria for Executing Actions	Criteria Node: Advanced Option

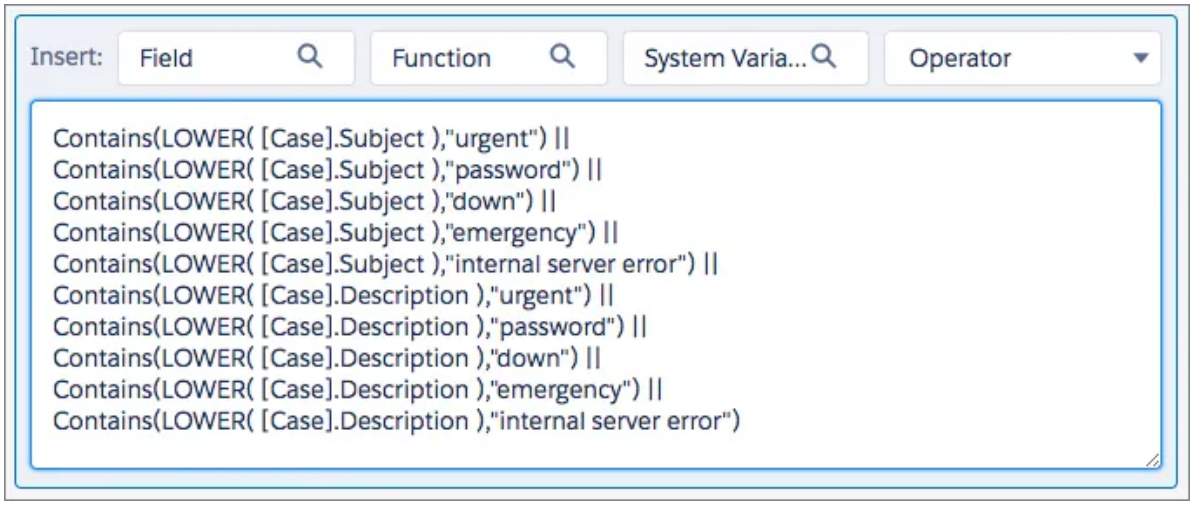
`Object`: Case
 	
Start the process: when a record is created or edited
	Escalation Keywords	`Contains(LOWER( Subject ),"urgent") ||`
		`Contains(LOWER( Subject ),"password") ||`
		`Contains(LOWER( Subject ),"down") ||`
		`Contains(LOWER( Subject ),"emergency") ||`
		`Contains(LOWER( Subject ),"internal server error") ||`
		`Contains(LOWER( Description ),"urgent") ||`
		`Contains(LOWER( Description ),"password") ||`
		`Contains(LOWER( Description ),"down") ||`
		`Contains(LOWER( Description ),"emergency") ||`
		`Contains(LOWER( Description ),"internal server error")`
		
		Copy
			Selected

	Platinum Contract Case Closes	• (Case: Priority equals High) and • (Case: Closed equals True) and • (Case: Contract Type equals Platinum)	Selected
	New Case	`&& ISNEW()` Copy	**Not selected**

1. In the button bar, click **View All Processes** to return to the process management page.
2. Click **New** and configure the process' properties.
    - Name: Case Management
    - Description: Automation for all cases
    - The process starts when: A record changes
3. Configure the object node.
    a. Click **Add Object**.
    b. Select **Case** and **when a record is created or edited**.
    c. Save the object node.
4. Click **Add Criteria**, and configure the Escalation Keywords criteria node.
    a. Name the criteria "Escalation Keywords".
    b. Select **Formula evaluates to true**.
    c. Copy the formula from the plan, and paste it into the formula editor.
    d. Update the fields in the formula so that they use Process Builder syntax. Use the field picker to insert the correct syntax for Subject and Description throughout the formula.

Note

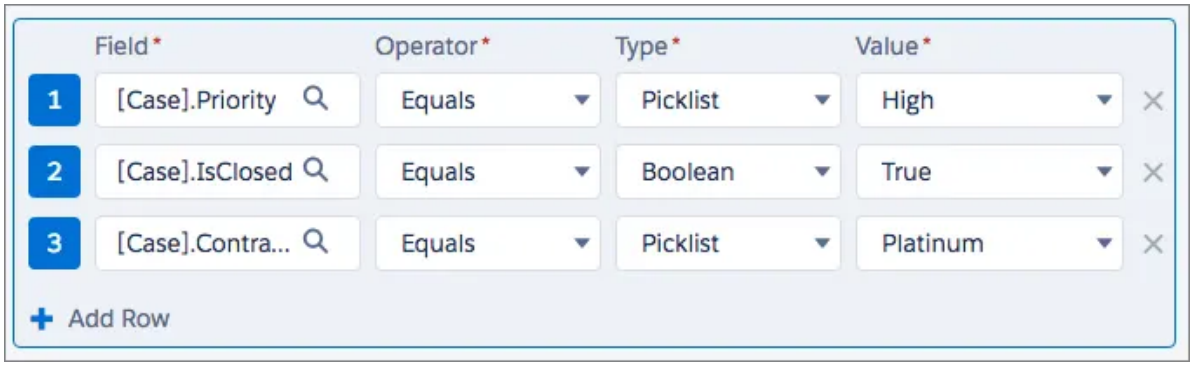
Functions and operators don't have different syntax in Process Builder, but field names do.



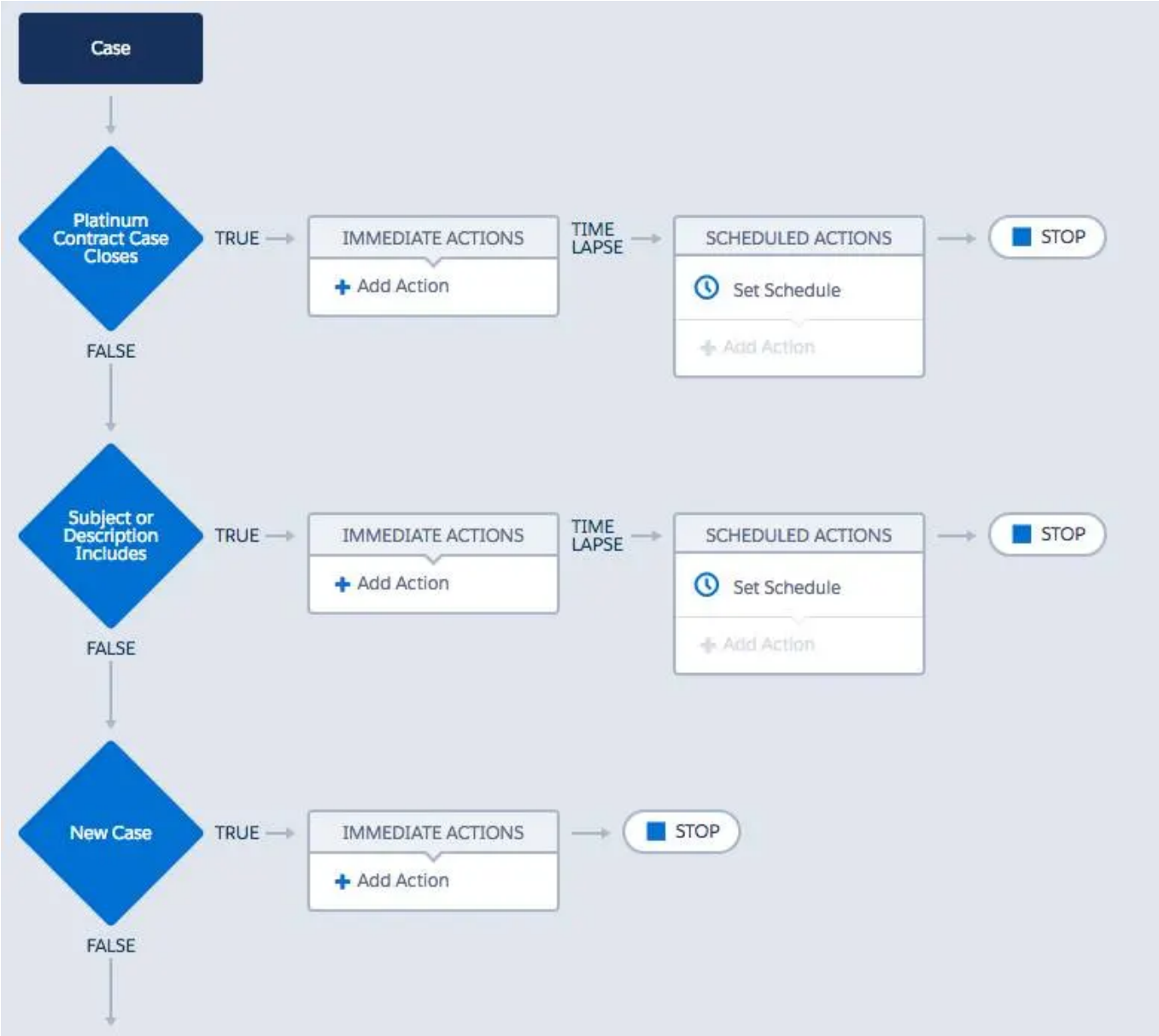
e. At the bottom of the criteria node, expand **> Advanced** and select **Yes**.

f. Save the criteria node.

5. Click **Add Criteria**, and configure the Platinum Contract Case Closes criteria node.
   a. Name the criteria "Platinum Contract Case Closes".
   b. Add two rows so that you have three filter conditions to configure.
   c. Select the fields and values for each row: Priority, Closed, and Contract Type.
   d. Set the values for each row so that Priority equals High, Closed equals True, and Contract Type equals Platinum.



   e. At the bottom of the criteria node, expand **> Advanced** and select **Yes**.
   f. Save the criteria node.
6. Click **Add Criteria**, and configure the New Case criteria node.
   a. Name the criteria "New Case".
   b. Select **Formula evaluates to true**.
   c. In the formula editor, enter `ISNEW()`.
   d. Leave the advanced option unselected, and save the criteria node.



That’s a wrap for all of the criteria nodes. Next, let’s convert the workflow rule actions into process actions.

Resources

- Configure the Process Trigger
- Add Process Criteria
- Avoid Unwanted Actions in Processes

Assessment Complete!

+500 points

Workflow Rule Migration
100%
Progress: 100%
Retake this Challenge
View more modules