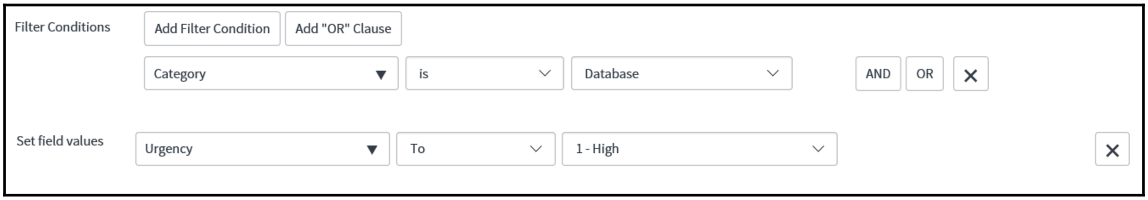
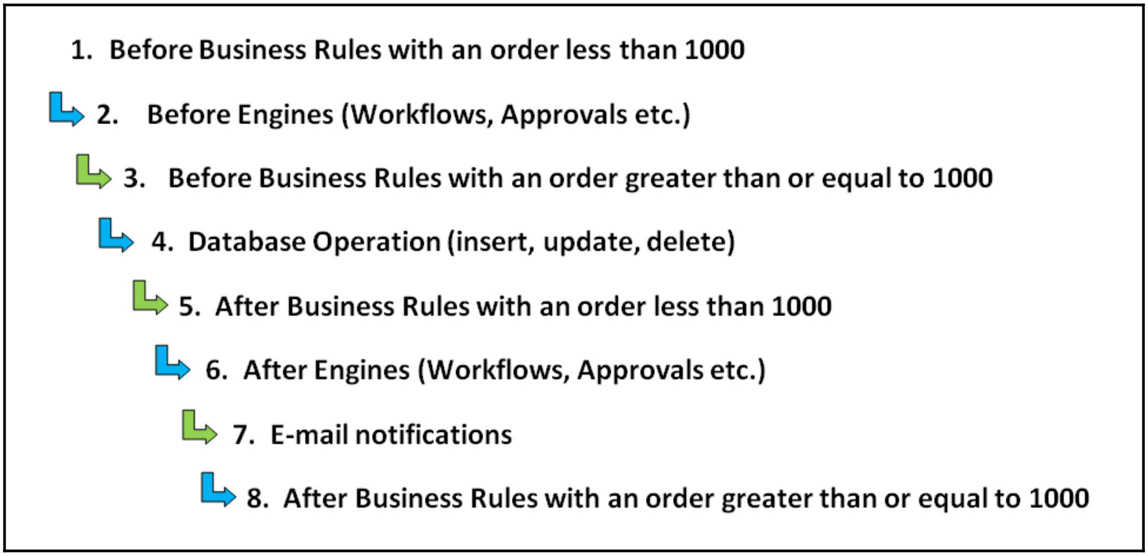
Chapter 1: Getting Started

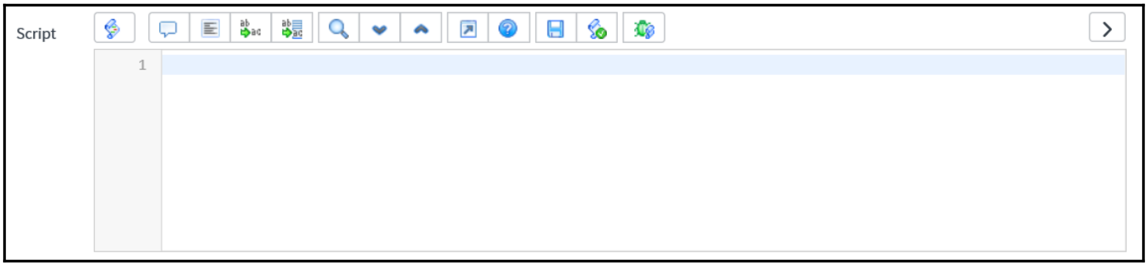


Filter Conditions: Category is Database

Set field values: Urgency To 1 - High

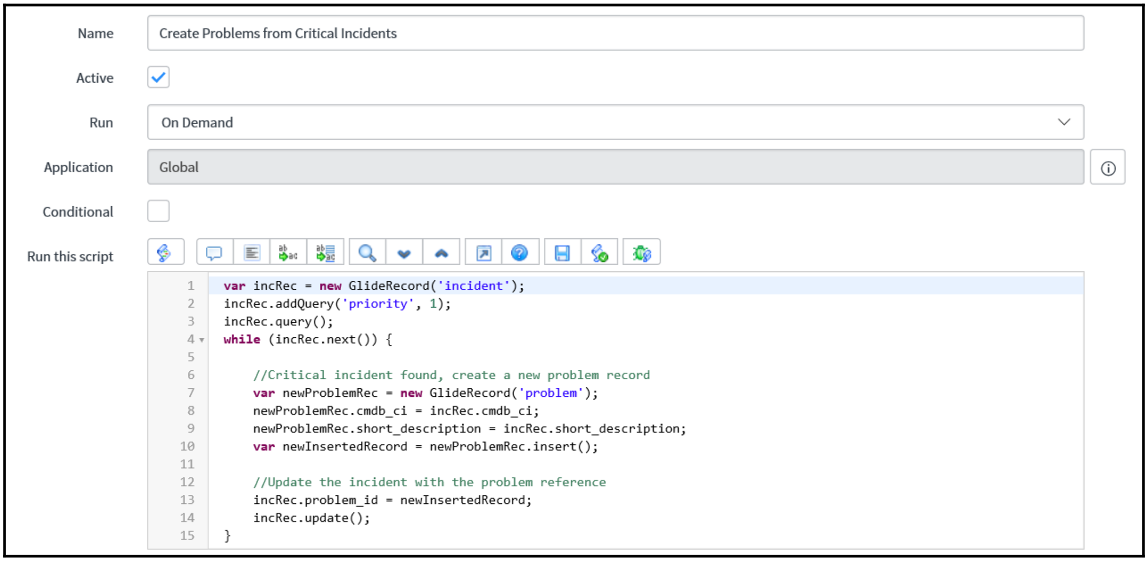


1. Before Business Rules with an order less than 1000
2. Before Engines (Workflows, Approvals etc.)
3. Before Business Rules with an order greater than or equal to 1000
4. Database Operation (insert, update, delete)
5. After Business Rules with an order less than 1000
6. After Engines (Workflows, Approvals etc.)
7. E-mail notifications
8. After Business Rules with an order greater than or equal to 1000



Script

Chapter 2: Exploring the ServiceNow Glide Class



Name: Create Problems from Critical Incidents

Active: ☑

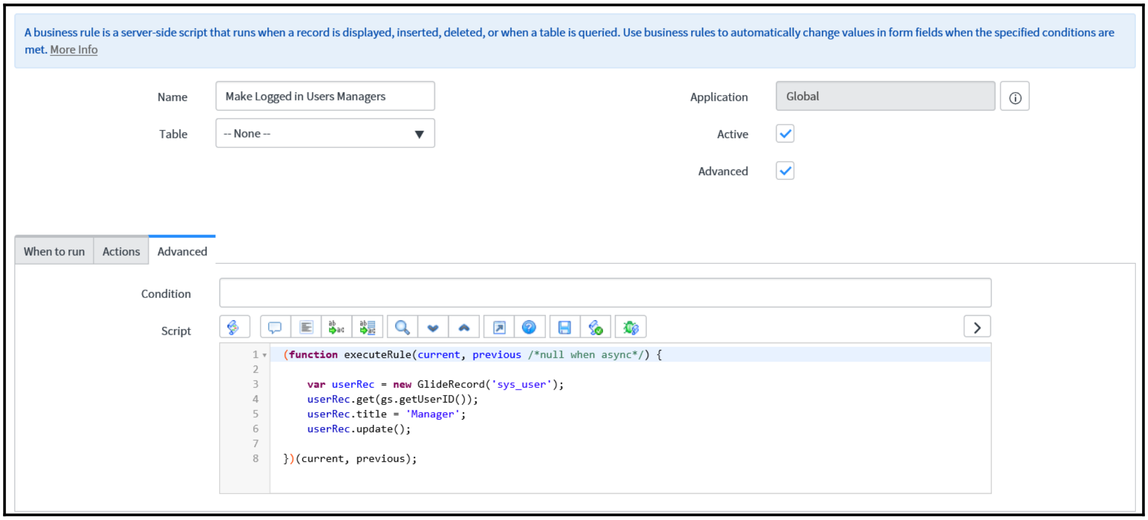
Run: On Demand

Application: Global

Conditional: □

Run this script

```
var incRec = new GlideRecord('incident');
incRec.addQuery('priority', 1);
incRec.query();
while (incRec.next()) {

    //Critical incident found, create a new problem record
    var newProblemRec = new GlideRecord('problem');
    newProblemRec.cmdb_ci = incRec.cmdb_ci;
    newProblemRec.short_description = incRec.short_description;
    var newInsertedRecord = newProblemRec.insert();

    //Update the incident with the problem reference
    incRec.problem_id = newInsertedRecord;
    incRec.update();
}
```



A business rule is a server-side script that runs when a record is displayed, inserted, deleted, or when a table is queried. Use business rules to automatically change values in form fields when the specified conditions are met. More Info

Name: Make Logged in Users Managers

Table: -- None --

Application: Global

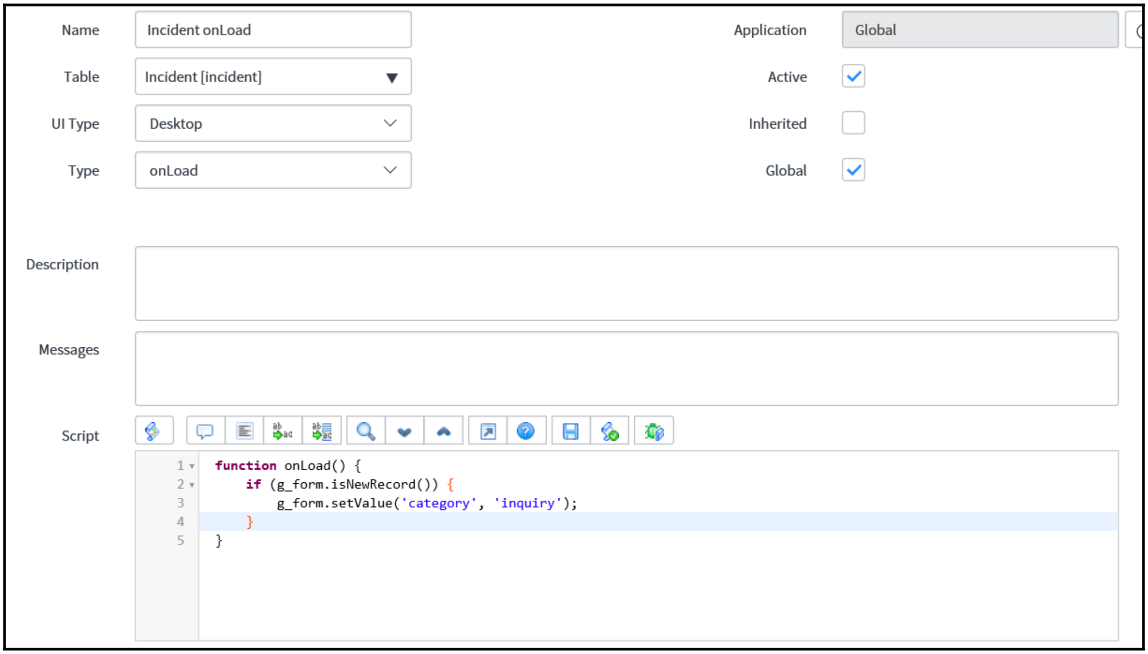
Active: ☑

Advanced: ☑

Condition:

Script

```
(function executeRule(current, previous /*null when async*/) {

    var userRec = new GlideRecord('sys_user');
    userRec.get(gs.getUserID());
    userRec.title = 'Manager';
    userRec.update();

})(current, previous);
```



Name	Incident onLoad	Application	Global
Table	Incident [incident]	Active	☑
UI Type	Desktop	Inherited	□
Type	onLoad	Global	☑

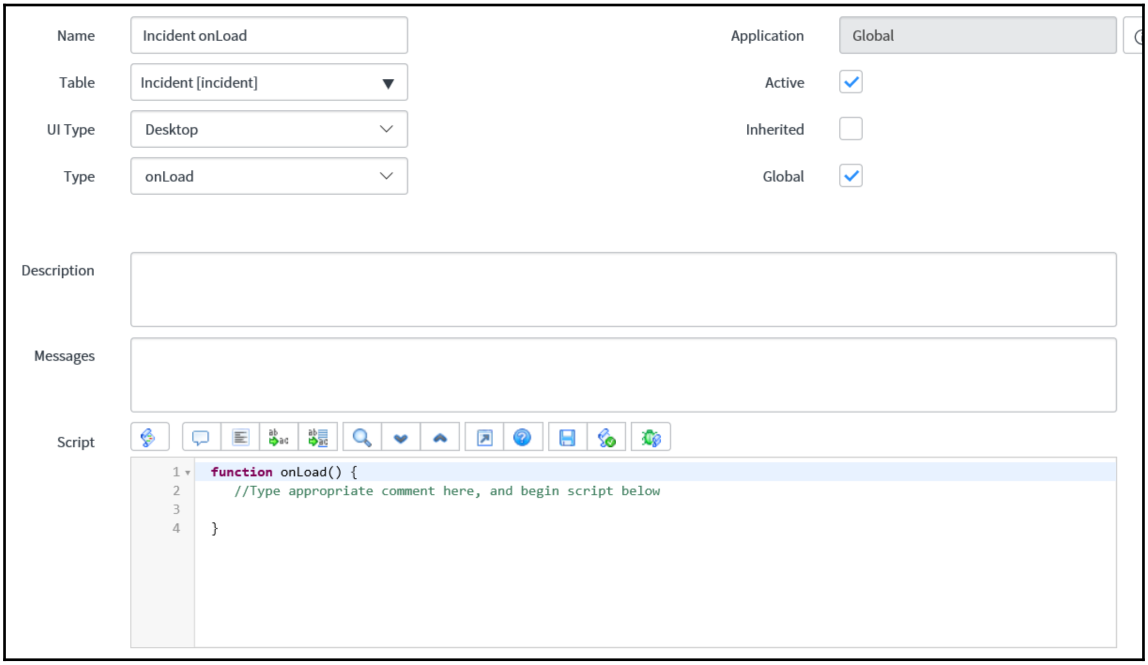
Description

Messages

Script

```
function onLoad() {
    if (g_form.isNewRecord()) {
        g_form.setValue('category', 'inquiry');
    }
}
```

Chapter 3: Introduction to Client-Side Scripting



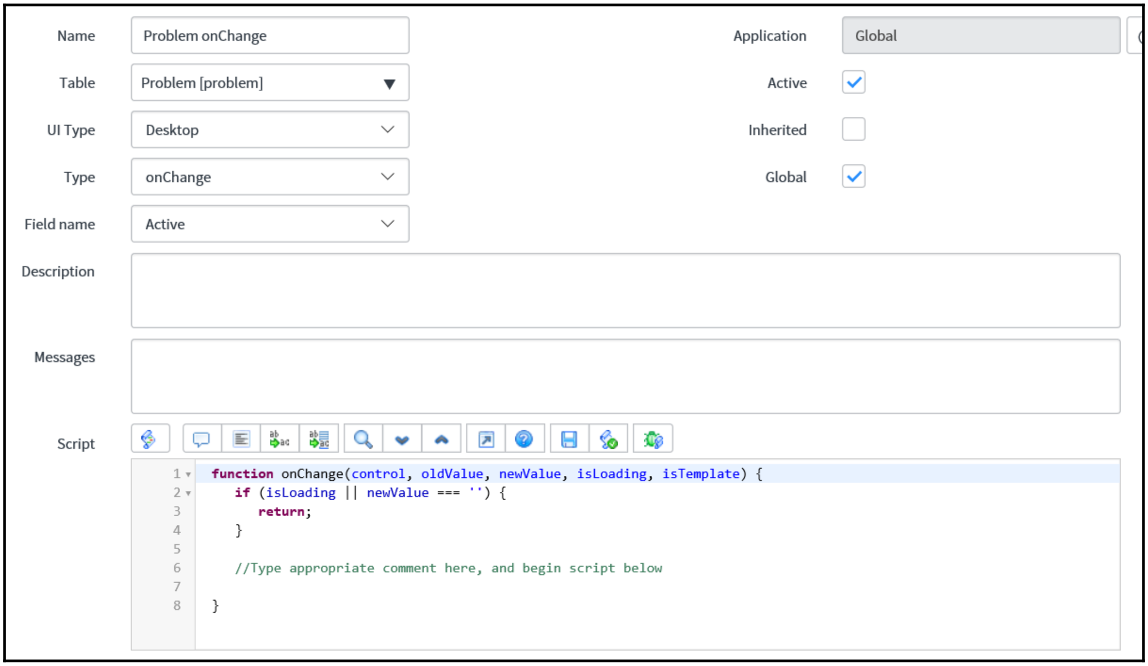
Name	Incident onLoad	Application	Global
Table	Incident [incident]	Active	☑
UI Type	Desktop	Inherited	□
Type	onLoad	Global	☑

Description

Messages

Script

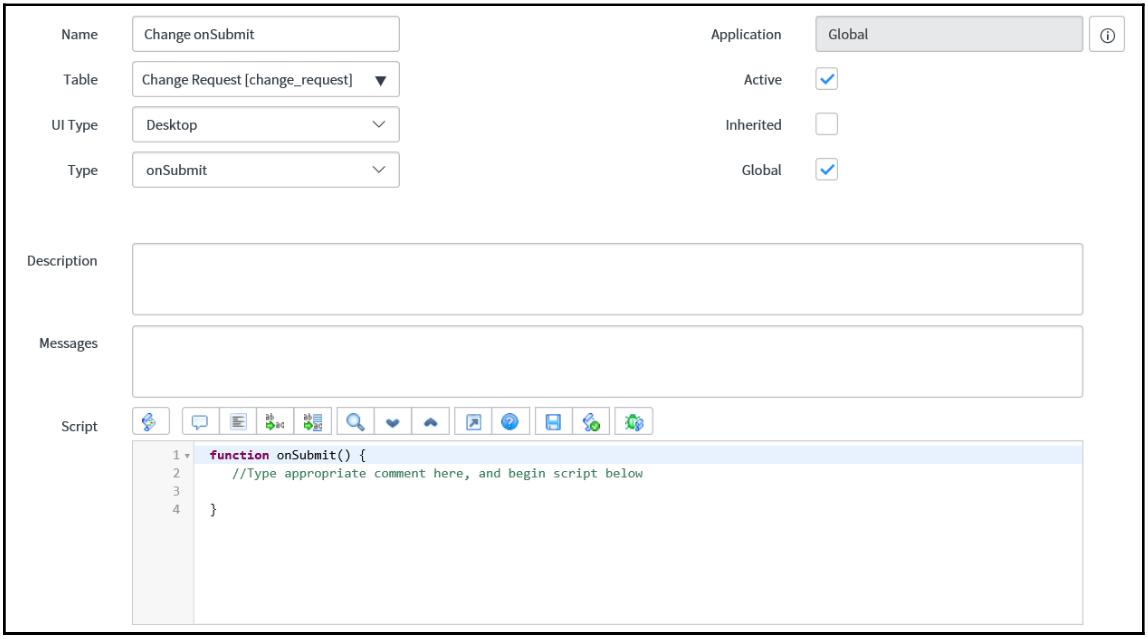
```
function onLoad() {
   //Type appropriate comment here, and begin script below

}
```



Name	Problem onChange	Application	Global
Table	Problem [problem]	Active	☑
UI Type	Desktop	Inherited	□
Type	onChange	Global	☑
Field name	Active		
Description			
Messages			

Script

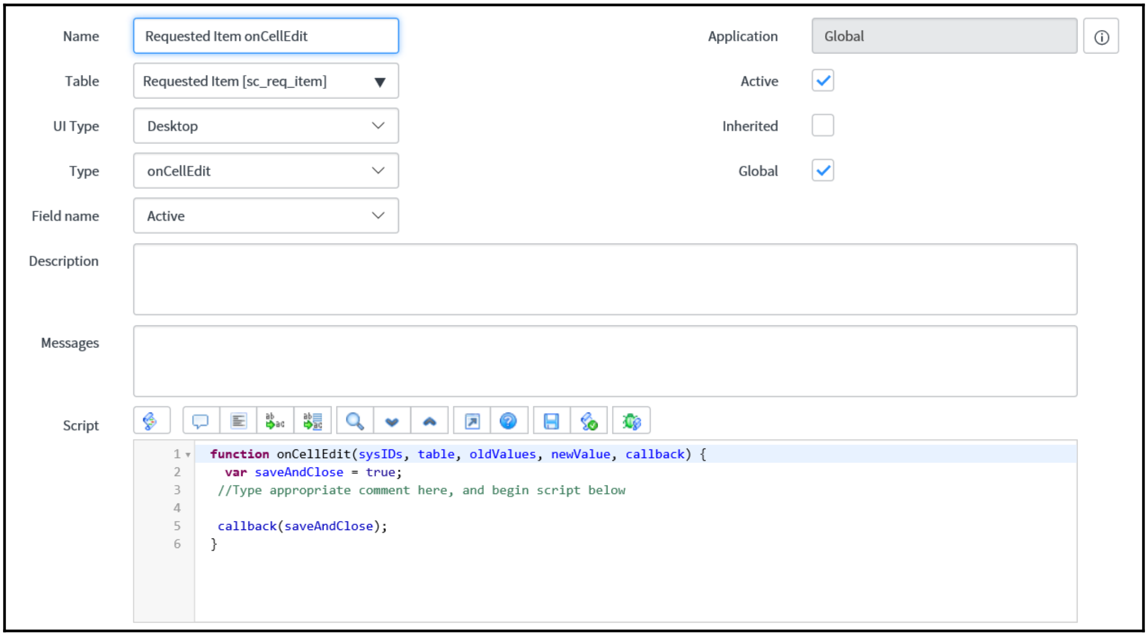
```
function onChange(control, oldValue, newValue, isLoading, isTemplate) {
   if (isLoading || newValue === '') {
      return;
   }

   //Type appropriate comment here, and begin script below

}
```



Name	Change onSubmit	Application	Global
Table	Change Request [change_request]	Active	☑
UI Type	Desktop	Inherited	□
Type	onSubmit	Global	☑
Description			
Messages			

Script

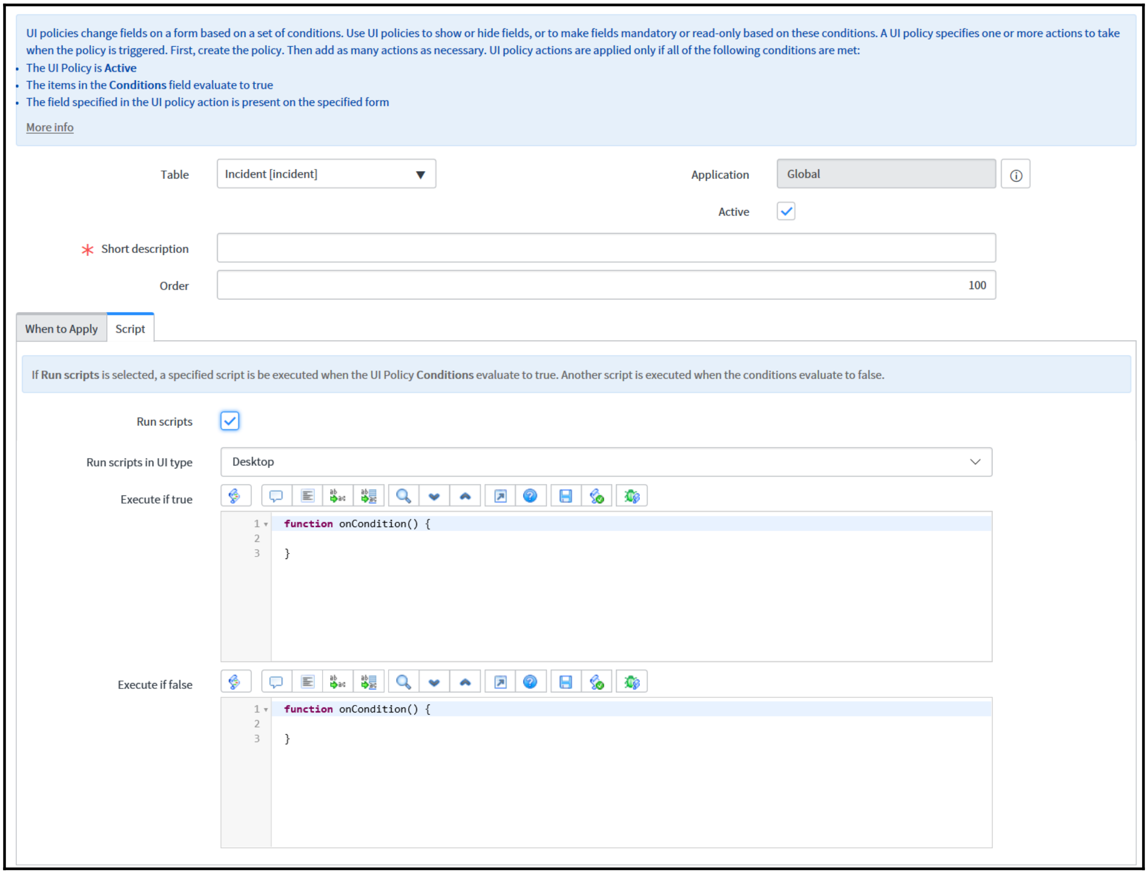
```
function onSubmit() {
   //Type appropriate comment here, and begin script below

}
```



Name	Requested Item onCellEdit	Application	Global
Table	Requested Item [sc_req_item]	Active	☑
UI Type	Desktop	Inherited	☐
Type	onCellEdit	Global	☑
Field name	Active		
Description			
Messages			

Script

```
function onCellEdit(sysIDs, table, oldValues, newValue, callback) {
  var saveAndClose = true;
 //Type appropriate comment here, and begin script below

 callback(saveAndClose);
}
```



UI policies change fields on a form based on a set of conditions. Use UI policies to show or hide fields, or to make fields mandatory or read-only based on these conditions. A UI policy specifies one or more actions to take when the policy is triggered. First, create the policy. Then add as many actions as necessary. UI policy actions are applied only if all of the following conditions are met:

- The UI Policy is **Active**
- The items in the **Conditions** field evaluate to true
- The field specified in the UI policy action is present on the specified form

More info

Table	Incident [incident]	Application	Global
		Active	☑
* Short description			
Order	100		

If **Run scripts** is selected, a specified script is be executed when the UI Policy **Conditions** evaluate to true. Another script is executed when the conditions evaluate to false.

Run scripts ☑

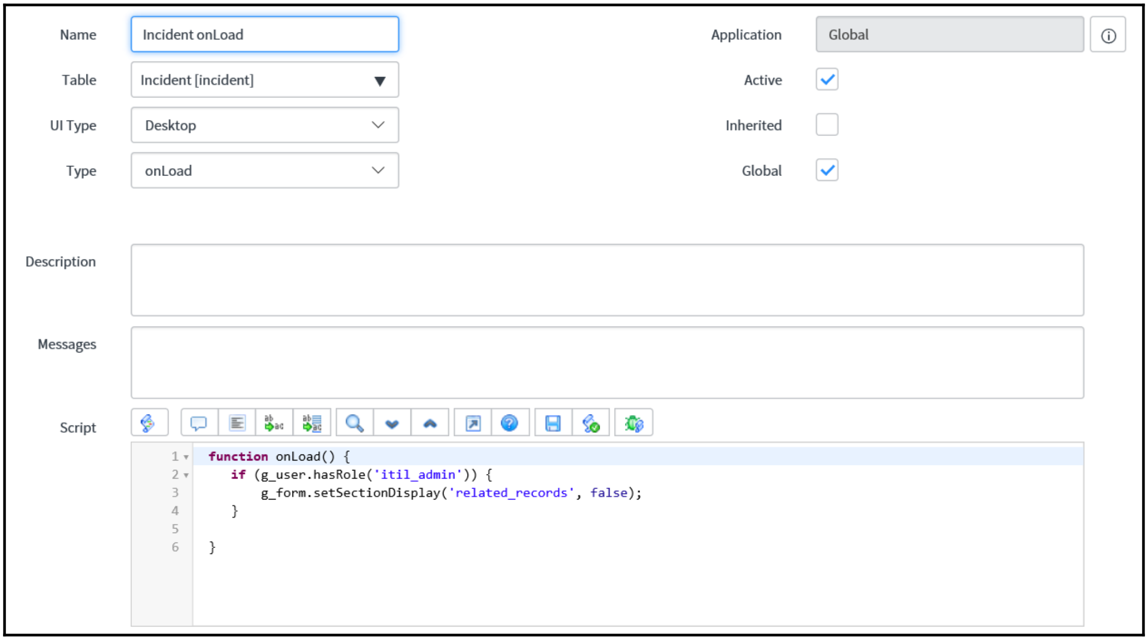
Run scripts in UI type: Desktop

Execute if true

```
function onCondition() {

}
```

Execute if false

```
function onCondition() {

}
```



Name: Incident onLoad

Table: Incident [incident]

UI Type: Desktop

Type: onLoad

Application: Global

Active ☑

Inherited □

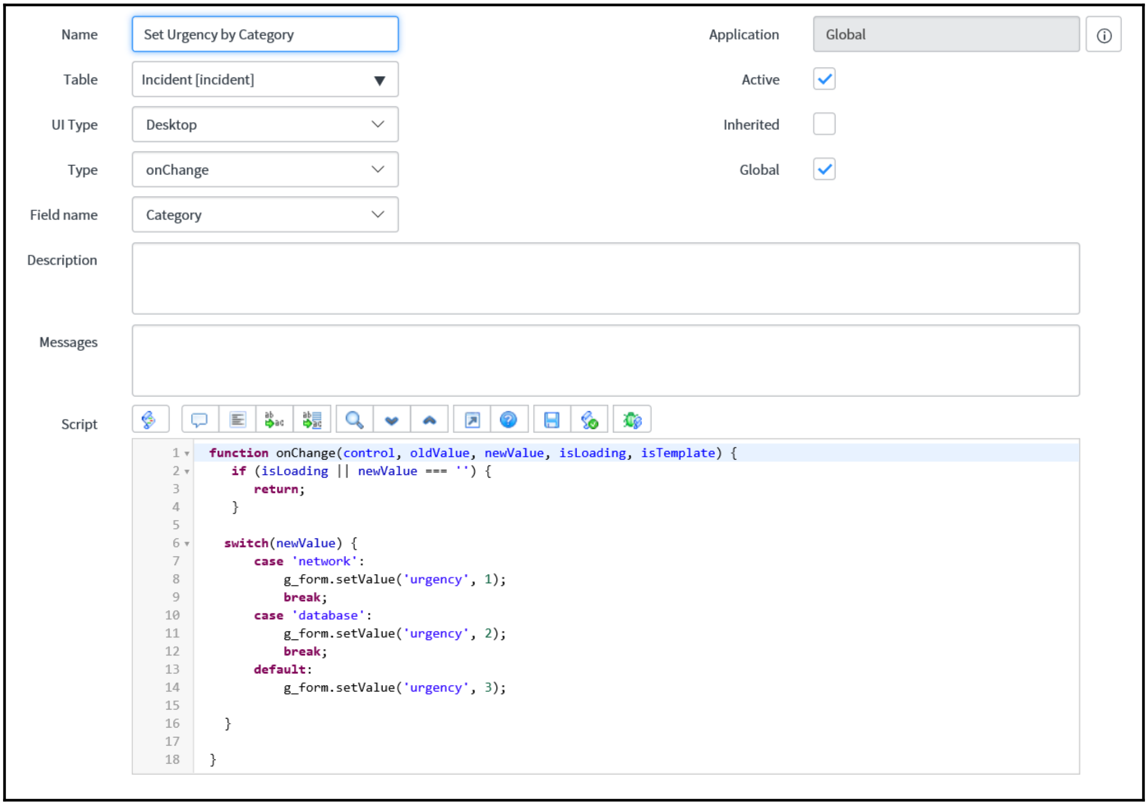
Global ☑

Description

Messages

Script

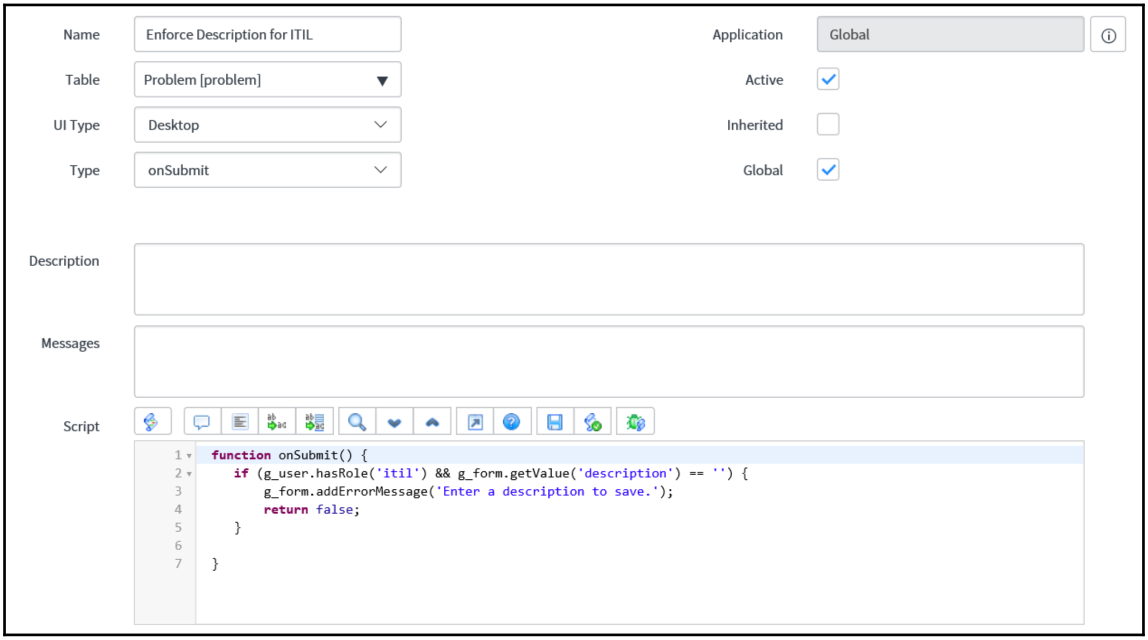
```
function onLoad() {
    if (g_user.hasRole('itil_admin')) {
        g_form.setSectionDisplay('related_records', false);
    }

}
```



Name	Set Urgency by Category	Application	Global
Table	Incident [incident]	Active	☑
UI Type	Desktop	Inherited	□
Type	onChange	Global	☑
Field name	Category		
Description			
Messages			

Script

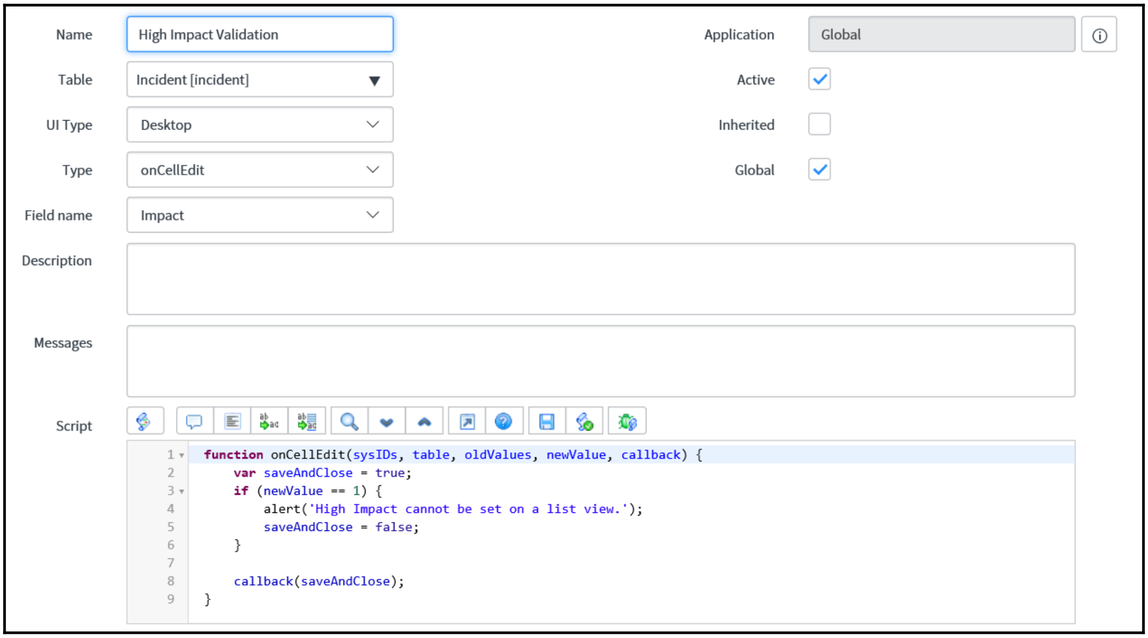
```
function onChange(control, oldValue, newValue, isLoading, isTemplate) {
   if (isLoading || newValue === '') {
      return;
   }

  switch(newValue) {
      case 'network':
          g_form.setValue('urgency', 1);
          break;
      case 'database':
          g_form.setValue('urgency', 2);
          break;
      default:
          g_form.setValue('urgency', 3);

  }

}
```



Name	Enforce Description for ITIL	Application	Global
Table	Problem [problem]	Active	☑
UI Type	Desktop	Inherited	□
Type	onSubmit	Global	☑
Description			
Messages			

Script

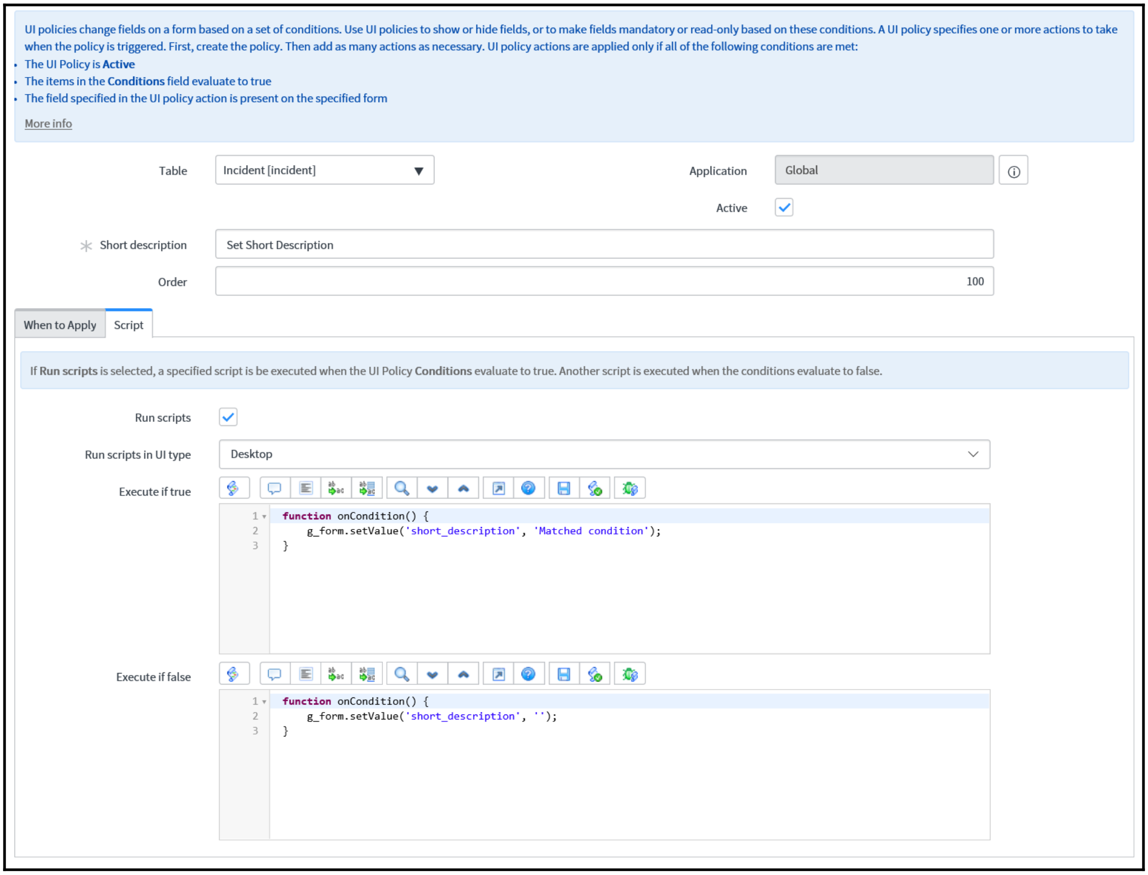
```
function onSubmit() {
    if (g_user.hasRole('itil') && g_form.getValue('description') == '') {
        g_form.addErrorMessage('Enter a description to save.');
        return false;
    }

}
```



Name	High Impact Validation	Application	Global
Table	Incident [incident]	Active	☑
UI Type	Desktop	Inherited	□
Type	onCellEdit	Global	☑
Field name	Impact		
Description			
Messages			

Script

```
function onCellEdit(sysIDs, table, oldValues, newValue, callback) {
    var saveAndClose = true;
    if (newValue == 1) {
        alert('High Impact cannot be set on a list view.');
        saveAndClose = false;
    }

    callback(saveAndClose);
}
```



UI policies change fields on a form based on a set of conditions. Use UI policies to show or hide fields, or to make fields mandatory or read-only based on these conditions. A UI policy specifies one or more actions to take when the policy is triggered. First, create the policy. Then add as many actions as necessary. UI policy actions are applied only if all of the following conditions are met:

- The UI Policy is **Active**
- The items in the **Conditions** field evaluate to true
- The field specified in the UI policy action is present on the specified form

More info

Table: Incident [incident]

Application: Global

Active: ☑

Short description: Set Short Description

Order: 100

If **Run scripts** is selected, a specified script is be executed when the UI Policy **Conditions** evaluate to true. Another script is executed when the conditions evaluate to false.

Run scripts: ☑

Run scripts in UI type: Desktop

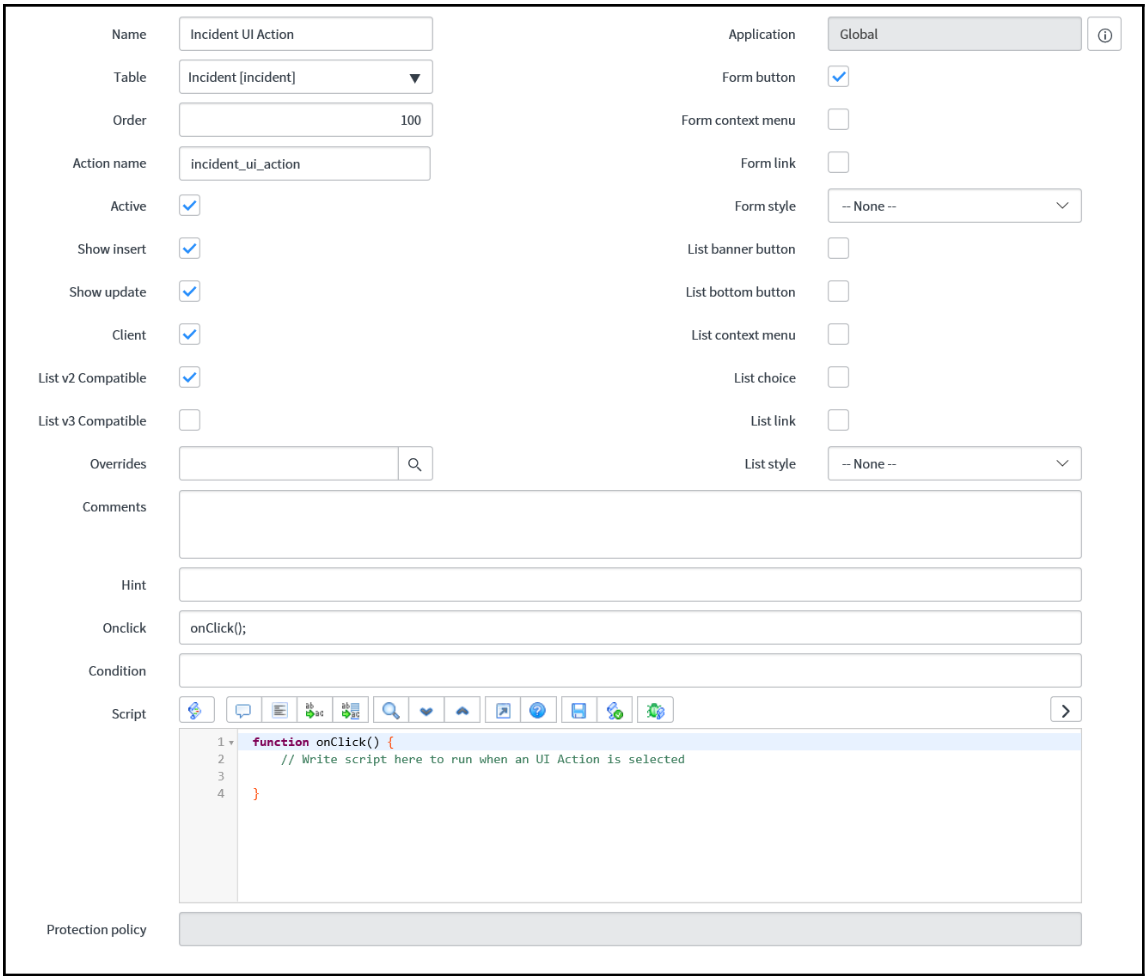
Execute if true

```
function onCondition() {
    g_form.setValue('short_description', 'Matched condition');
}
```

Execute if false

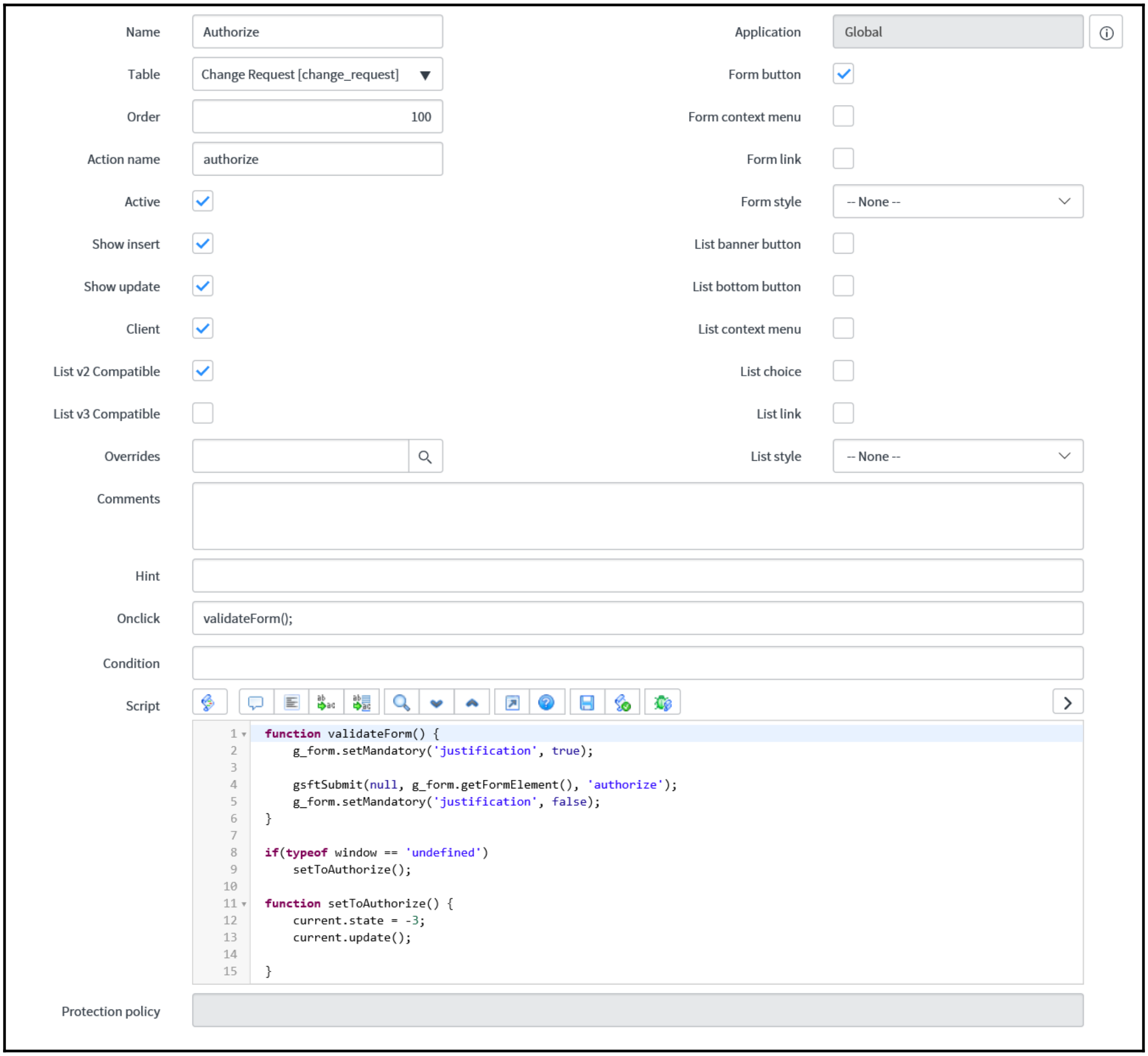
```
function onCondition() {
    g_form.setValue('short_description', '');
}
```

Chapter 4: Advanced Client-Side Scripting



Name	Incident UI Action	Application	Global
Table	Incident [incident]	Form button	☑
Order	100	Form context menu	☐
Action name	incident_ui_action	Form link	☐
Active	☑	Form style	-- None --
Show insert	☑	List banner button	☐
Show update	☑	List bottom button	☐
Client	☑	List context menu	☐
List v2 Compatible	☑	List choice	☐
List v3 Compatible	☐	List link	☐
Overrides		List style	-- None --
Comments			
Hint			
Onclick	onClick();		
Condition			
Script			
Protection policy			

```
function onClick() {
    // Write script here to run when an UI Action is selected

}
```



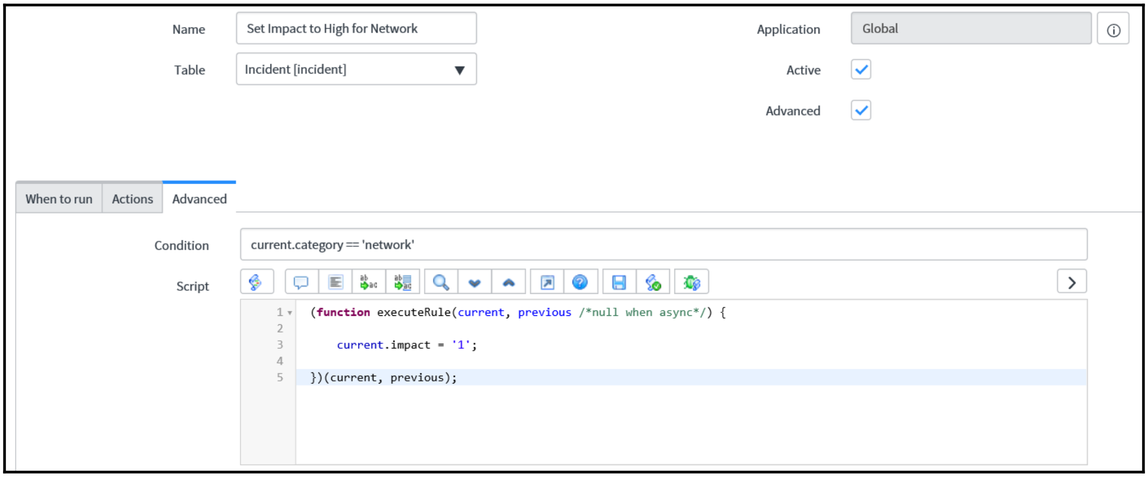
Name	Authorize	Application	Global
Table	Change Request [change_request]	Form button	☑
Order	100	Form context menu	□
Action name	authorize	Form link	□
Active	☑	Form style	-- None --
Show insert	☑	List banner button	□
Show update	☑	List bottom button	□
Client	☑	List context menu	□
List v2 Compatible	☑	List choice	□
List v3 Compatible	□	List link	□
Overrides		List style	-- None --
Comments			
Hint			
Onclick	validateForm();		
Condition			

Script

```
function validateForm() {
    g_form.setMandatory('justification', true);

    gsftSubmit(null, g_form.getFormElement(), 'authorize');
    g_form.setMandatory('justification', false);
}

if(typeof window == 'undefined')
    setToAuthorize();

function setToAuthorize() {
    current.state = -3;
    current.update();

}
```

Protection policy

Chapter 5: Introduction to Server-Side Scripting



Name: Set Impact to High for Network

Table: Incident [incident]

Application: Global

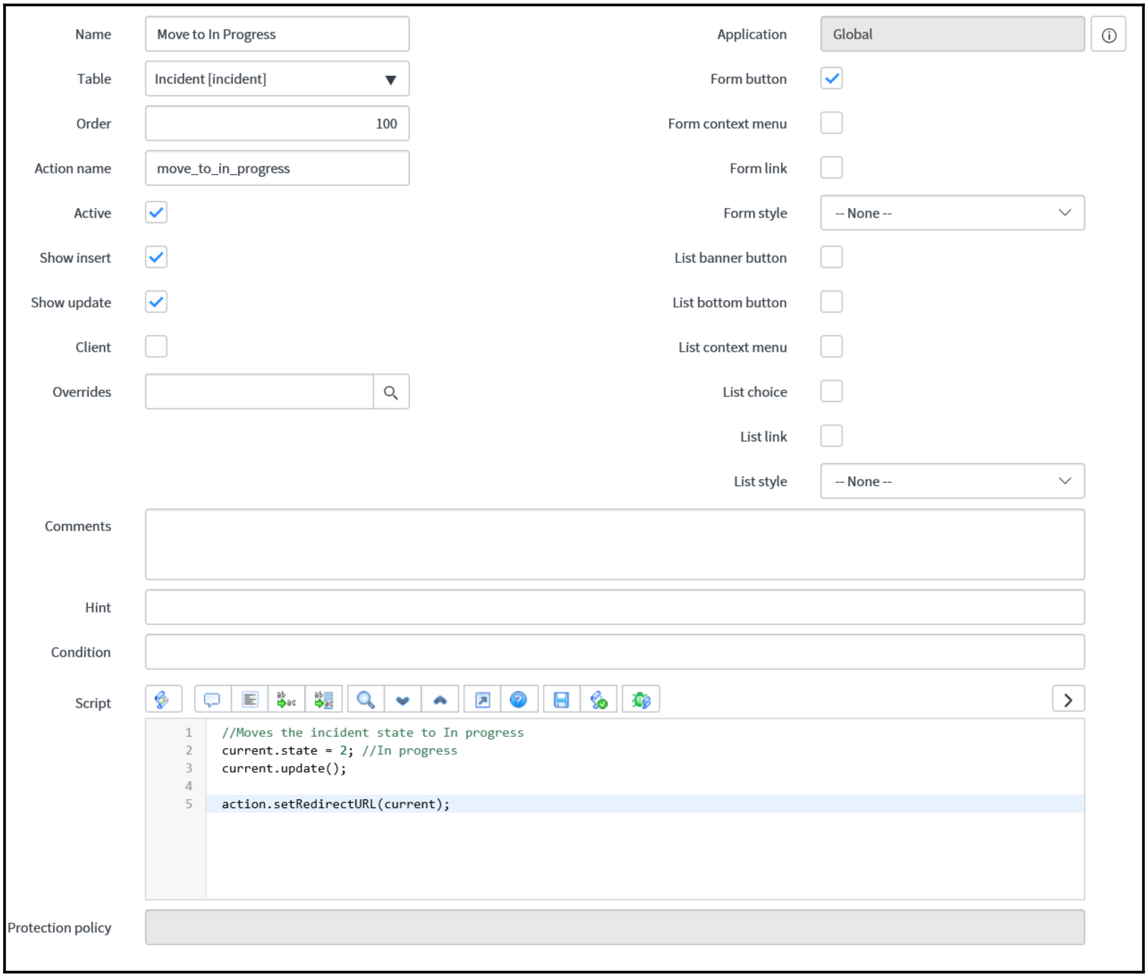
Active ☑

Advanced ☑

Condition: current.category == 'network'

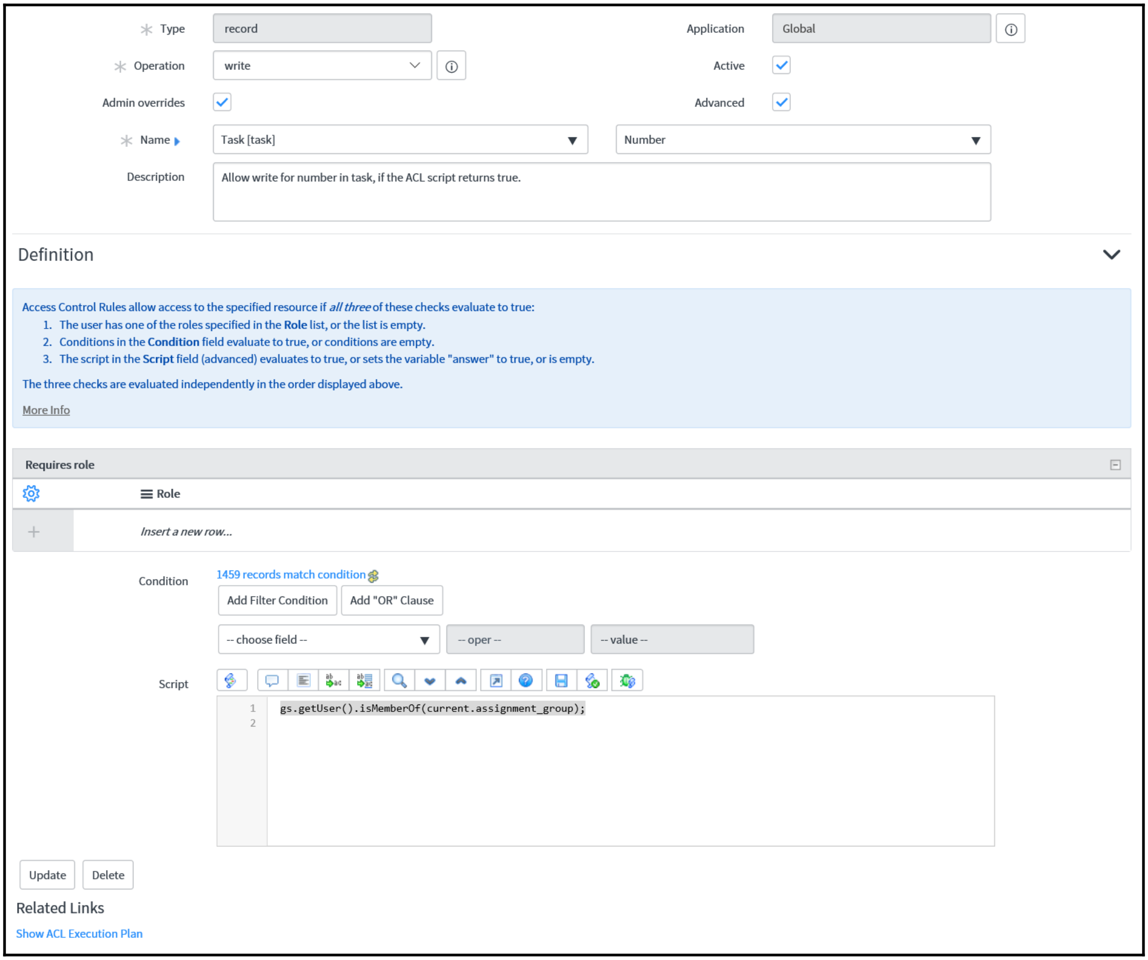
Script

```
(function executeRule(current, previous /*null when async*/) {

    current.impact = '1';

})(current, previous);
```



Name	Move to In Progress	Application	Global
Table	Incident [incident]	Form button	☑
Order	100	Form context menu	□
Action name	move_to_in_progress	Form link	□
Active	☑	Form style	-- None --
Show insert	☑	List banner button	□
Show update	☑	List bottom button	□
Client	□	List context menu	□
Overrides		List choice	□
		List link	□
		List style	-- None --
Comments			
Hint			
Condition			
Script			
Protection policy			

```
//Moves the incident state to In progress
current.state = 2; //In progress
current.update();

action.setRedirectURL(current);
```



Type	record	Application	Global
Operation	write	Active	☑
Admin overrides	☑	Advanced	☑
Name	Task [task]	Number	
Description	Allow write for number in task, if the ACL script returns true.		

Definition

Access Control Rules allow access to the specified resource if *all three* of these checks evaluate to true:

1. The user has one of the roles specified in the **Role** list, or the list is empty.
2. Conditions in the **Condition** field evaluate to true, or conditions are empty.
3. The script in the **Script** field (advanced) evaluates to true, or sets the variable "answer" to true, or is empty.

The three checks are evaluated independently in the order displayed above.

More Info

Requires role

Role
Insert a new row...

Condition: 1459 records match condition

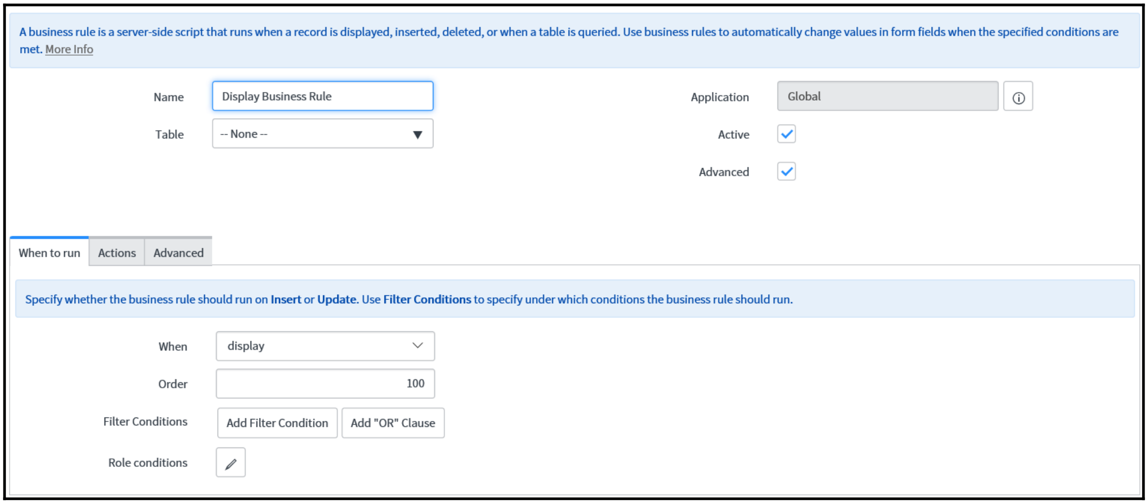
-- choose field -- | -- oper -- | -- value --

Script:

```
gs.getUser().isMemberOf(current.assignment_group);
```

Related Links

Show ACL Execution Plan

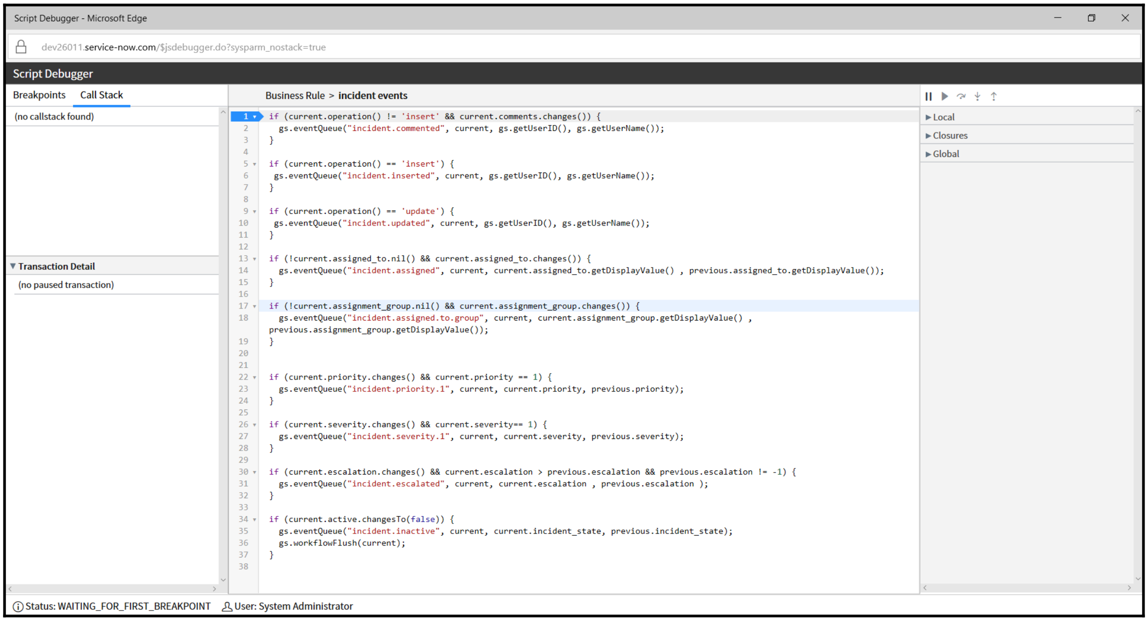


A business rule is a server-side script that runs when a record is displayed, inserted, deleted, or when a table is queried. Use business rules to automatically change values in form fields when the specified conditions are met. More Info

Name	Display Business Rule	Application	Global
Table	-- None --	Active	☑
		Advanced	☑

Specify whether the business rule should run on **Insert** or **Update**. Use **Filter Conditions** to specify under which conditions the business rule should run.

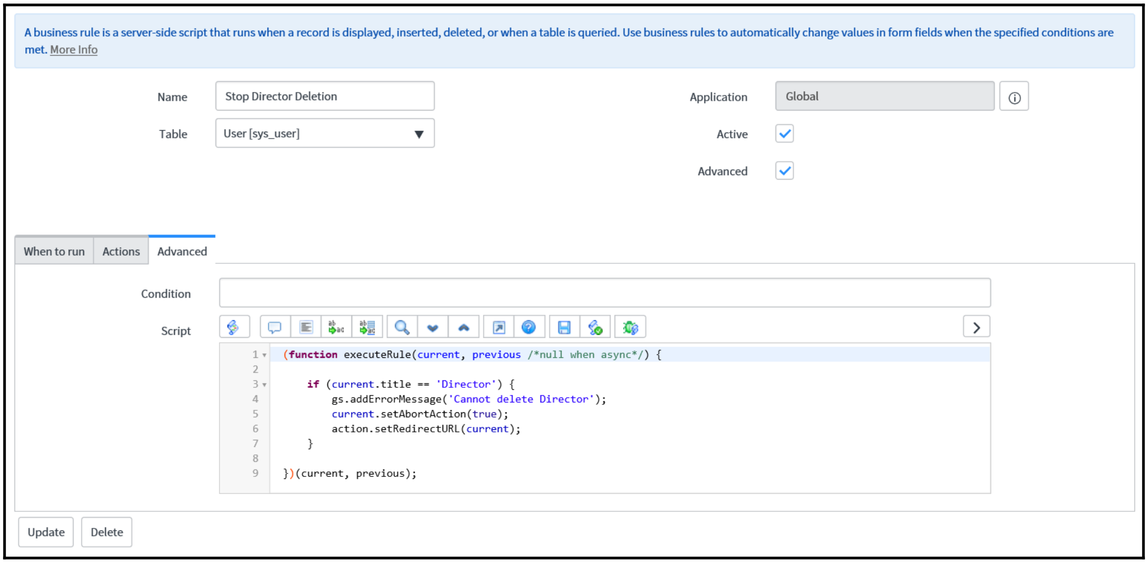
When	display
Order	100
Filter Conditions	Add Filter Condition Add "OR" Clause
Role conditions	



Script Debugger - Microsoft Edge
dev26011.service-now.com/$jsdebugger.do?sysparm_nostack=true
Script Debugger
Breakpoints
Call Stack
(no callstack found)
Transaction Detail
(no paused transaction)
Business Rule > incident events
Local
Closures
Global

```
if (current.operation() != 'insert' && current.comments.changes()) {
  gs.eventQueue("incident.commented", current, gs.getUserID(), gs.getUserName());
}

if (current.operation() == 'insert') {
 gs.eventQueue("incident.inserted", current, gs.getUserID(), gs.getUserName());
}

if (current.operation() == 'update') {
 gs.eventQueue("incident.updated", current, gs.getUserID(), gs.getUserName());
}

if (!current.assigned_to.nil() && current.assigned_to.changes()) {
  gs.eventQueue("incident.assigned", current, current.assigned_to.getDisplayValue() , previous.assigned_to.getDisplayValue());
}

if (!current.assignment_group.nil() && current.assignment_group.changes()) {
  gs.eventQueue("incident.assigned.to.group", current, current.assignment_group.getDisplayValue() ,
previous.assignment_group.getDisplayValue());
}


if (current.priority.changes() && current.priority == 1) {
  gs.eventQueue("incident.priority.1", current, current.priority, previous.priority);
}

if (current.severity.changes() && current.severity== 1) {
  gs.eventQueue("incident.severity.1", current, current.severity, previous.severity);
}

if (current.escalation.changes() && current.escalation > previous.escalation && previous.escalation != -1) {
  gs.eventQueue("incident.escalated", current, current.escalation , previous.escalation );
}

if (current.active.changesTo(false)) {
  gs.eventQueue("incident.inactive", current, current.incident_state, previous.incident_state);
  gs.workflowFlush(current);
}
```

Status: WAITING_FOR_FIRST_BREAKPOINT   User: System Administrator



A business rule is a server-side script that runs when a record is displayed, inserted, deleted, or when a table is queried. Use business rules to automatically change values in form fields when the specified conditions are met. More Info

Name: Stop Director Deletion

Table: User [sys_user]

Application: Global

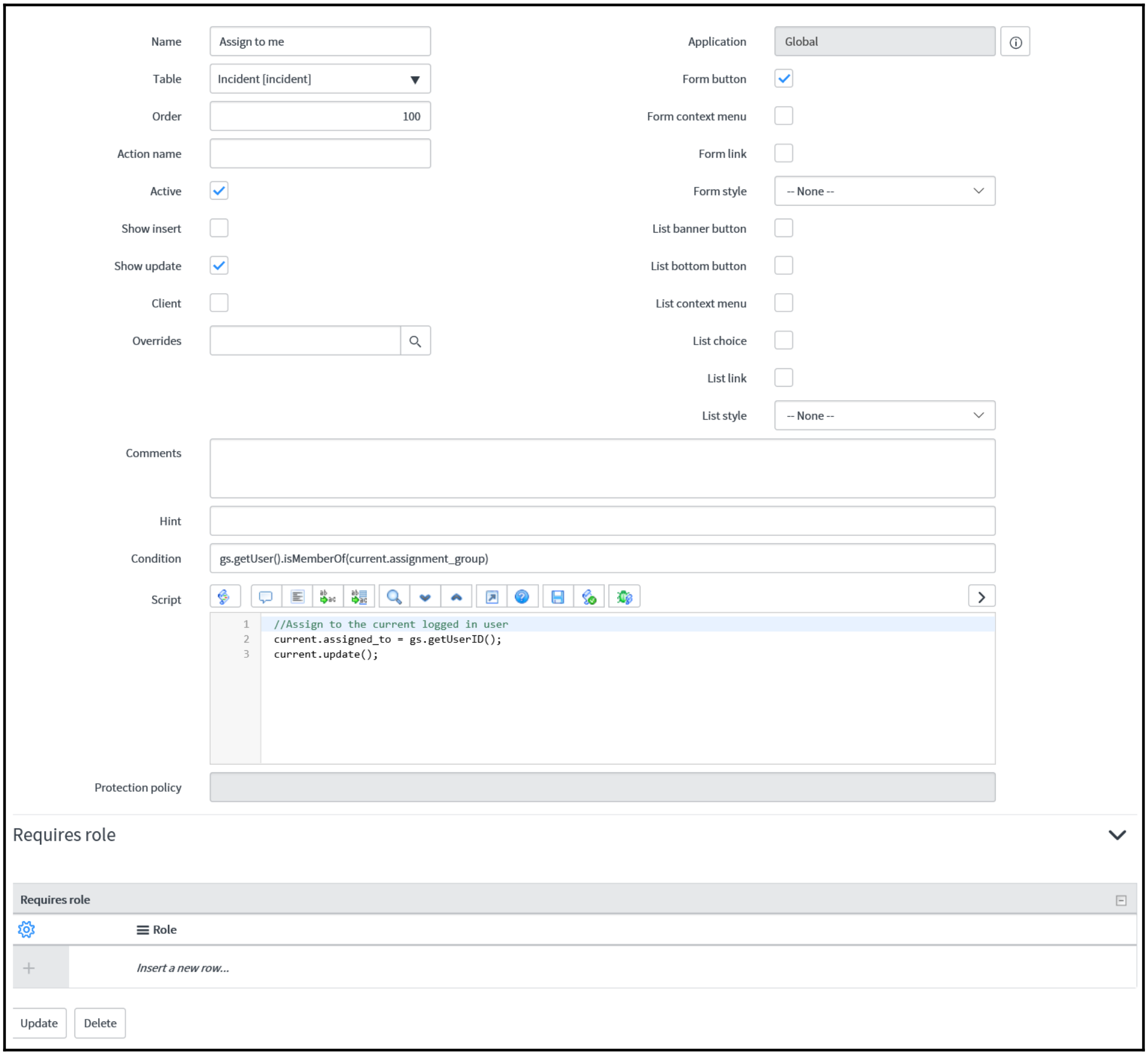
Active: ☑

Advanced: ☑

Condition:

Script:

```
(function executeRule(current, previous /*null when async*/) {

    if (current.title == 'Director') {
        gs.addErrorMessage('Cannot delete Director');
        current.setAbortAction(true);
        action.setRedirectURL(current);
    }

})(current, previous);
```



Name	Assign to me	Application	Global
Table	Incident [incident]	Form button	☑
Order	100	Form context menu	□
Action name		Form link	□
Active	☑	Form style	-- None --
Show insert	□	List banner button	□
Show update	☑	List bottom button	□
Client	□	List context menu	□
Overrides		List choice	□
		List link	□
		List style	-- None --

Comments

Hint

Condition: gs.getUser().isMemberOf(current.assignment_group)

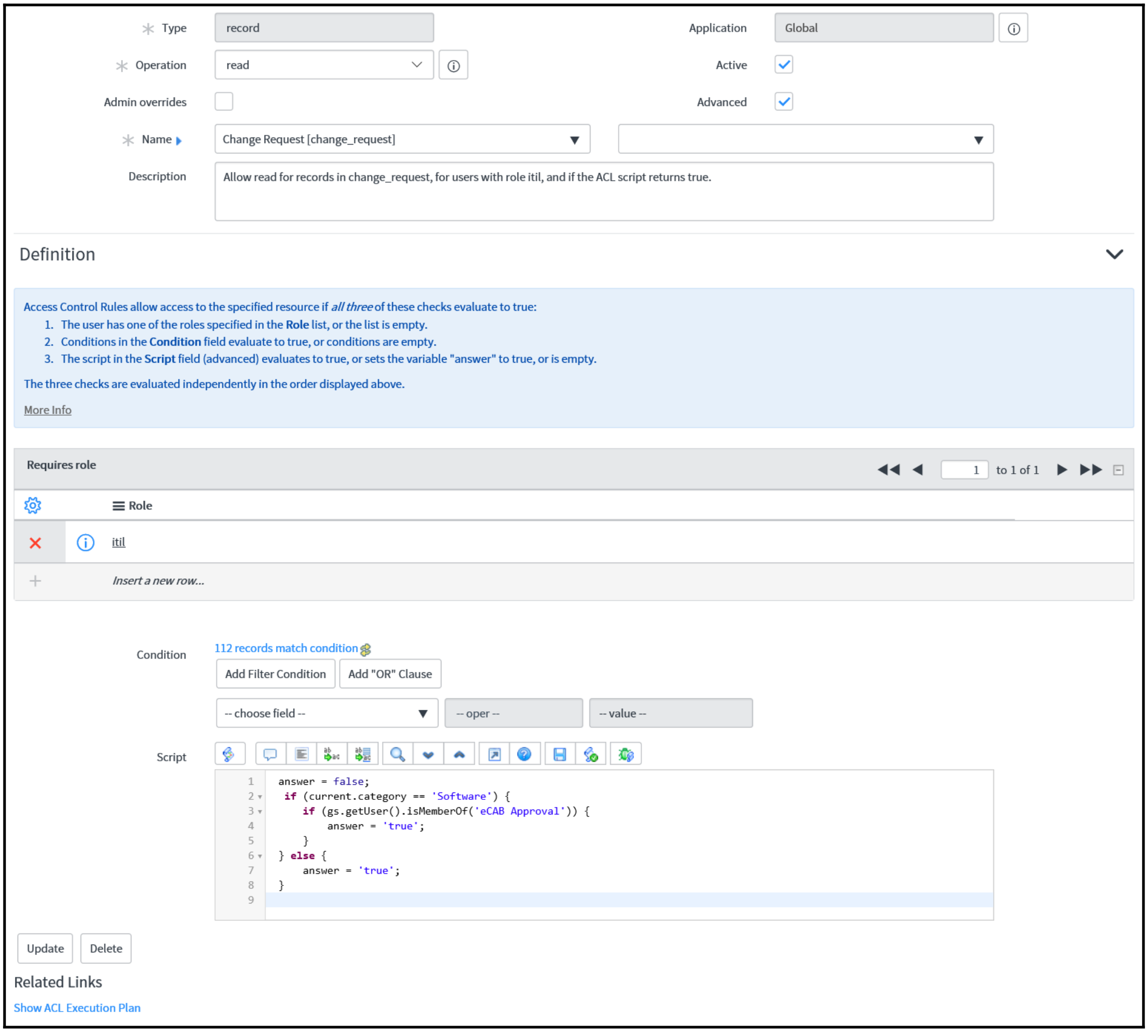
Script

```
//Assign to the current logged in user
current.assigned_to = gs.getUserID();
current.update();
```

Protection policy

Requires role

Requires role
Role
Insert a new row...



Type	record	Application	Global
Operation	read	Active	☑
Admin overrides	□	Advanced	☑
Name	Change Request [change_request]		
Description	Allow read for records in change_request, for users with role itil, and if the ACL script returns true.		

Definition

Access Control Rules allow access to the specified resource if *all three* of these checks evaluate to true:

1. The user has one of the roles specified in the **Role** list, or the list is empty.
2. Conditions in the **Condition** field evaluate to true, or conditions are empty.
3. The script in the **Script** field (advanced) evaluates to true, or sets the variable "answer" to true, or is empty.

The three checks are evaluated independently in the order displayed above.

More Info

Requires role — 1 to 1 of 1

Role
itil
Insert a new row...

Condition: 112 records match condition

Add Filter Condition | Add "OR" Clause

-- choose field -- | -- oper -- | -- value --

Script

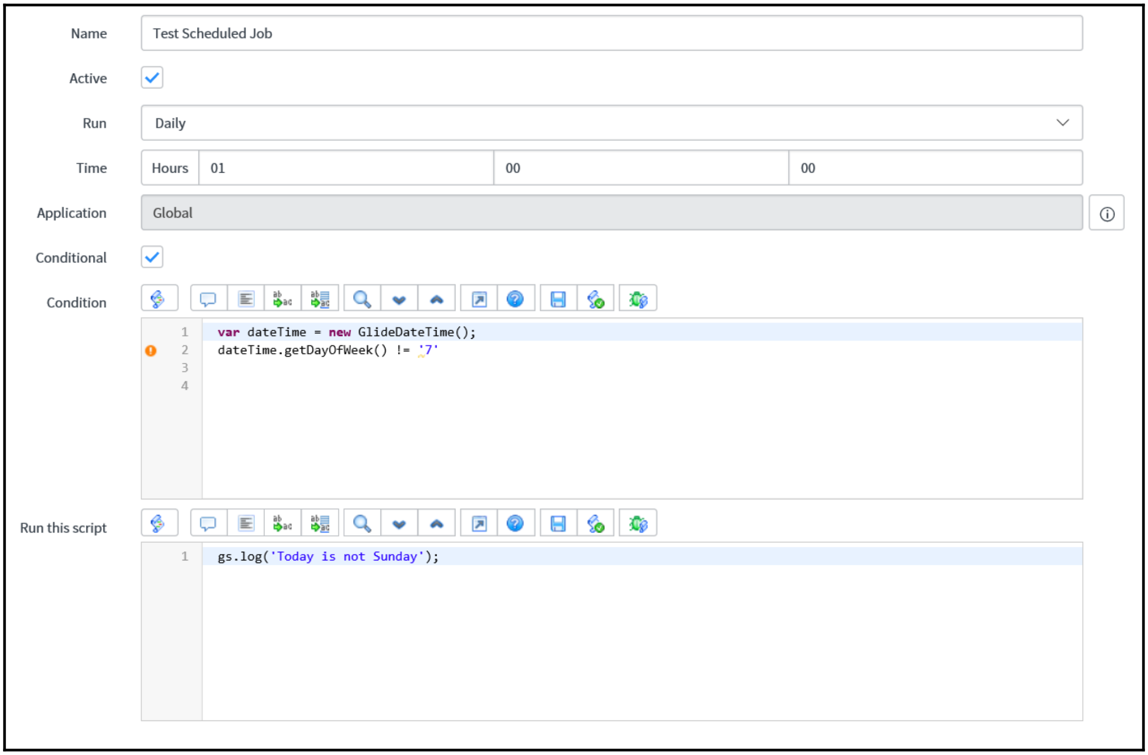
```
answer = false;
 if (current.category == 'Software') {
    if (gs.getUser().isMemberOf('eCAB Approval')) {
        answer = 'true';
    }
} else {
    answer = 'true';
}
```

Update | Delete

Related Links

Show ACL Execution Plan

Chapter 6: Advanced Server-Side Scripting



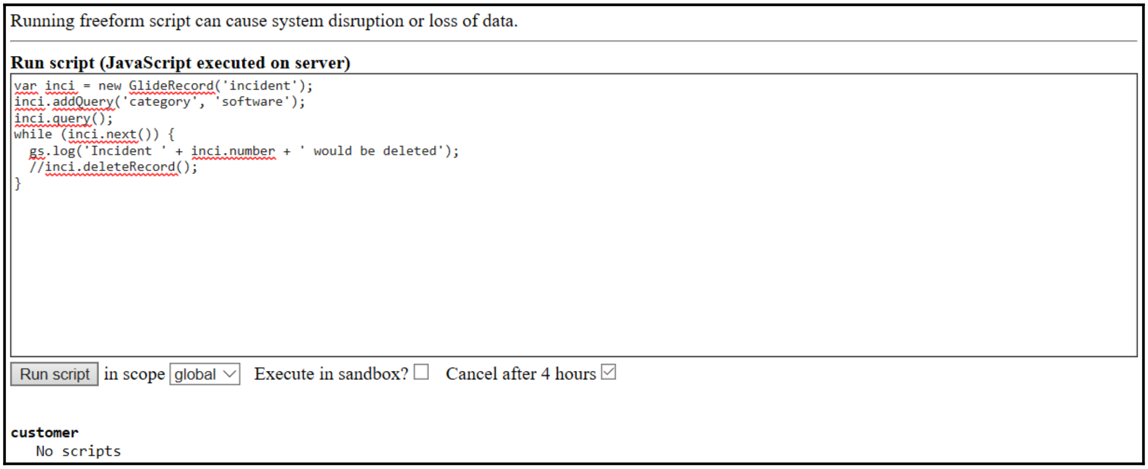
Name	Test Scheduled Job
Active	☑
Run	Daily
Time	Hours 01 00 00
Application	Global
Conditional	☑

Condition

```
var dateTime = new GlideDateTime();
dateTime.getDayOfWeek() != '7'
```

Run this script

```
gs.log('Today is not Sunday');
```



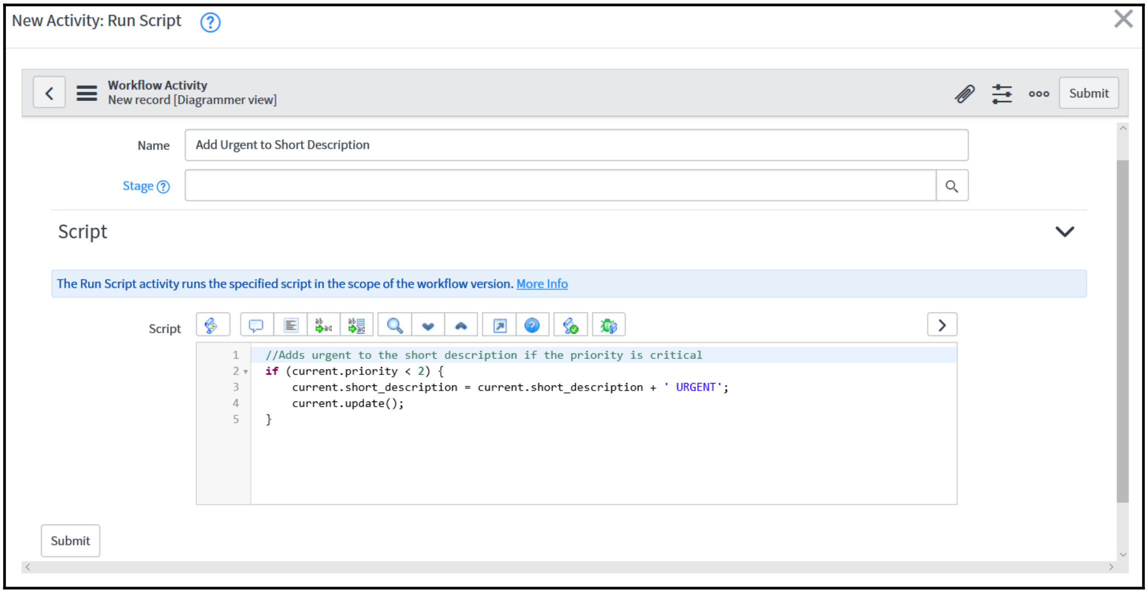
Running freeform script can cause system disruption or loss of data.

Run script (JavaScript executed on server)

```
var inci = new GlideRecord('incident');
inci.addQuery('category', 'software');
inci.query();
while (inci.next()) {
  gs.log('Incident ' + inci.number + ' would be deleted');
  //inci.deleteRecord();
}
```

in scope global Execute in sandbox? □ Cancel after 4 hours ☑

```
customer
   No scripts
```



Name: Add Urgent to Short Description

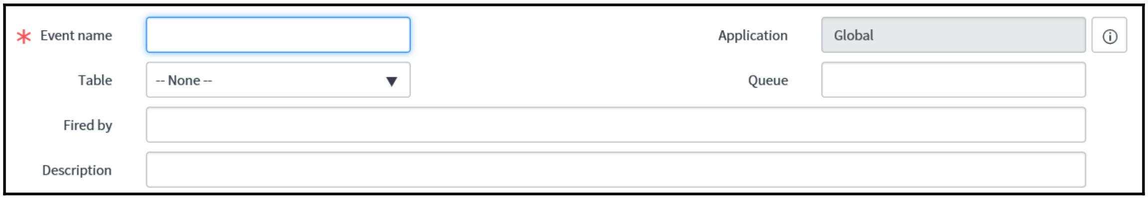
Stage:

Script

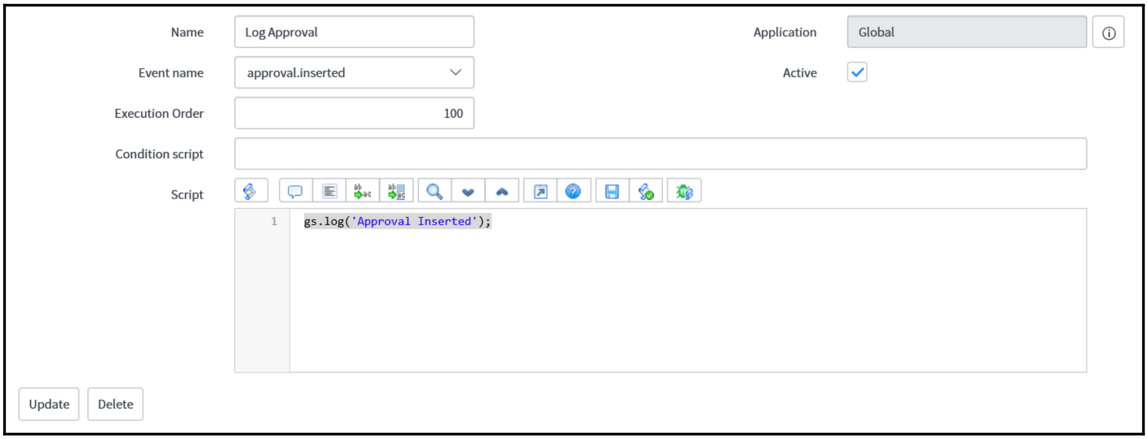
The Run Script activity runs the specified script in the scope of the workflow version. More Info

Script:

```
//Adds urgent to the short description if the priority is critical
if (current.priority < 2) {
    current.short_description = current.short_description + ' URGENT';
    current.update();
}
```



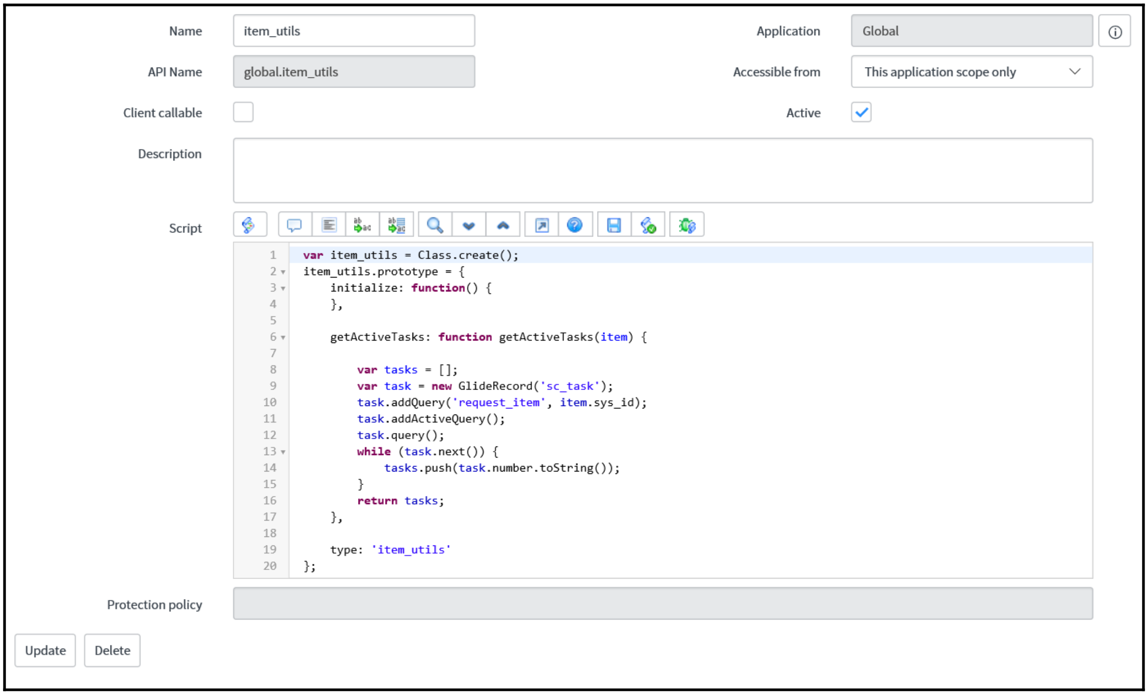
* Event name		Application	Global
Table	-- None --	Queue	
Fired by			
Description			



Name	Log Approval	Application	Global
Event name	approval.inserted	Active	☑
Execution Order	100		
Condition script			

Script:

```
gs.log('Approval Inserted');
```

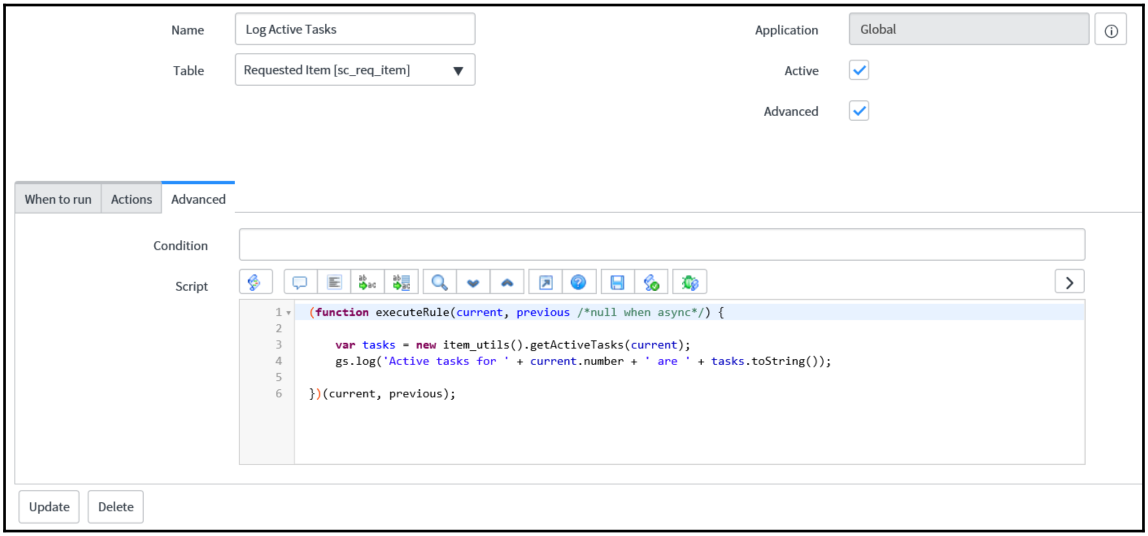


Name	item_utils	Application	Global
API Name	global.item_utils	Accessible from	This application scope only
Client callable	□	Active	☑
Description			

Script

```
var item_utils = Class.create();
item_utils.prototype = {
    initialize: function() {
    },

    getActiveTasks: function getActiveTasks(item) {

        var tasks = [];
        var task = new GlideRecord('sc_task');
        task.addQuery('request_item', item.sys_id);
        task.addActiveQuery();
        task.query();
        while (task.next()) {
            tasks.push(task.number.toString());
        }
        return tasks;
    },

    type: 'item_utils'
};
```

Protection policy

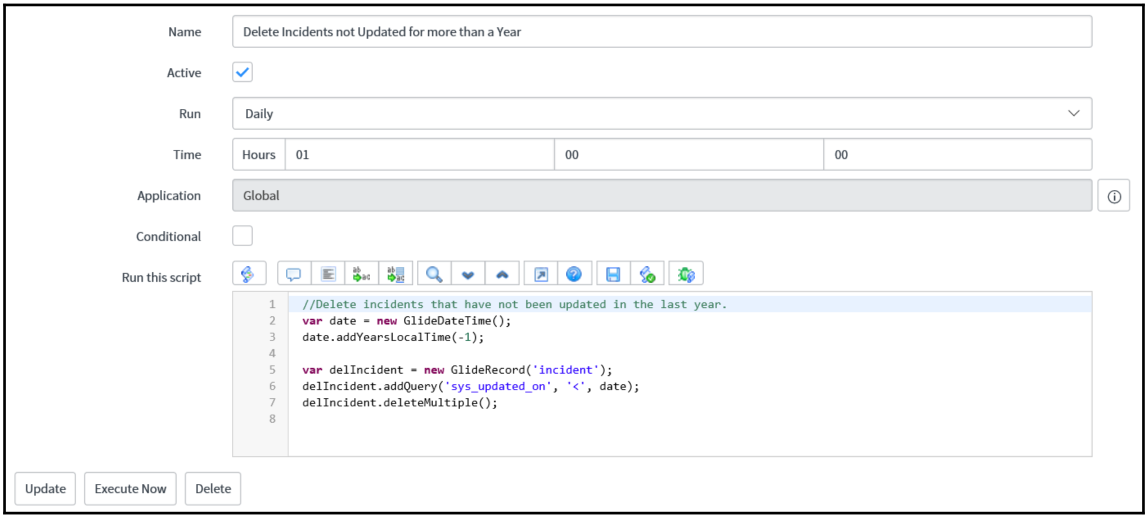


Name	Log Active Tasks	Application	Global
Table	Requested Item [sc_req_item]	Active	☑
		Advanced	☑

Condition

Script

```
(function executeRule(current, previous /*null when async*/) {

    var tasks = new item_utils().getActiveTasks(current);
    gs.log('Active tasks for ' + current.number + ' are ' + tasks.toString());

})(current, previous);
```



Name: Delete Incidents not Updated for more than a Year

Active: ☑

Run: Daily

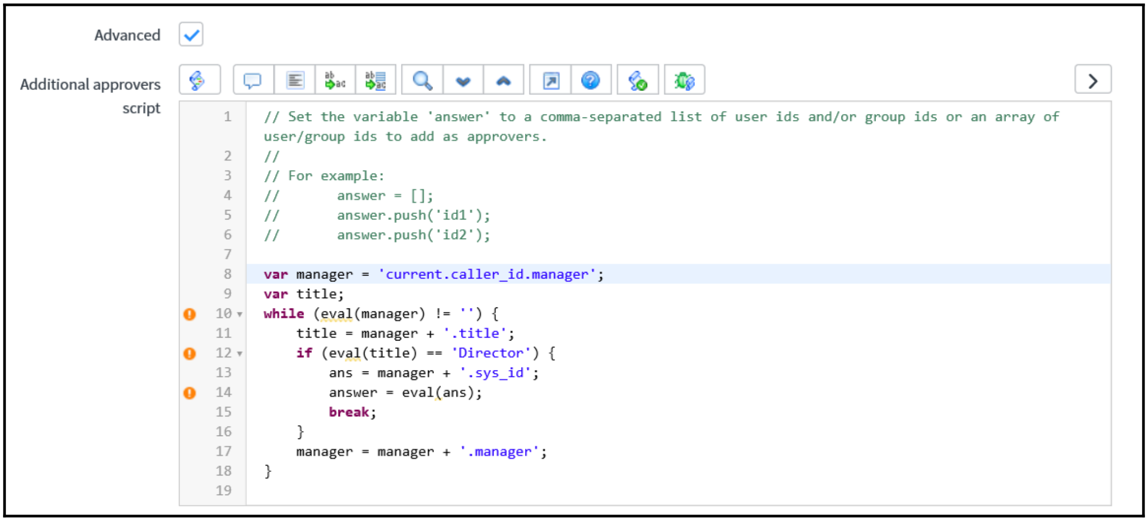
Time: Hours 01 | 00 | 00

Application: Global

Conditional: □

Run this script:

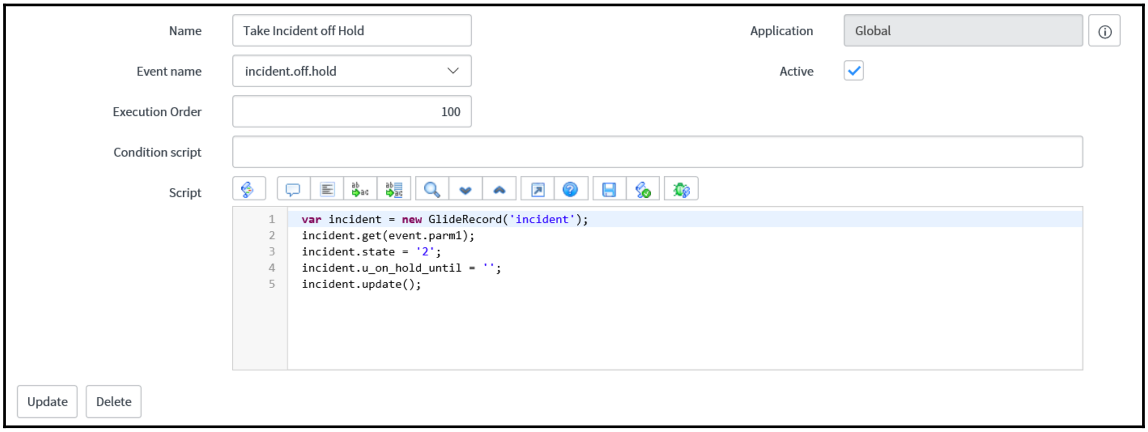
```
//Delete incidents that have not been updated in the last year.
var date = new GlideDateTime();
date.addYearsLocalTime(-1);

var delIncident = new GlideRecord('incident');
delIncident.addQuery('sys_updated_on', '<', date);
delIncident.deleteMultiple();

```



Advanced: ☑

Additional approvers script:

```
// Set the variable 'answer' to a comma-separated list of user ids and/or group ids or an array of user/group ids to add as approvers.
//
// For example:
//       answer = [];
//       answer.push('id1');
//       answer.push('id2');

var manager = 'current.caller_id.manager';
var title;
while (eval(manager) != '') {
    title = manager + '.title';
    if (eval(title) == 'Director') {
        ans = manager + '.sys_id';
        answer = eval(ans);
        break;
    }
    manager = manager + '.manager';
}

```

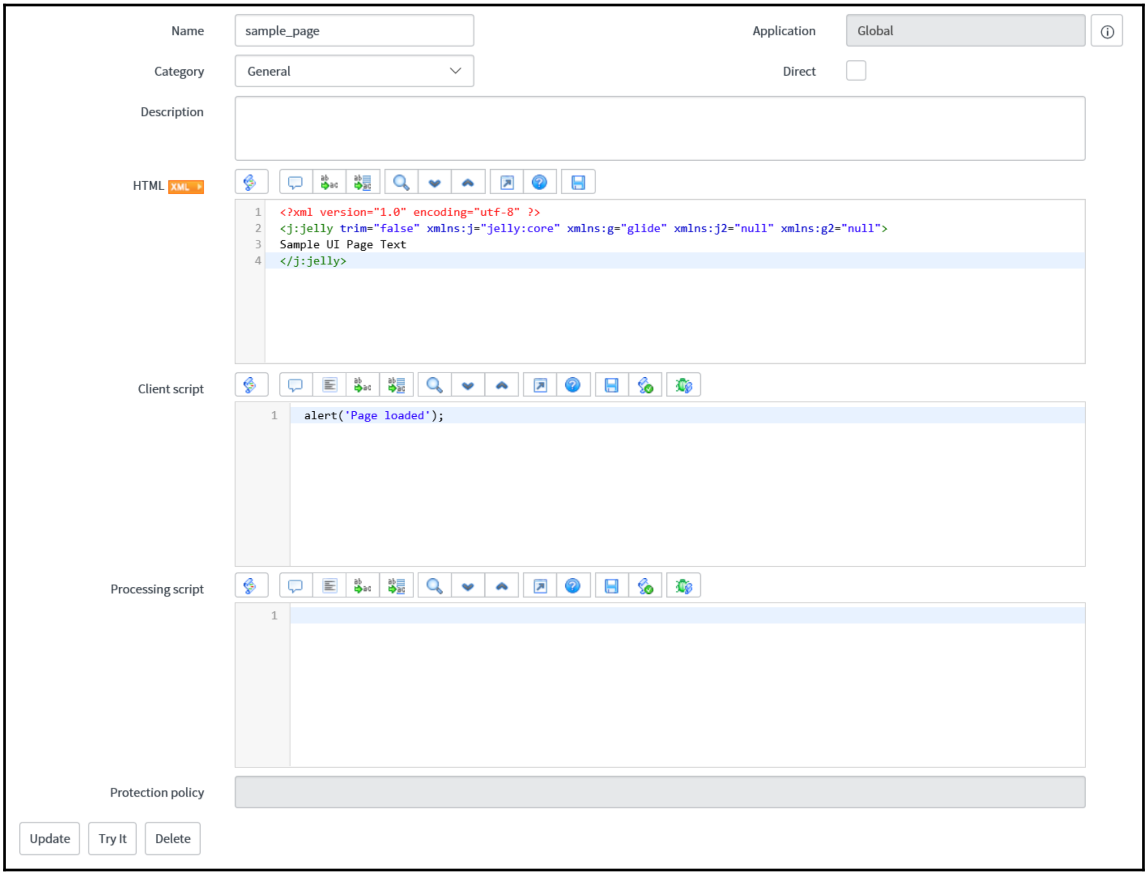


Name	Take Incident off Hold
Event name	incident.off.hold
Execution Order	100
Condition script	
Application	Global
Active	☑

Script

```
var incident = new GlideRecord('incident');
incident.get(event.parm1);
incident.state = '2';
incident.u_on_hold_until = '';
incident.update();
```

Chapter 7: Introduction to Custom Pages



Name	sample_page	Application	Global
Category	General	Direct	□
Description			

HTML XML

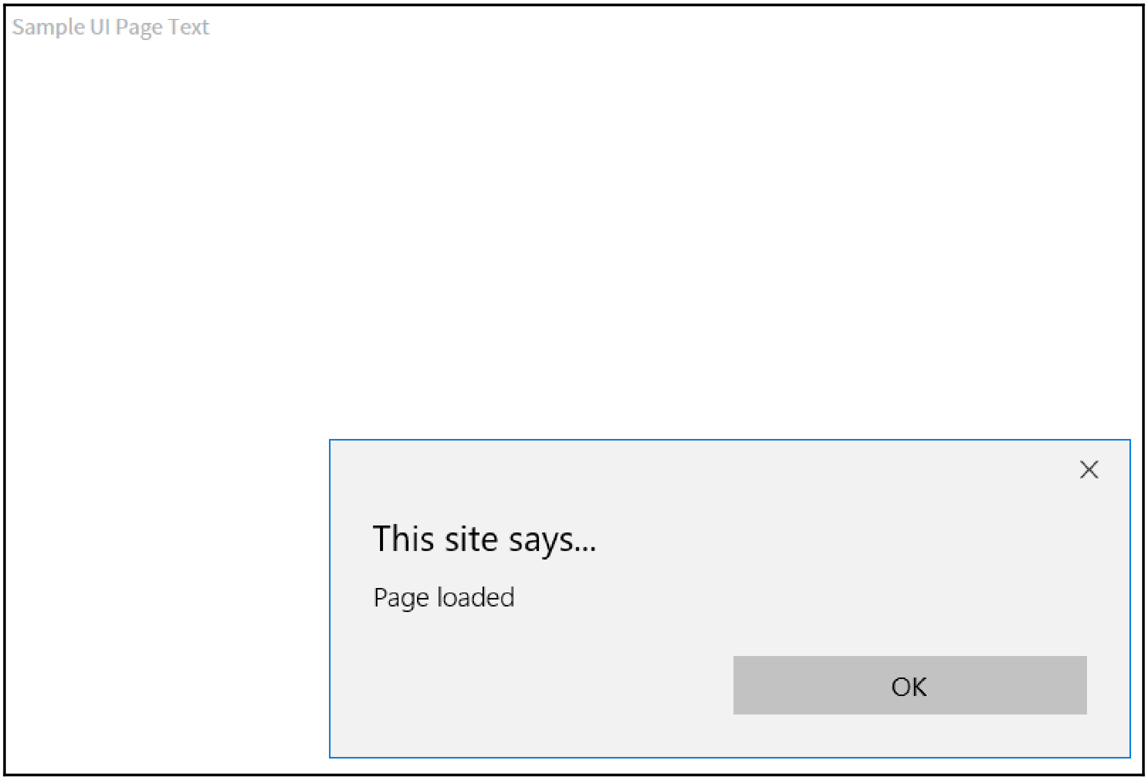
```
<?xml version="1.0" encoding="utf-8" ?>
<j:jelly trim="false" xmlns:j="jelly:core" xmlns:g="glide" xmlns:j2="null" xmlns:g2="null">
Sample UI Page Text
</j:jelly>
```

Client script

```
alert('Page loaded');
```

Processing script

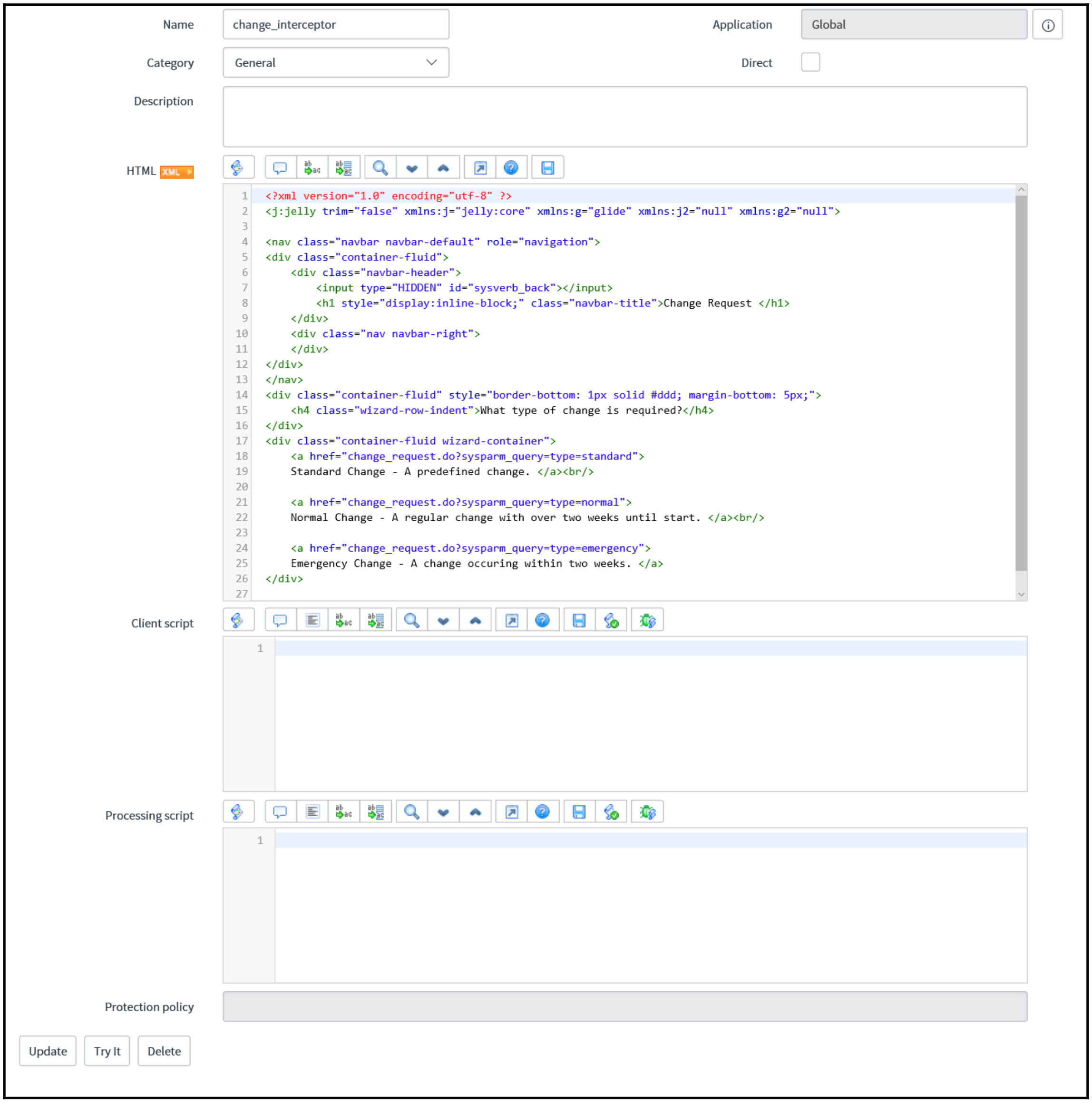
Protection policy



Sample UI Page Text

This site says...

Page loaded



Name	change_interceptor	Application	Global
Category	General	Direct	□
Description			

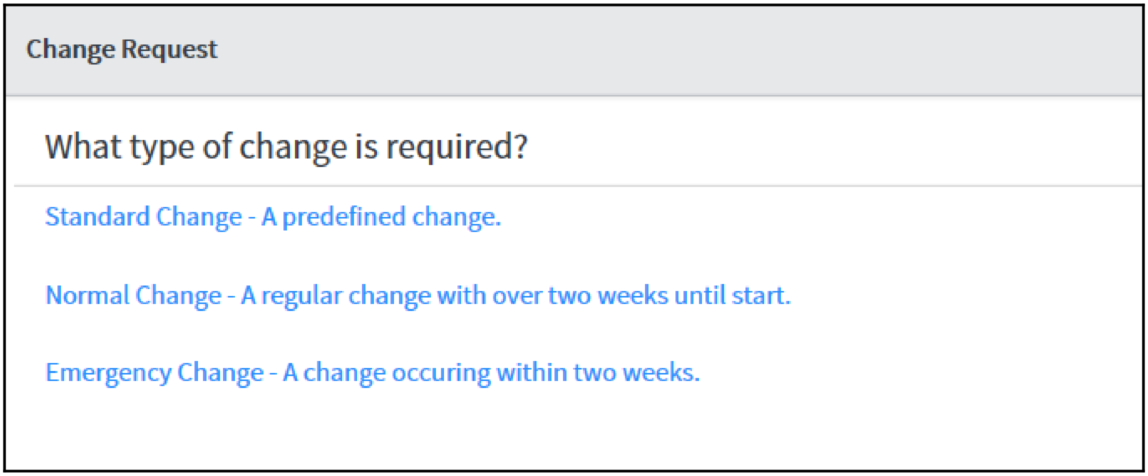
HTML

```
<?xml version="1.0" encoding="utf-8" ?>
<j:jelly trim="false" xmlns:j="jelly:core" xmlns:g="glide" xmlns:j2="null" xmlns:g2="null">

<nav class="navbar navbar-default" role="navigation">
<div class="container-fluid">
    <div class="navbar-header">
        <input type="HIDDEN" id="sysverb_back"></input>
        <h1 style="display:inline-block;" class="navbar-title">Change Request </h1>
    </div>
    <div class="nav navbar-right">
    </div>
</div>
</nav>
<div class="container-fluid" style="border-bottom: 1px solid #ddd; margin-bottom: 5px;">
    <h4 class="wizard-row-indent">What type of change is required?</h4>
</div>
<div class="container-fluid wizard-container">
    <a href="change_request.do?sysparm_query=type=standard">
    Standard Change - A predefined change. </a><br/>

    <a href="change_request.do?sysparm_query=type=normal">
    Normal Change - A regular change with over two weeks until start. </a><br/>

    <a href="change_request.do?sysparm_query=type=emergency">
    Emergency Change - A change occuring within two weeks. </a>
</div>

```

Client script

Processing script

Protection policy



Change Request

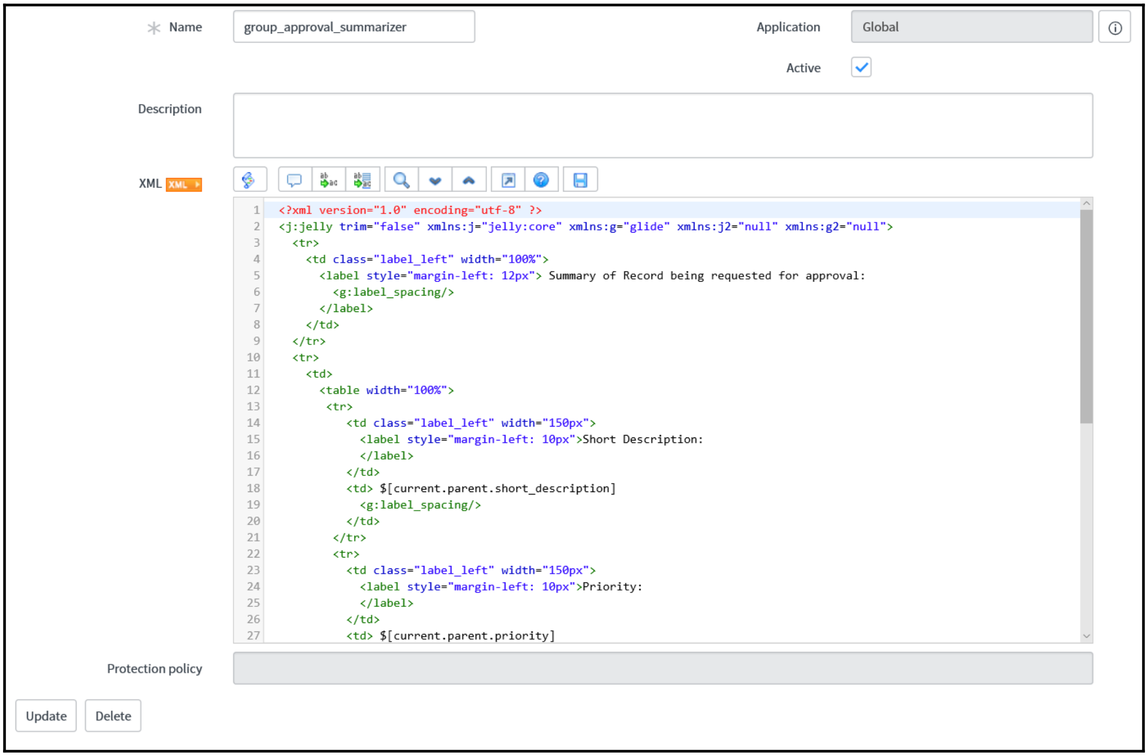
What type of change is required?

Standard Change - A predefined change.

Normal Change - A regular change with over two weeks until start.

Emergency Change - A change occuring within two weeks.

Chapter 8: Scripting with Jelly

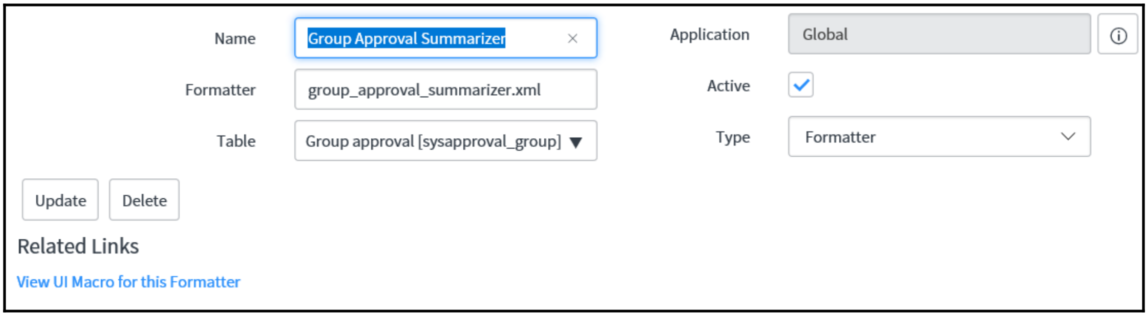


Name	group_approval_summarizer	Application	Global
		Active	☑
Description			

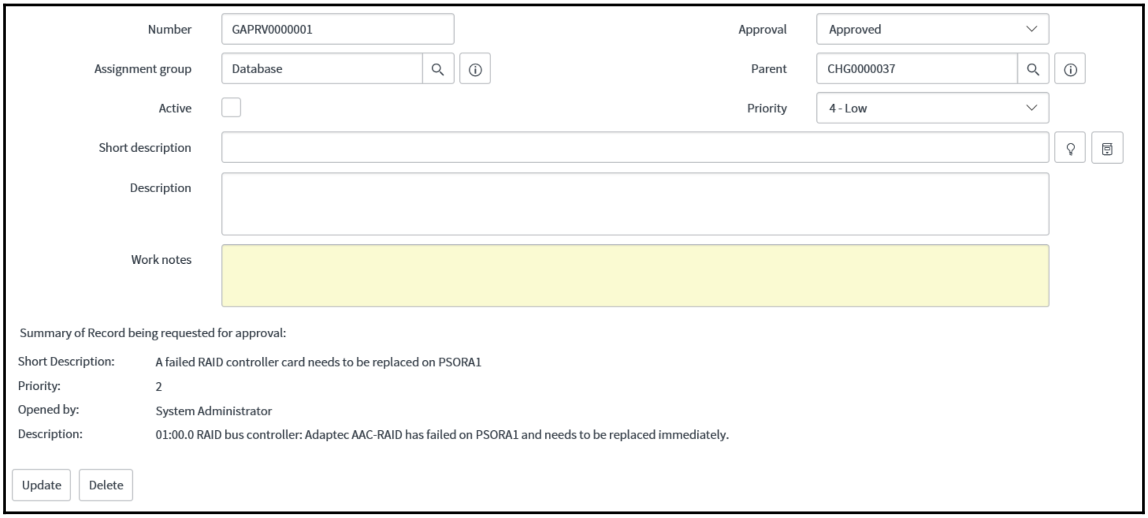
XML

```
<?xml version="1.0" encoding="utf-8" ?>
<j:jelly trim="false" xmlns:j="jelly:core" xmlns:g="glide" xmlns:j2="null" xmlns:g2="null">
  <tr>
    <td class="label_left" width="100%">
      <label style="margin-left: 12px"> Summary of Record being requested for approval:
        <g:label_spacing/>
      </label>
    </td>
  </tr>
  <tr>
    <td>
      <table width="100%">
       <tr>
          <td class="label_left" width="150px">
            <label style="margin-left: 10px">Short Description:
            </label>
          </td>
          <td> $[current.parent.short_description]
            <g:label_spacing/>
          </td>
        </tr>
        <tr>
          <td class="label_left" width="150px">
            <label style="margin-left: 10px">Priority:
            </label>
          </td>
          <td> $[current.parent.priority]
```

Protection policy



Name	Group Approval Summarizer	Application	Global
Formatter	group_approval_summarizer.xml	Active	☑
Table	Group approval [sysapproval_group]	Type	Formatter

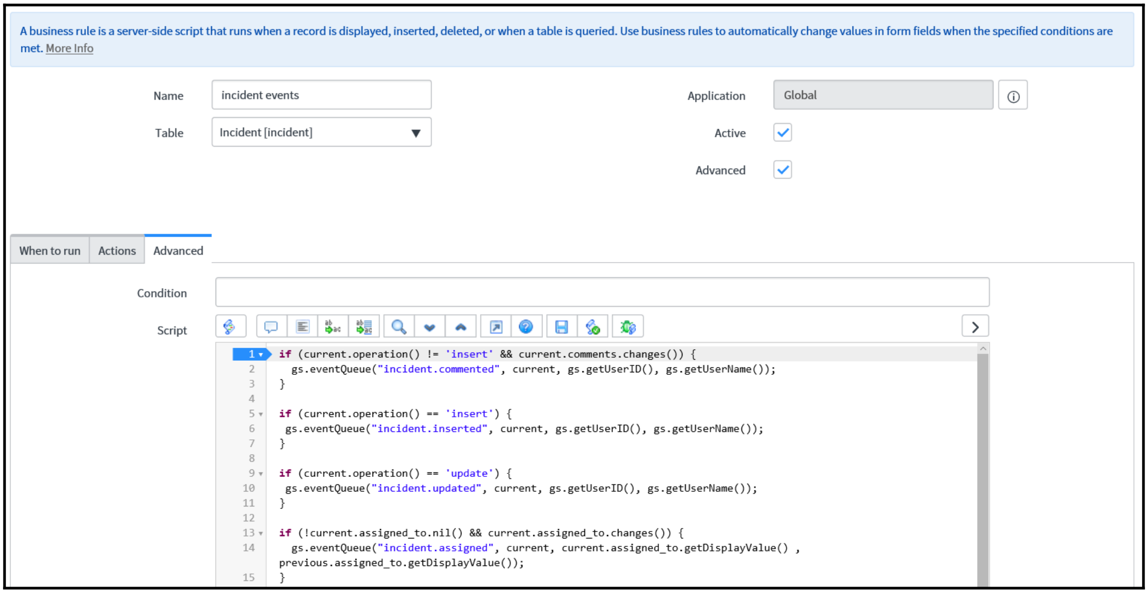


Number	GAPRV0000001	Approval	Approved
Assignment group	Database	Parent	CHG0000037
Active	□	Priority	4 - Low
Short description			
Description			
Work notes			

Summary of Record being requested for approval:

Short Description:	A failed RAID controller card needs to be replaced on PSORA1
Priority:	2
Opened by:	System Administrator
Description:	01:00.0 RAID bus controller: Adaptec AAC-RAID has failed on PSORA1 and needs to be replaced immediately.

Chapter 9: Debugging the Script



A business rule is a server-side script that runs when a record is displayed, inserted, deleted, or when a table is queried. Use business rules to automatically change values in form fields when the specified conditions are met. More Info

Name: incident events

Table: Incident [incident]

Application: Global

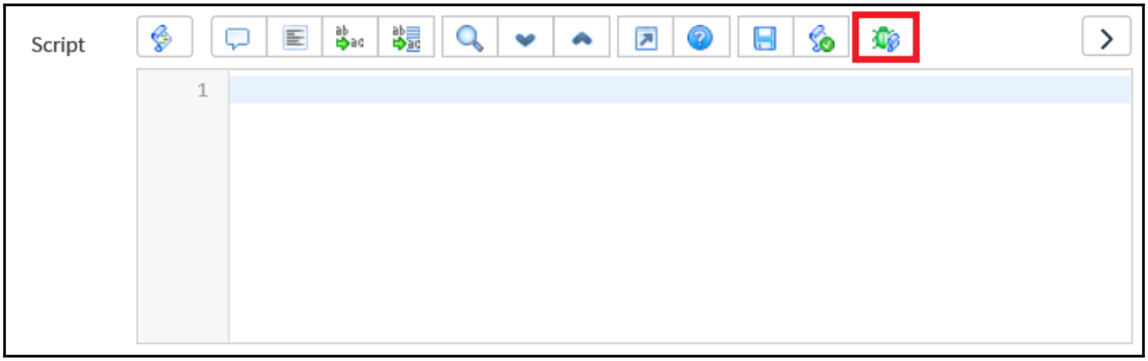
Active: ☑

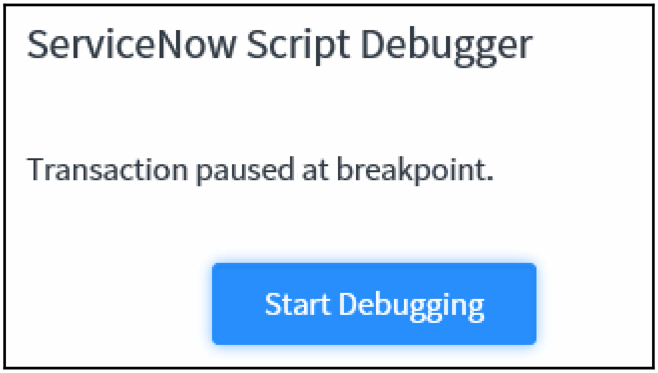
Advanced: ☑

Condition:

Script:

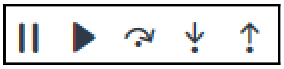
```
if (current.operation() != 'insert' && current.comments.changes()) {
  gs.eventQueue("incident.commented", current, gs.getUserID(), gs.getUserName());
}

if (current.operation() == 'insert') {
 gs.eventQueue("incident.inserted", current, gs.getUserID(), gs.getUserName());
}

if (current.operation() == 'update') {
 gs.eventQueue("incident.updated", current, gs.getUserID(), gs.getUserName());
}

if (!current.assigned_to.nil() && current.assigned_to.changes()) {
  gs.eventQueue("incident.assigned", current, current.assigned_to.getDisplayValue() ,
previous.assigned_to.getDisplayValue());
}
```

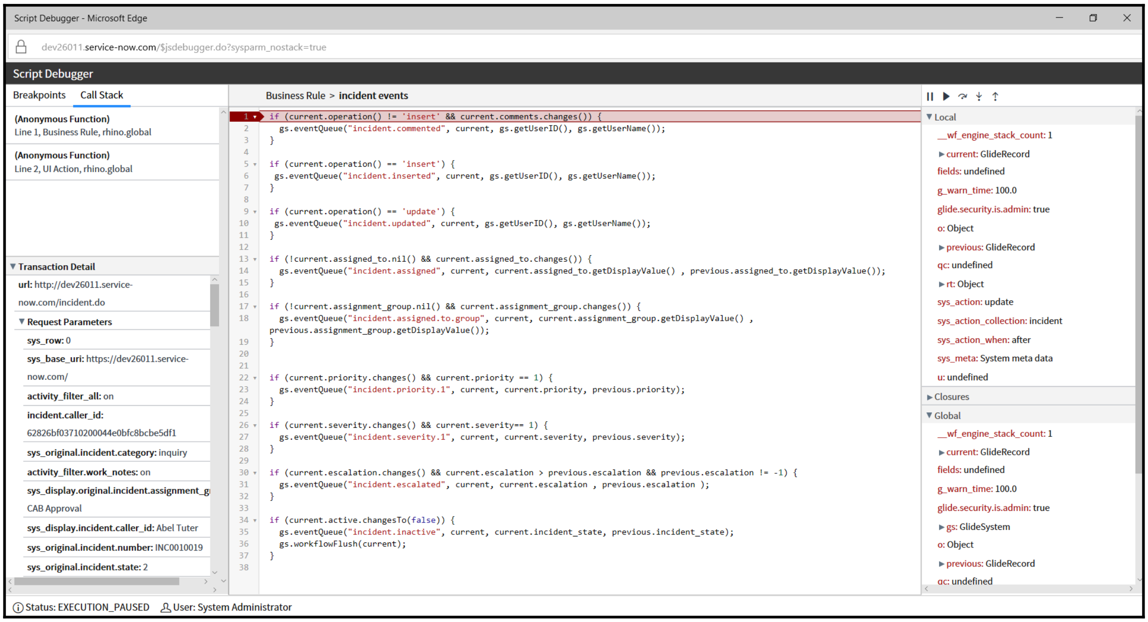




ServiceNow Script Debugger

Transaction paused at breakpoint.





Script Debugger - Microsoft Edge
dev26011.service-now.com/$jsdebugger.do?sysparm_nostack=true
Script Debugger
Breakpoints
Call Stack
Business Rule > incident events

(Anonymous Function)
Line 1, Business Rule, rhino.global

(Anonymous Function)
Line 2, UI Action, rhino.global

▼ Transaction Detail

url: http://dev26011.service-now.com/incident.do

▼ Request Parameters

- **sys_row:** 0
- **sys_base_uri:** https://dev26011.service-now.com/
- **activity_filter_all:** on
- **incident.caller_id:** 62826bf03710200044e0bfc8bcbe5df1
- **sys_original.incident.category:** inquiry
- **activity_filter.work_notes:** on
- **sys_display.original.incident.assignment_g** CAB Approval
- **sys_display.incident.caller_id:** Abel Tuter
- **sys_original.incident.number:** INC0010019
- **sys_original.incident.state:** 2

```
if (current.operation() != 'insert' && current.comments.changes()) {
  gs.eventQueue("incident.commented", current, gs.getUserID(), gs.getUserName());
}

if (current.operation() == 'insert') {
 gs.eventQueue("incident.inserted", current, gs.getUserID(), gs.getUserName());
}

if (current.operation() == 'update') {
 gs.eventQueue("incident.updated", current, gs.getUserID(), gs.getUserName());
}

if (!current.assigned_to.nil() && current.assigned_to.changes()) {
  gs.eventQueue("incident.assigned", current, current.assigned_to.getDisplayValue() , previous.assigned_to.getDisplayValue());
}

if (!current.assignment_group.nil() && current.assignment_group.changes()) {
  gs.eventQueue("incident.assigned.to.group", current, current.assignment_group.getDisplayValue() ,
previous.assignment_group.getDisplayValue());
}


if (current.priority.changes() && current.priority == 1) {
  gs.eventQueue("incident.priority.1", current, current.priority, previous.priority);
}

if (current.severity.changes() && current.severity== 1) {
  gs.eventQueue("incident.severity.1", current, current.severity, previous.severity);
}

if (current.escalation.changes() && current.escalation > previous.escalation && previous.escalation != -1) {
  gs.eventQueue("incident.escalated", current, current.escalation , previous.escalation );
}

if (current.active.changesTo(false)) {
  gs.eventQueue("incident.inactive", current, current.incident_state, previous.incident_state);
  gs.workflowFlush(current);
}

```

▼ Local

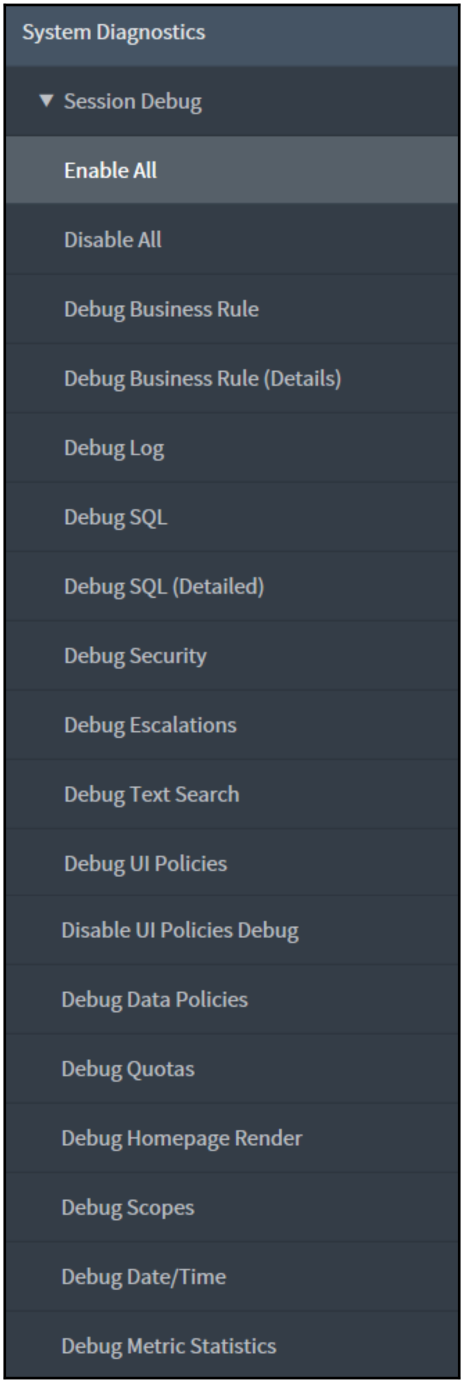
- __wf_engine_stack_count: 1
- ▶ current: GlideRecord
- fields: undefined
- g_warn_time: 100.0
- glide.security.is.admin: true
- o: Object
- ▶ previous: GlideRecord
- qc: undefined
- ▶ rt: Object
- sys_action: update
- sys_action_collection: incident
- sys_action_when: after
- sys_meta: System meta data
- u: undefined

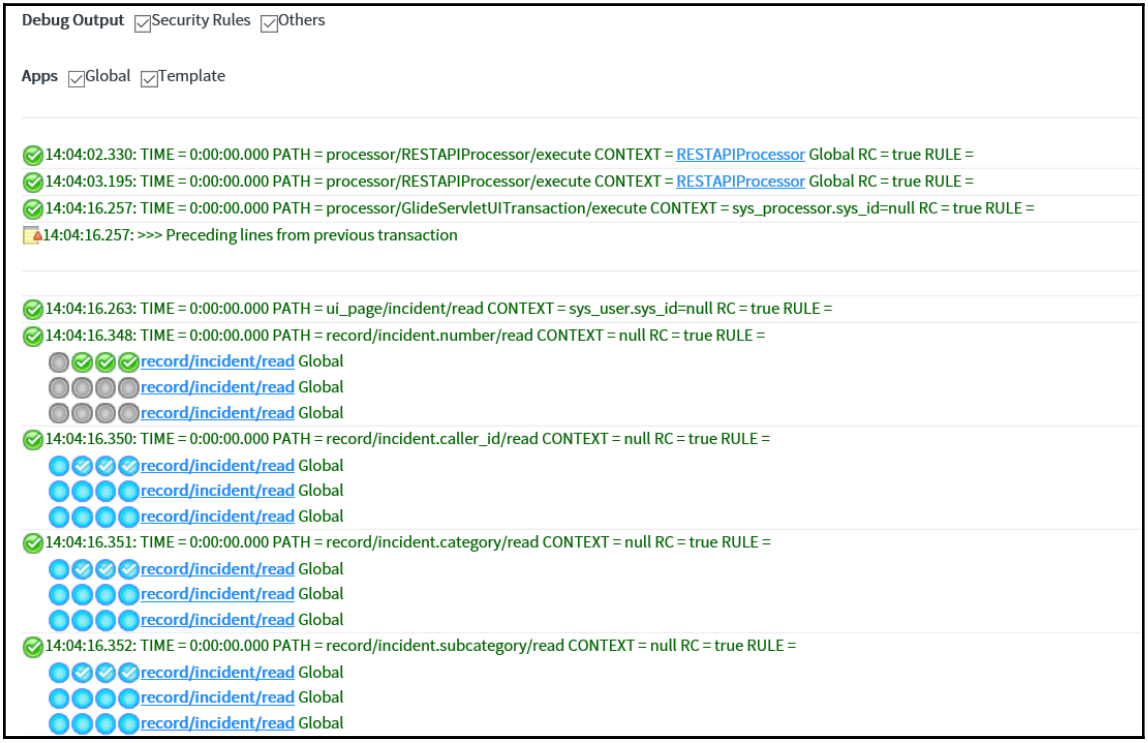
▶ Closures

▼ Global

- __wf_engine_stack_count: 1
- ▶ current: GlideRecord
- fields: undefined
- g_warn_time: 100.0
- glide.security.is.admin: true
- ▶ gs: GlideSystem
- o: Object
- ▶ previous: GlideRecord
- qc: undefined

Status: EXECUTION_PAUSED User: System Administrator

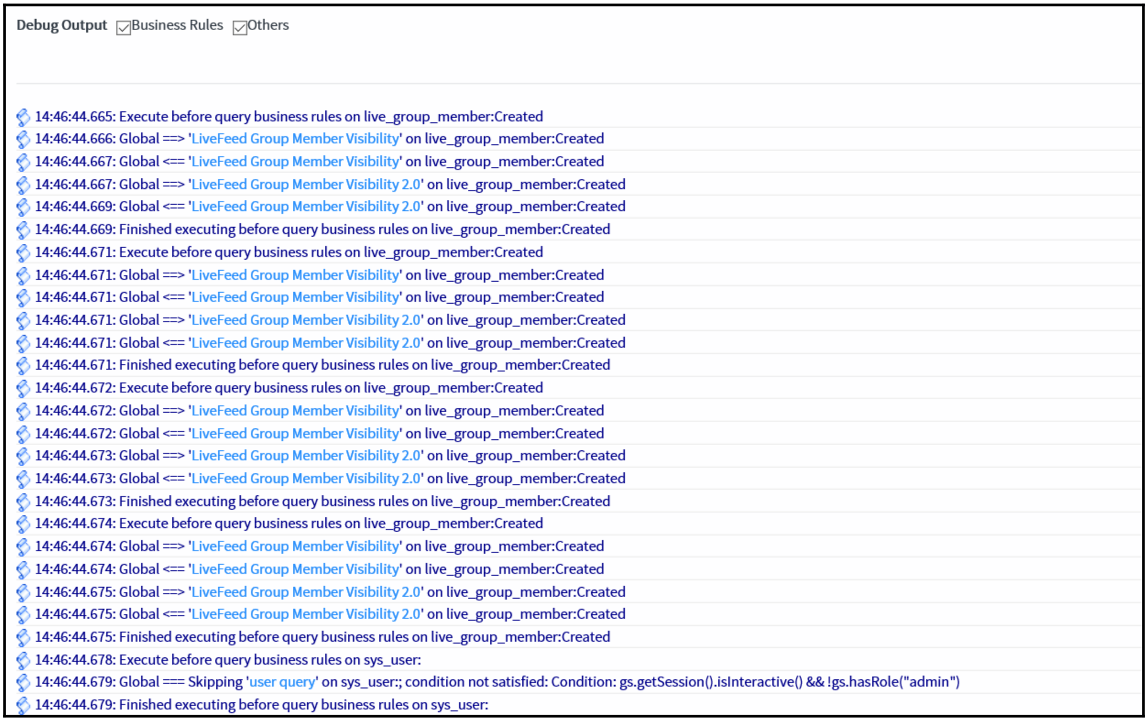

System Diagnostics
▼ Session Debug
Enable All
Disable All
Debug Business Rule
Debug Business Rule (Details)
Debug Log
Debug SQL
Debug SQL (Detailed)
Debug Security
Debug Escalations
Debug Text Search
Debug UI Policies
Disable UI Policies Debug
Debug Data Policies
Debug Quotas
Debug Homepage Render
Debug Scopes
Debug Date/Time
Debug Metric Statistics



Debug Output ☑Security Rules ☑Others

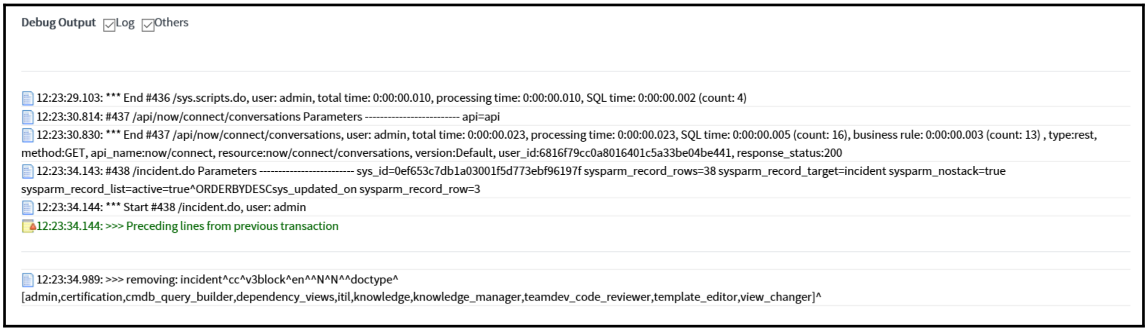
Apps ☑Global ☑Template

```
✓14:04:02.330: TIME = 0:00:00.000 PATH = processor/RESTAPIProcessor/execute CONTEXT = RESTAPIProcessor Global RC = true RULE =
✓14:04:03.195: TIME = 0:00:00.000 PATH = processor/RESTAPIProcessor/execute CONTEXT = RESTAPIProcessor Global RC = true RULE =
✓14:04:16.257: TIME = 0:00:00.000 PATH = processor/GlideServletUITransaction/execute CONTEXT = sys_processor.sys_id=null RC = true RULE =
 14:04:16.257: >>> Preceding lines from previous transaction

✓14:04:16.263: TIME = 0:00:00.000 PATH = ui_page/incident/read CONTEXT = sys_user.sys_id=null RC = true RULE =
✓14:04:16.348: TIME = 0:00:00.000 PATH = record/incident.number/read CONTEXT = null RC = true RULE =
    ●✓✓✓record/incident/read Global
    ●●●●record/incident/read Global
    ●●●●record/incident/read Global
✓14:04:16.350: TIME = 0:00:00.000 PATH = record/incident.caller_id/read CONTEXT = null RC = true RULE =
    ●✓✓✓record/incident/read Global
    ●●●●record/incident/read Global
    ●●●●record/incident/read Global
✓14:04:16.351: TIME = 0:00:00.000 PATH = record/incident.category/read CONTEXT = null RC = true RULE =
    ●✓✓✓record/incident/read Global
    ●●●●record/incident/read Global
    ●●●●record/incident/read Global
✓14:04:16.352: TIME = 0:00:00.000 PATH = record/incident.subcategory/read CONTEXT = null RC = true RULE =
    ●✓✓✓record/incident/read Global
    ●●●●record/incident/read Global
    ●●●●record/incident/read Global
```



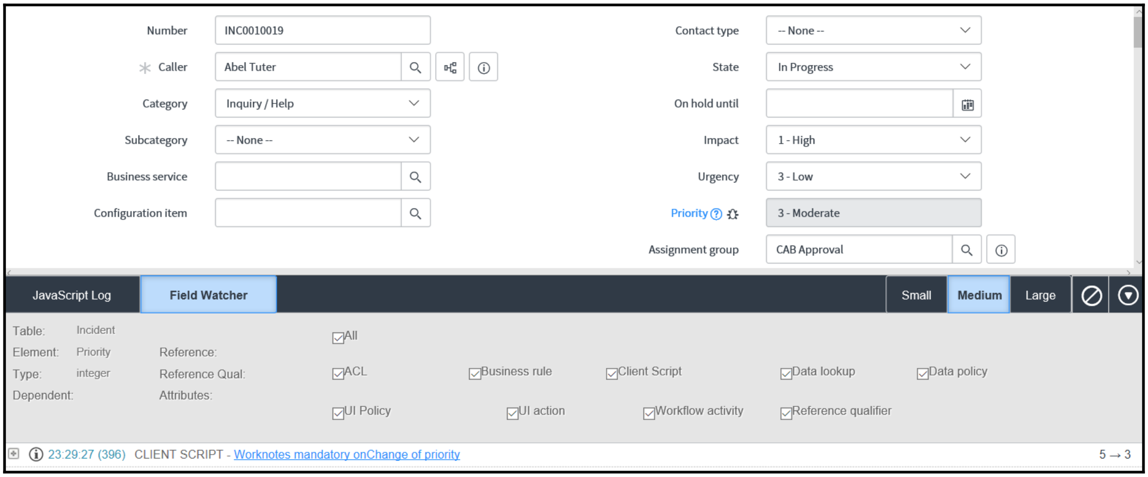
Debug Output ☑Business Rules ☑Others

```
14:46:44.665: Execute before query business rules on live_group_member:Created
14:46:44.666: Global ==> 'LiveFeed Group Member Visibility' on live_group_member:Created
14:46:44.667: Global <== 'LiveFeed Group Member Visibility' on live_group_member:Created
14:46:44.667: Global ==> 'LiveFeed Group Member Visibility 2.0' on live_group_member:Created
14:46:44.669: Global <== 'LiveFeed Group Member Visibility 2.0' on live_group_member:Created
14:46:44.669: Finished executing before query business rules on live_group_member:Created
14:46:44.671: Execute before query business rules on live_group_member:Created
14:46:44.671: Global ==> 'LiveFeed Group Member Visibility' on live_group_member:Created
14:46:44.671: Global <== 'LiveFeed Group Member Visibility' on live_group_member:Created
14:46:44.671: Global ==> 'LiveFeed Group Member Visibility 2.0' on live_group_member:Created
14:46:44.671: Global <== 'LiveFeed Group Member Visibility 2.0' on live_group_member:Created
14:46:44.671: Finished executing before query business rules on live_group_member:Created
14:46:44.672: Execute before query business rules on live_group_member:Created
14:46:44.672: Global ==> 'LiveFeed Group Member Visibility' on live_group_member:Created
14:46:44.672: Global <== 'LiveFeed Group Member Visibility' on live_group_member:Created
14:46:44.673: Global ==> 'LiveFeed Group Member Visibility 2.0' on live_group_member:Created
14:46:44.673: Global <== 'LiveFeed Group Member Visibility 2.0' on live_group_member:Created
14:46:44.673: Finished executing before query business rules on live_group_member:Created
14:46:44.674: Execute before query business rules on live_group_member:Created
14:46:44.674: Global ==> 'LiveFeed Group Member Visibility' on live_group_member:Created
14:46:44.674: Global <== 'LiveFeed Group Member Visibility' on live_group_member:Created
14:46:44.675: Global ==> 'LiveFeed Group Member Visibility 2.0' on live_group_member:Created
14:46:44.675: Global <== 'LiveFeed Group Member Visibility 2.0' on live_group_member:Created
14:46:44.675: Finished executing before query business rules on live_group_member:Created
14:46:44.678: Execute before query business rules on sys_user:
14:46:44.679: Global === Skipping 'user query' on sys_user:; condition not satisfied: Condition: gs.getSession().isInteractive() && !gs.hasRole("admin")
14:46:44.679: Finished executing before query business rules on sys_user:
```



Debug Output ☑Log ☑Others

```
12:23:29.103: *** End #436 /sys.scripts.do, user: admin, total time: 0:00:00.010, processing time: 0:00:00.010, SQL time: 0:00:00.002 (count: 4)
12:23:30.814: #437 /api/now/connect/conversations Parameters ------------------------ api=api
12:23:30.830: *** End #437 /api/now/connect/conversations, user: admin, total time: 0:00:00.023, processing time: 0:00:00.023, SQL time: 0:00:00.005 (count: 16), business rule: 0:00:00.003 (count: 13) , type:rest,
method:GET, api_name:now/connect, resource:now/connect/conversations, version:Default, user_id:6816f79cc0a8016401c5a33be04be441, response_status:200
12:23:34.143: #438 /incident.do Parameters ------------------------ sys_id=0ef653c7db1a03001f5d773ebf96197f sysparm_record_rows=38 sysparm_record_target=incident sysparm_nostack=true
sysparm_record_list=active=true^ORDERBYDESCsys_updated_on sysparm_record_row=3
12:23:34.144: *** Start #438 /incident.do, user: admin
12:23:34.144: >>> Preceding lines from previous transaction

12:23:34.989: >>> removing: incident^cc^v3block^en^^N^N^^doctype^
[admin,certification,cmdb_query_builder,dependency_views,itil,knowledge,knowledge_manager,teamdev_code_reviewer,template_editor,view_changer]^
```

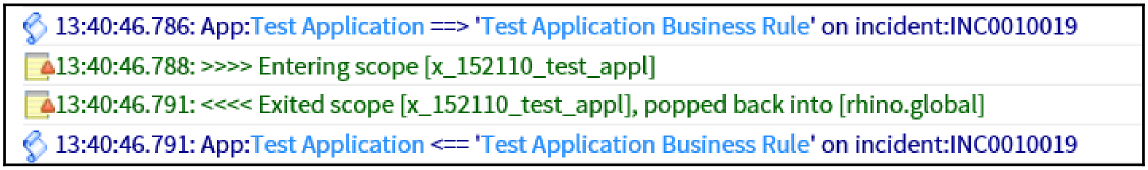


Number	INC0010019	Contact type	-- None --
Caller	Abel Tuter	State	In Progress
Category	Inquiry / Help	On hold until	
Subcategory	-- None --	Impact	1 - High
Business service		Urgency	3 - Low
Configuration item		Priority	3 - Moderate
		Assignment group	CAB Approval

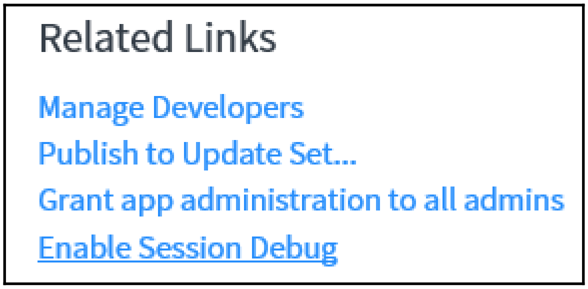
Table: Incident
Element: Priority Reference:
Type: integer Reference Qual:
Dependent: Attributes:

☑All
☑ACL ☑Business rule ☑Client Script ☑Data lookup ☑Data policy
☑UI Policy ☑UI action ☑Workflow activity ☑Reference qualifier

```
23:29:27 (396)  CLIENT SCRIPT - Worknotes mandatory onChange of priority    5 → 3
```

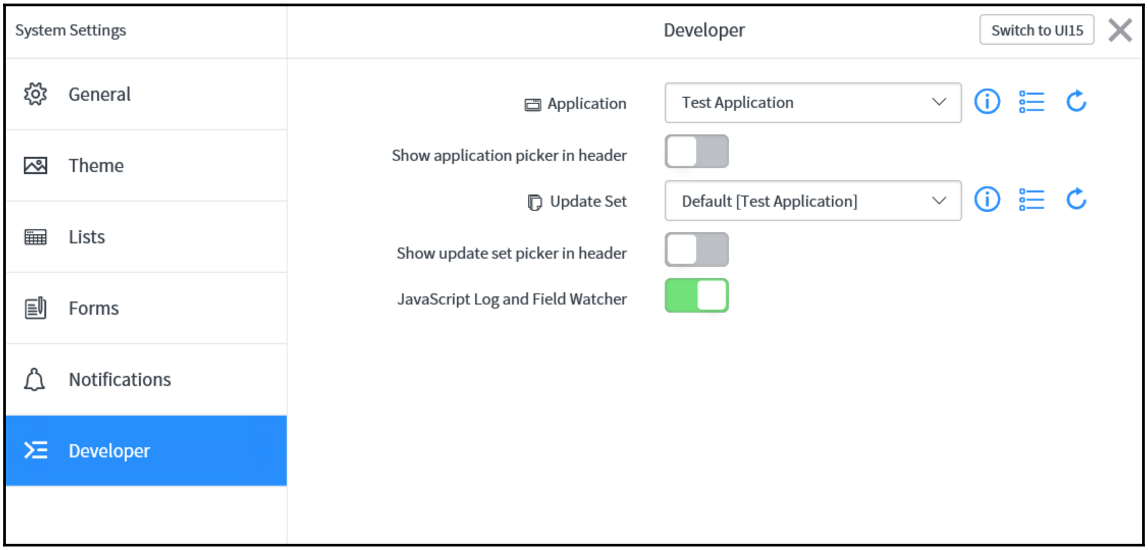


```
13:40:46.786: App:Test Application ==> 'Test Application Business Rule' on incident:INC0010019
13:40:46.788: >>>> Entering scope [x_152110_test_appl]
13:40:46.791: <<<< Exited scope [x_152110_test_appl], popped back into [rhino.global]
13:40:46.791: App:Test Application <== 'Test Application Business Rule' on incident:INC0010019
```



Related Links

Manage Developers
Publish to Update Set...
Grant app administration to all admins
Enable Session Debug



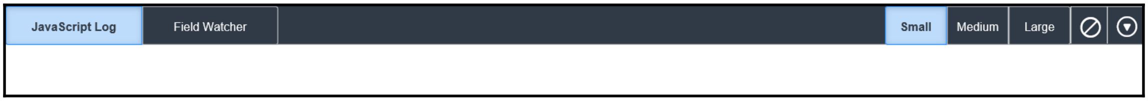
Application: Test Application

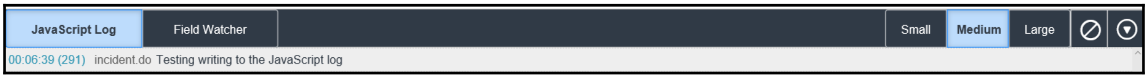
Show application picker in header: off

Update Set: Default [Test Application]

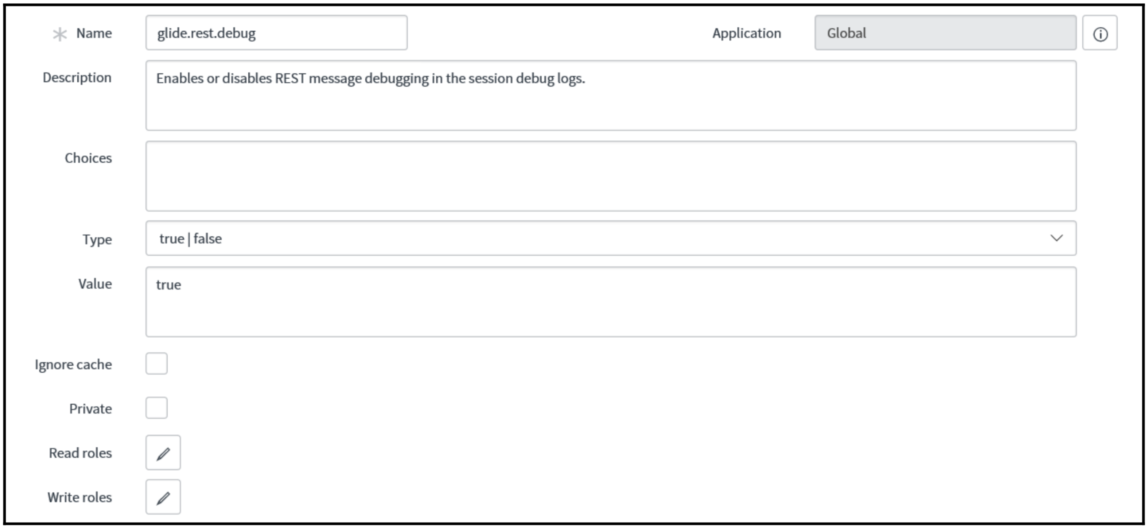
Show update set picker in header: off

JavaScript Log and Field Watcher: on

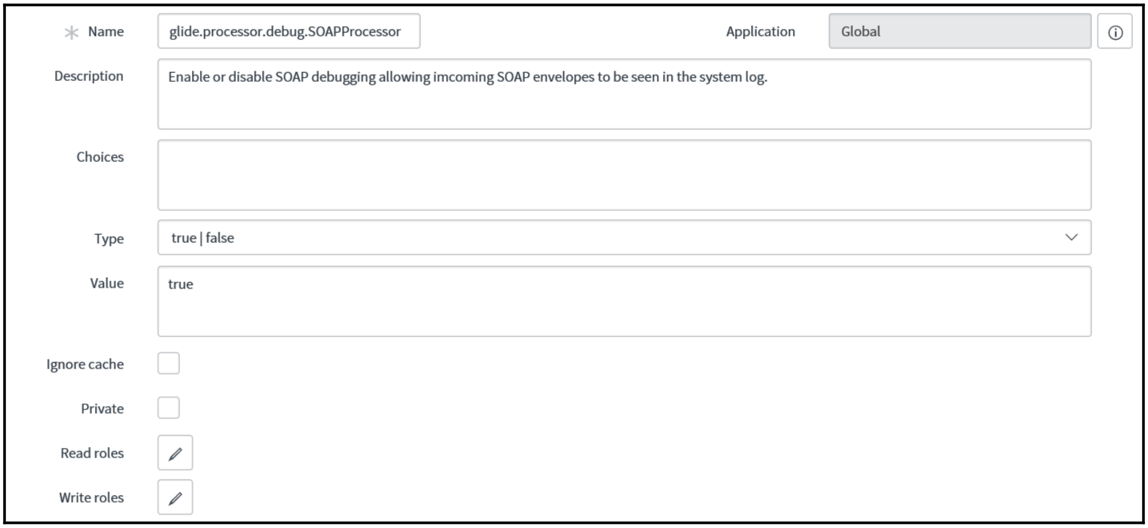




```
00:06:39 (291)  incident.do Testing writing to the JavaScript log
```

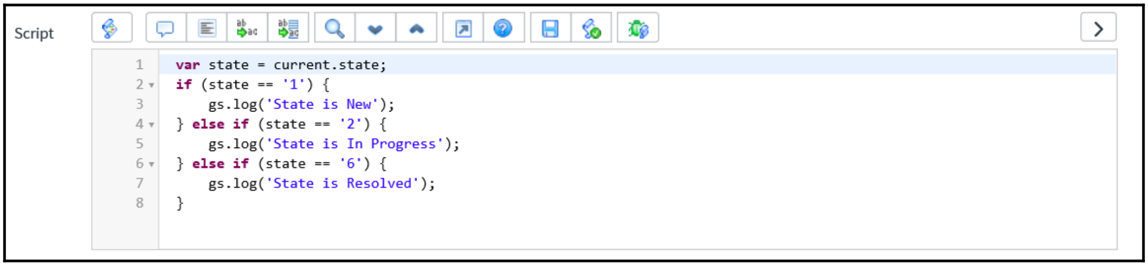


* Name	glide.rest.debug
Application	Global
Description	Enables or disables REST message debugging in the session debug logs.
Choices	
Type	true | false
Value	true
Ignore cache	□
Private	□
Read roles	
Write roles	

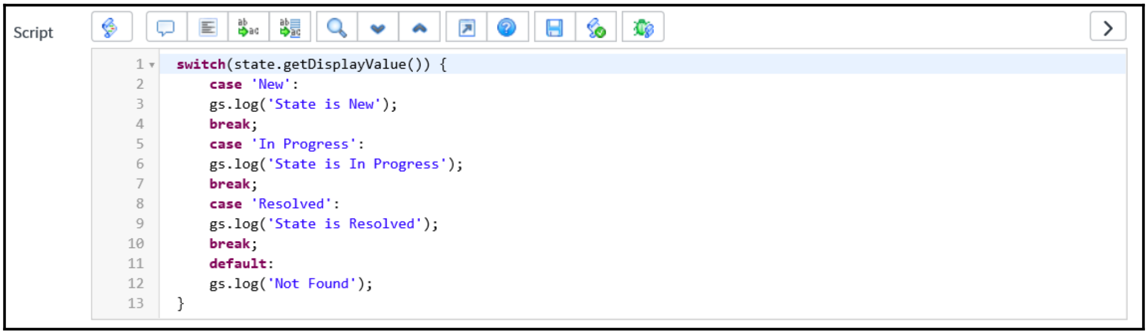


* Name	glide.processor.debug.SOAPProcessor	Application	Global
Description	Enable or disable SOAP debugging allowing imcoming SOAP envelopes to be seen in the system log.		
Choices			
Type	true | false		
Value	true		
Ignore cache	□		
Private	□		
Read roles			
Write roles			

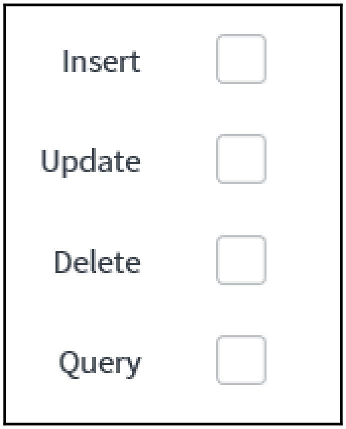
Chapter 10: Best Practices



```
var state = current.state;
if (state == '1') {
    gs.log('State is New');
} else if (state == '2') {
    gs.log('State is In Progress');
} else if (state == '6') {
    gs.log('State is Resolved');
}
```



```
switch(state.getDisplayValue()) {
    case 'New':
    gs.log('State is New');
    break;
    case 'In Progress':
    gs.log('State is In Progress');
    break;
    case 'Resolved':
    gs.log('State is Resolved');
    break;
    default:
    gs.log('Not Found');
}
```

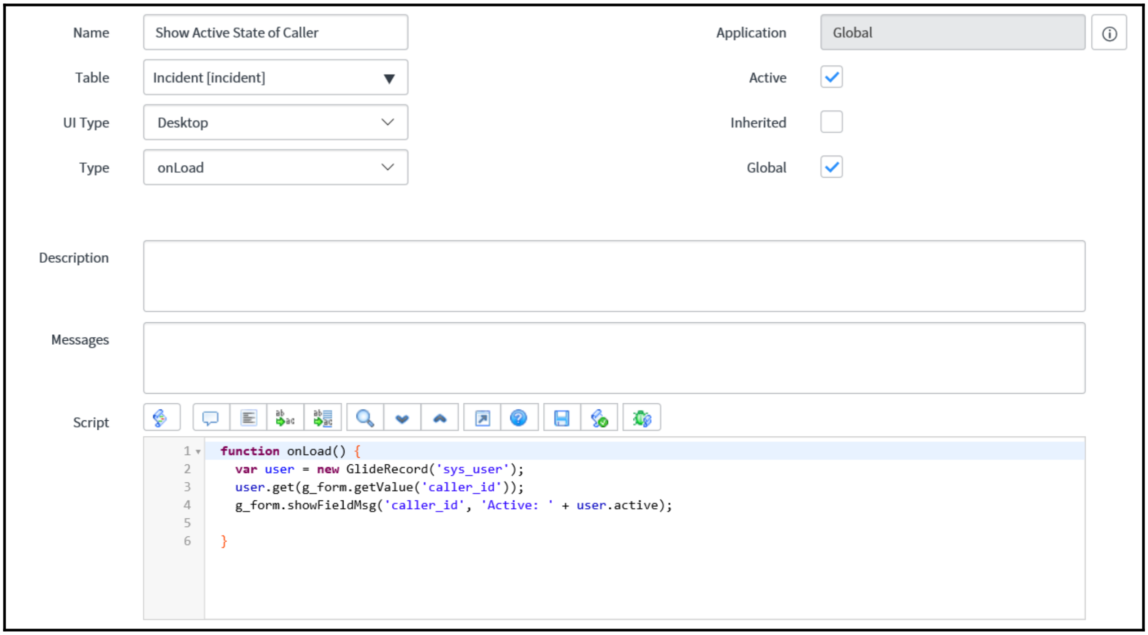


Insert □

Update □

Delete □

Query □



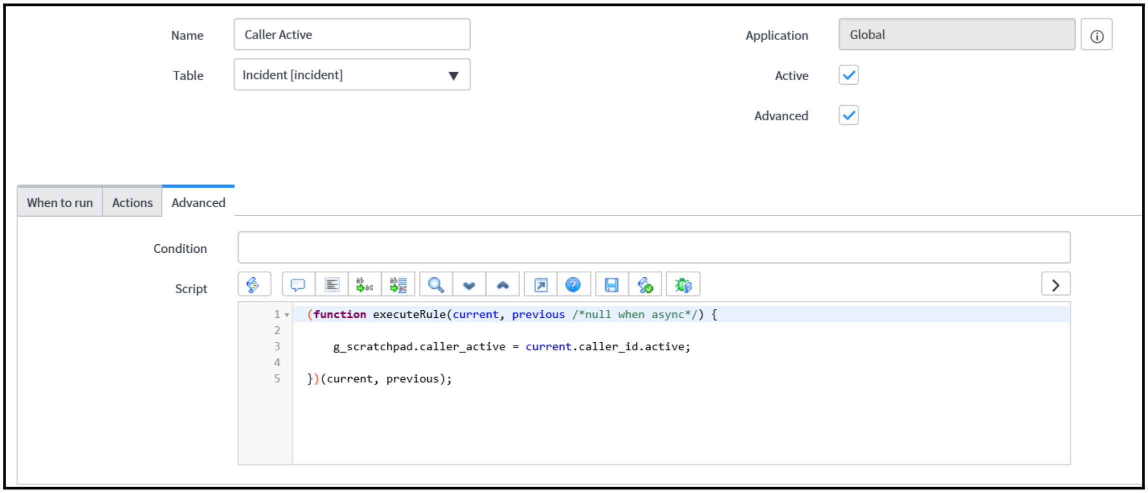
Name	Show Active State of Caller	Application	Global
Table	Incident [incident]	Active	☑
UI Type	Desktop	Inherited	□
Type	onLoad	Global	☑

Description

Messages

Script

```
function onLoad() {
  var user = new GlideRecord('sys_user');
  user.get(g_form.getValue('caller_id'));
  g_form.showFieldMsg('caller_id', 'Active: ' + user.active);

}
```

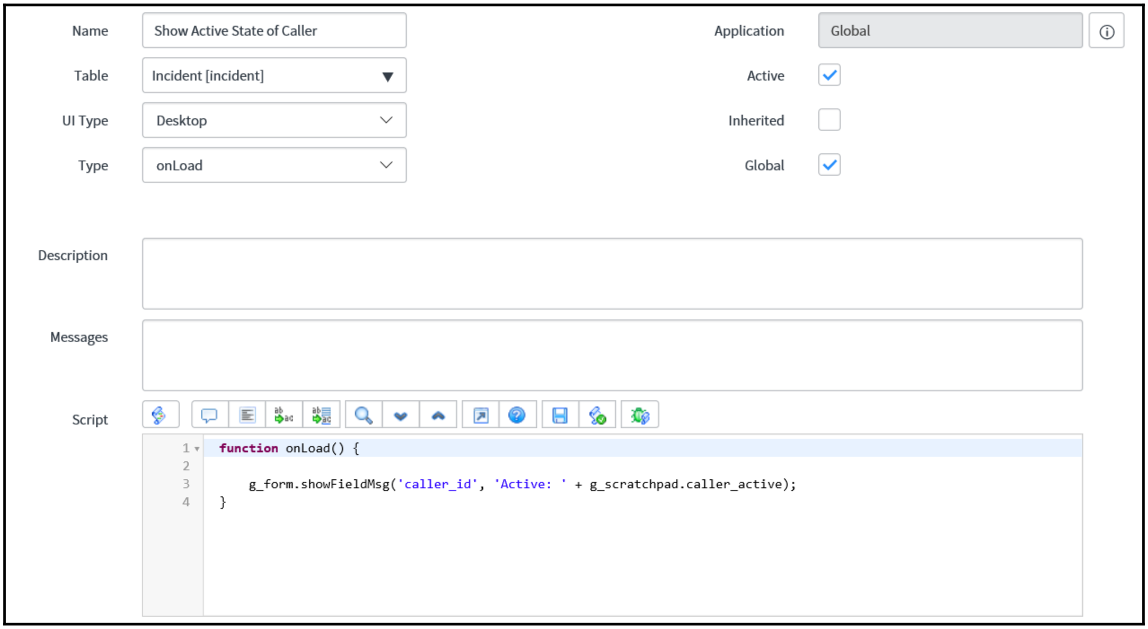


Name	Caller Active	Application	Global
Table	Incident [incident]	Active	☑
		Advanced	☑

Condition

Script

```
(function executeRule(current, previous /*null when async*/) {

    g_scratchpad.caller_active = current.caller_id.active;

})(current, previous);
```



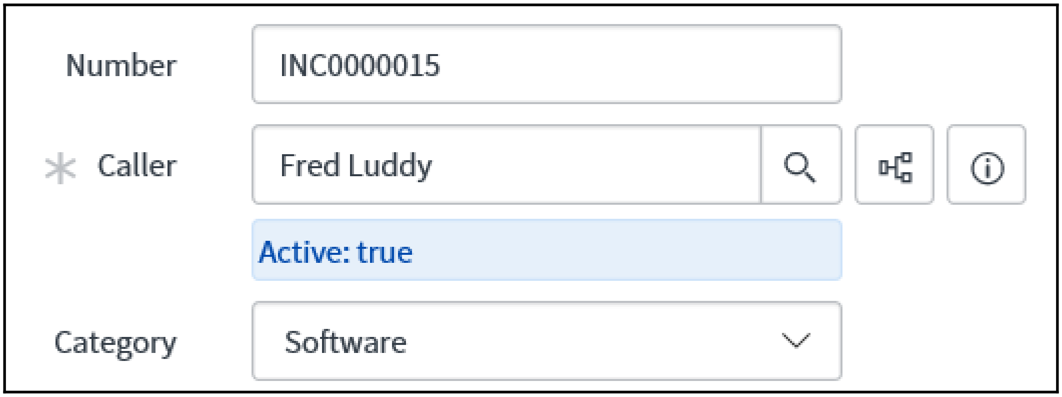
Name	Show Active State of Caller	Application	Global
Table	Incident [incident]	Active	☑
UI Type	Desktop	Inherited	□
Type	onLoad	Global	☑

Description

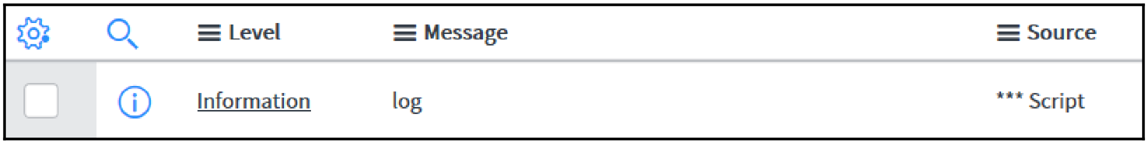
Messages

Script

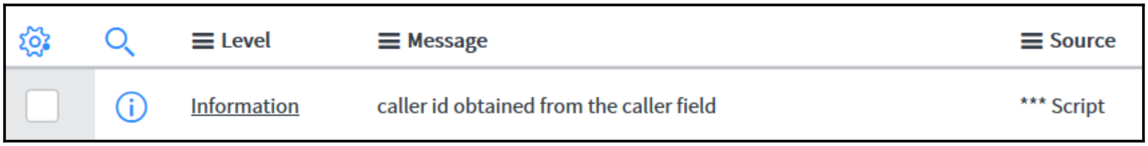
```
function onLoad() {

    g_form.showFieldMsg('caller_id', 'Active: ' + g_scratchpad.caller_active);
}
```



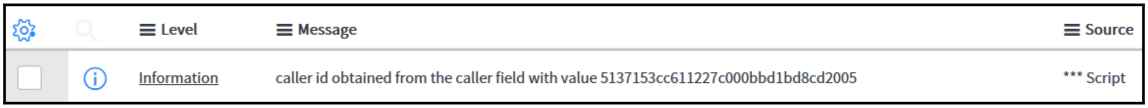
Number	INC0000015
* Caller	Fred Luddy
	Active: true
Category	Software



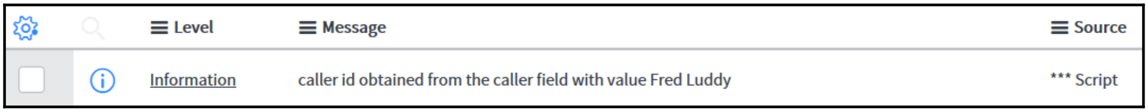
		Level	Message	Source
□	ⓘ	Information	log	*** Script



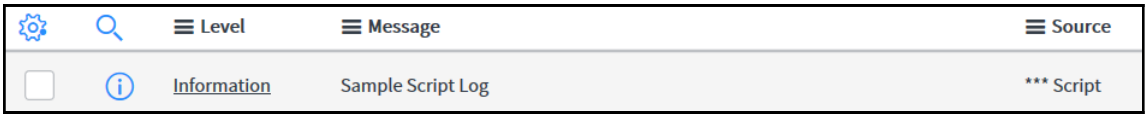
		Level	Message	Source
□	ⓘ	Information	caller id obtained from the caller field	*** Script



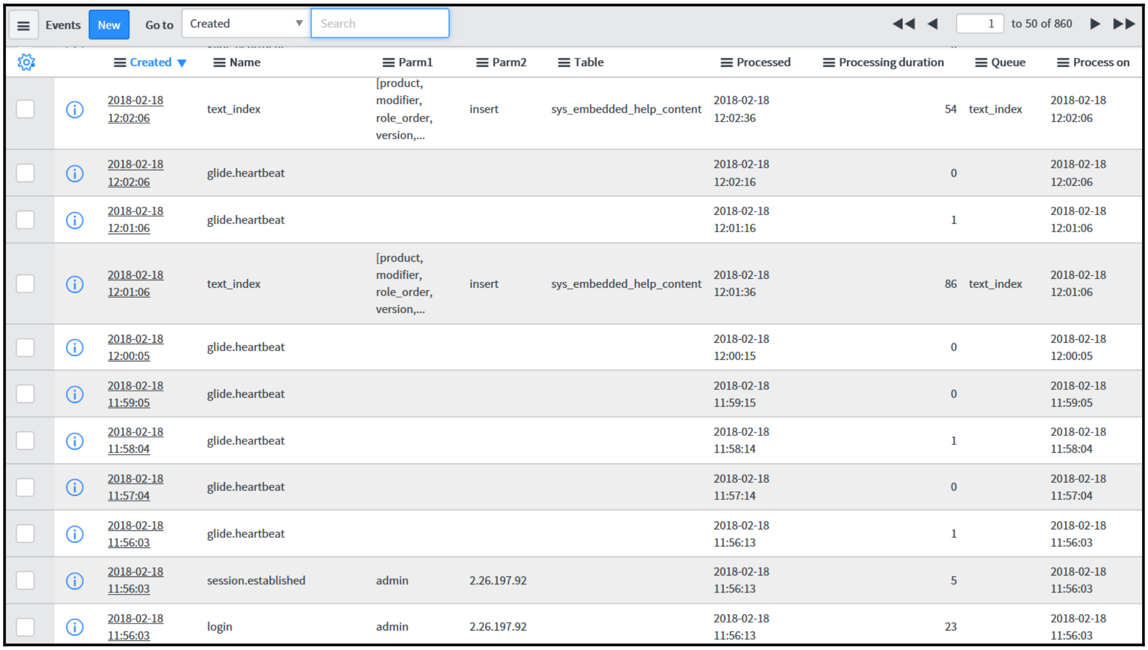
		Level	Message	Source
□	ⓘ	Information	caller id obtained from the caller field with value 5137153cc611227c000bbd1bd8cd2005	*** Script



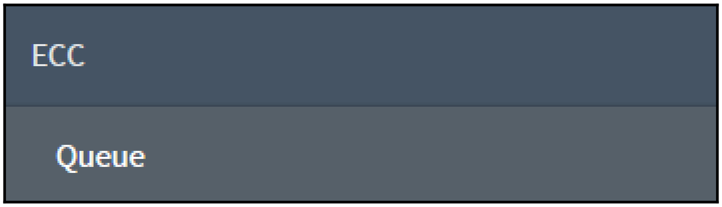
		Level	Message	Source
□	ⓘ	Information	caller id obtained from the caller field with value Fred Luddy	*** Script

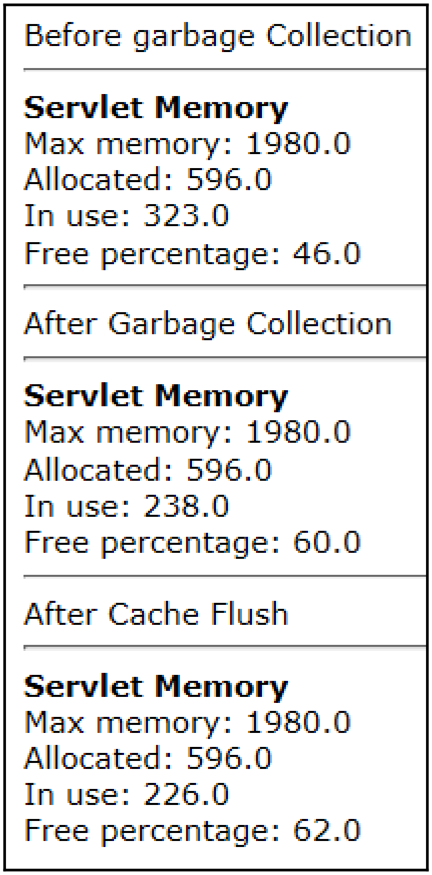


		Level	Message	Source
□	ⓘ	Information	Sample Script Log	*** Script

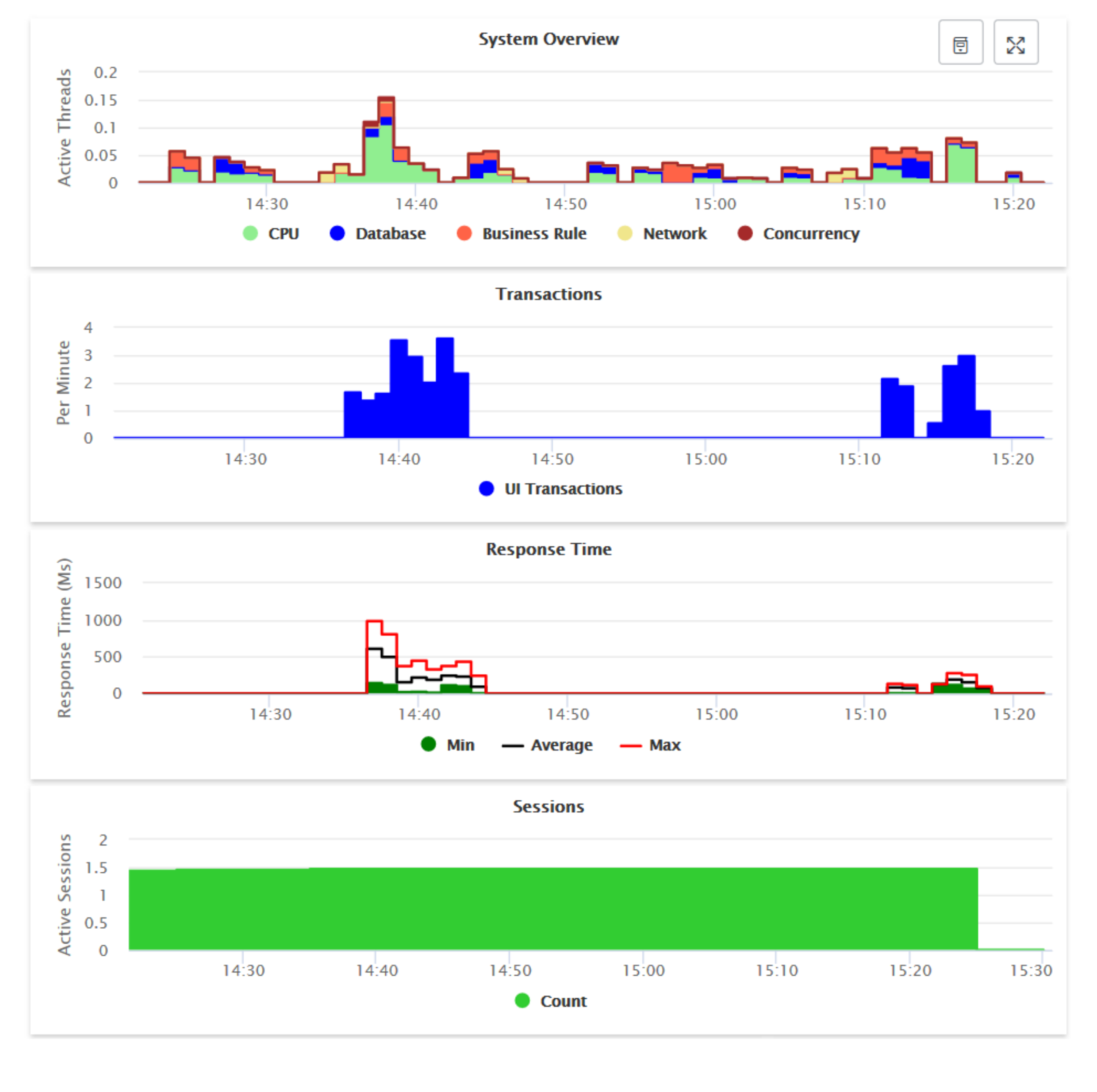


		Created ▼	Name	Parm1	Parm2	Table	Processed	Processing duration	Queue	Process on
□	ⓘ	2018-02-18 12:02:06	text_index	[product, modifier, role_order, version,...	insert	sys_embedded_help_content	2018-02-18 12:02:36	54	text_index	2018-02-18 12:02:06
□	ⓘ	2018-02-18 12:02:06	glide.heartbeat				2018-02-18 12:02:16	0		2018-02-18 12:02:06
□	ⓘ	2018-02-18 12:01:06	glide.heartbeat				2018-02-18 12:01:16	1		2018-02-18 12:01:06
□	ⓘ	2018-02-18 12:01:06	text_index	[product, modifier, role_order, version,...	insert	sys_embedded_help_content	2018-02-18 12:01:36	86	text_index	2018-02-18 12:01:06
□	ⓘ	2018-02-18 12:00:05	glide.heartbeat				2018-02-18 12:00:15	0		2018-02-18 12:00:05
□	ⓘ	2018-02-18 11:59:05	glide.heartbeat				2018-02-18 11:59:15	0		2018-02-18 11:59:05
□	ⓘ	2018-02-18 11:58:04	glide.heartbeat				2018-02-18 11:58:14	1		2018-02-18 11:58:04
□	ⓘ	2018-02-18 11:57:04	glide.heartbeat				2018-02-18 11:57:14	0		2018-02-18 11:57:04
□	ⓘ	2018-02-18 11:56:03	glide.heartbeat				2018-02-18 11:56:13	1		2018-02-18 11:56:03
□	ⓘ	2018-02-18 11:56:03	session.established	admin	2.26.197.92		2018-02-18 11:56:13	5		2018-02-18 11:56:03
□	ⓘ	2018-02-18 11:56:03	login	admin	2.26.197.92		2018-02-18 11:56:13	23		2018-02-18 11:56:03

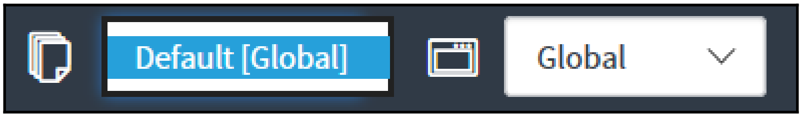


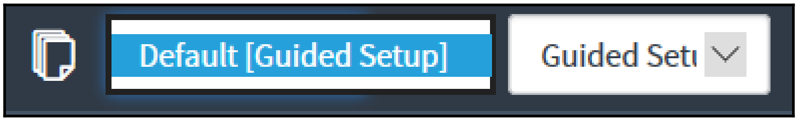


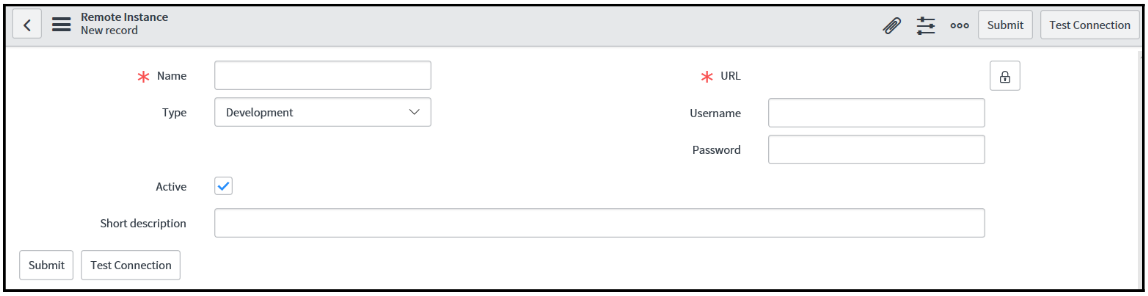
```
Before garbage Collection

Servlet Memory
Max memory: 1980.0
Allocated: 596.0
In use: 323.0
Free percentage: 46.0

After Garbage Collection

Servlet Memory
Max memory: 1980.0
Allocated: 596.0
In use: 238.0
Free percentage: 60.0

After Cache Flush

Servlet Memory
Max memory: 1980.0
Allocated: 596.0
In use: 226.0
Free percentage: 62.0
```


System Overview
Active Threads
0.2
0.15
0.1
0.05
0
14:30
14:40
14:50
15:00
15:10
15:20
● CPU
● Database
● Business Rule
● Network
● Concurrency
Transactions
Per Minute
4
3
2
1
0
14:30
14:40
14:50
15:00
15:10
15:20
● UI Transactions
Response Time
Response Time (Ms)
1500
1000
500
0
14:30
14:40
14:50
15:00
15:10
15:20
● Min
— Average
— Max
Sessions
Active Sessions
2
1.5
1
0.5
0
14:30
14:40
14:50
15:00
15:10
15:20
15:30
● Count

Chapter 11: Deployments with the Update Sets







* Name

Type: Development

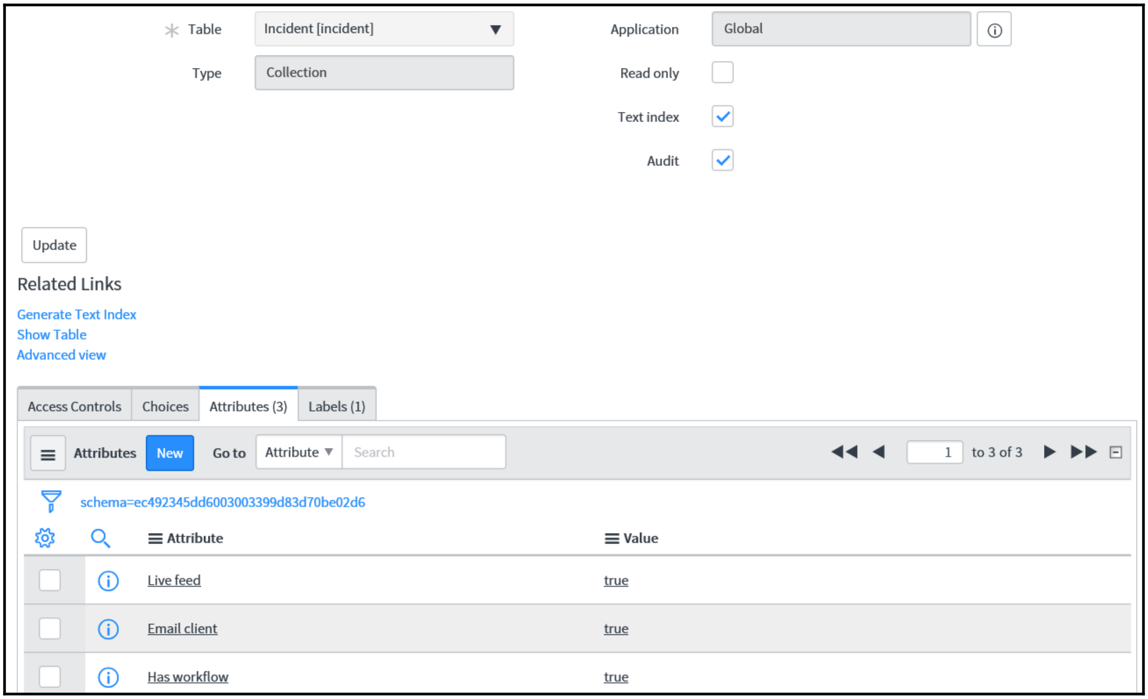
* URL

Username

Password

Active ☑

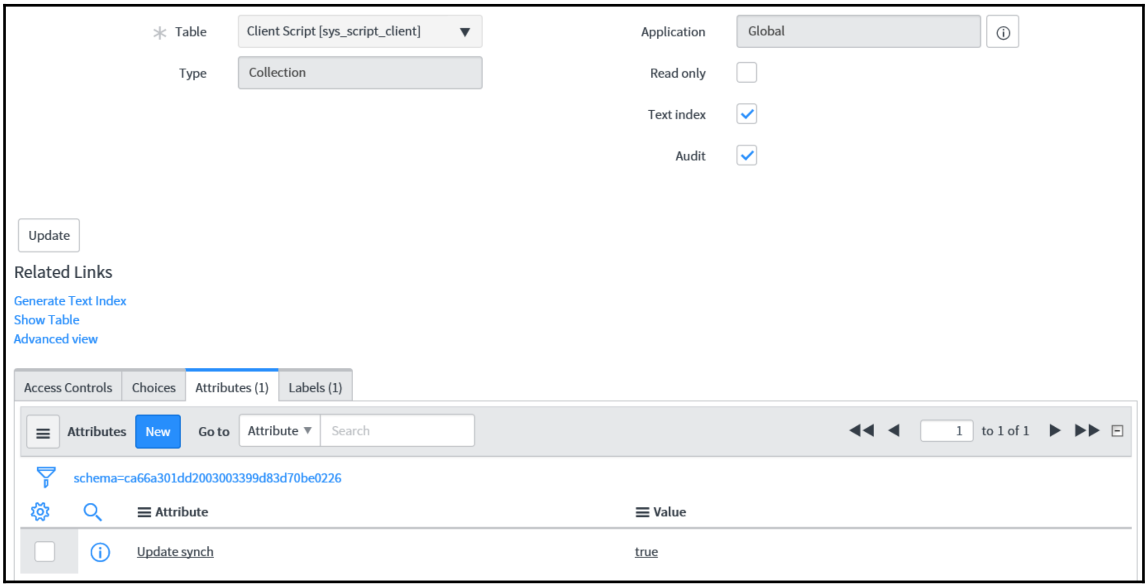
Short description



* Table	Incident [incident]	Application	Global
Type	Collection	Read only	☐
		Text index	☑
		Audit	☑

schema=ec492345dd6003003399d83d70be02d6

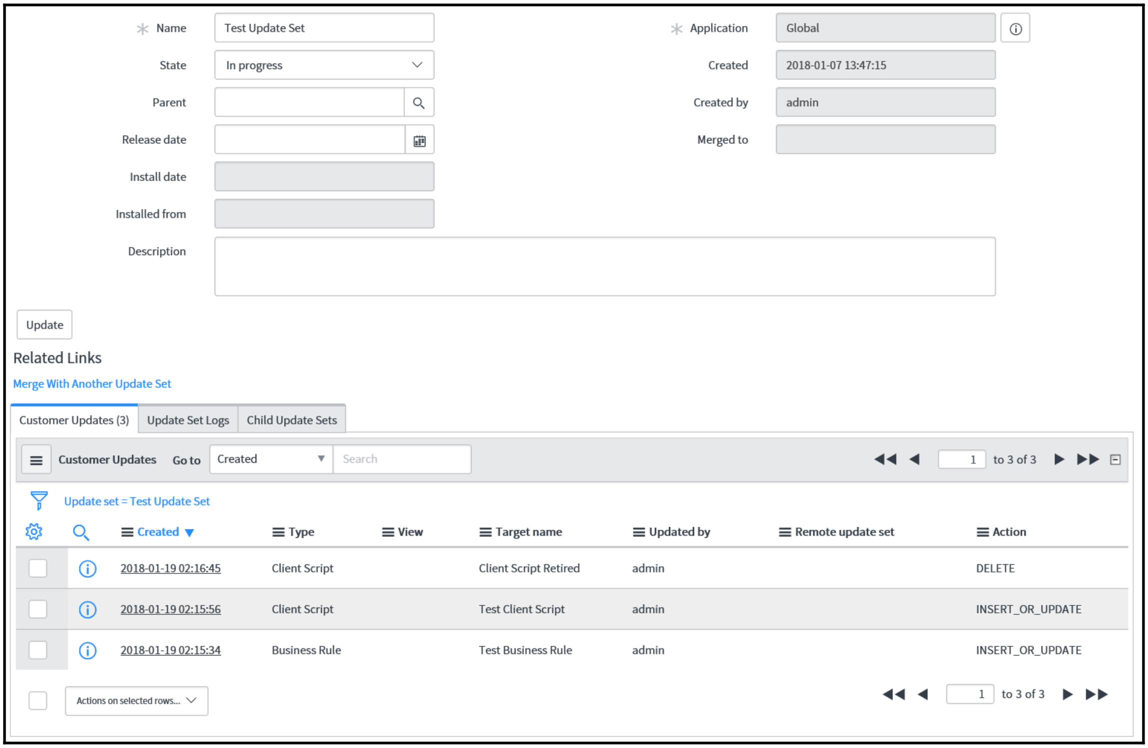
Attribute	Value
Live feed	true
Email client	true
Has workflow	true



* Table	Client Script [sys_script_client]	Application	Global
Type	Collection	Read only	☐
		Text index	☑
		Audit	☑

schema=ca66a301dd2003003399d83d70be0226

Attribute	Value
Update synch	true



* Name	Test Update Set	* Application	Global
State	In progress	Created	2018-01-07 13:47:15
Parent		Created by	admin
Release date		Merged to	
Install date			
Installed from			
Description			

Update

Related Links

Merge With Another Update Set

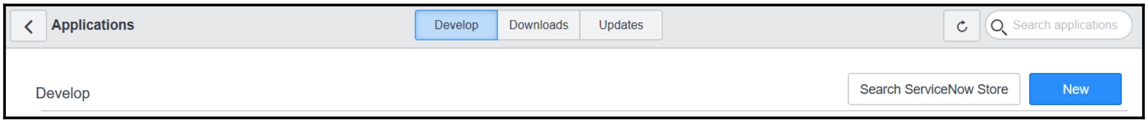
Customer Updates (3) | Update Set Logs | Child Update Sets

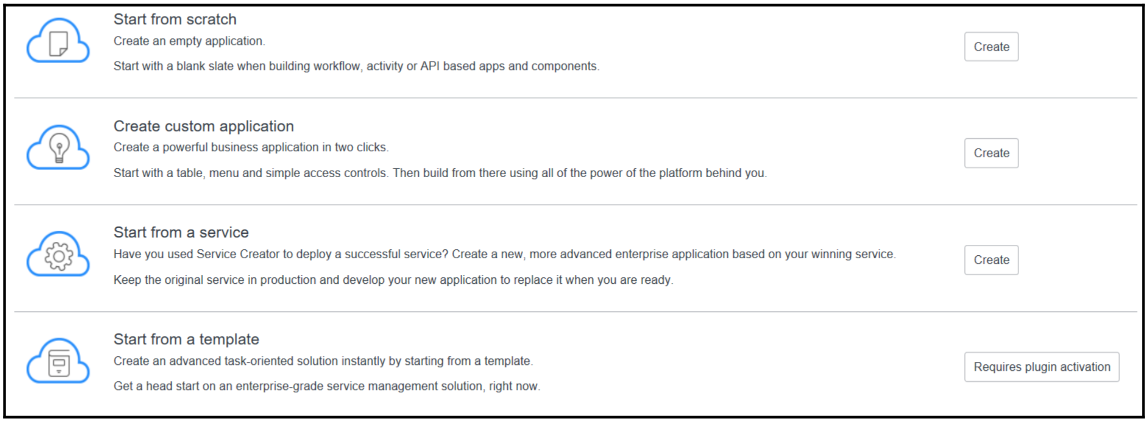
Customer Updates Go to Created Search 1 to 3 of 3

Update set = Test Update Set

Created	Type	View	Target name	Updated by	Remote update set	Action
2018-01-19 02:16:45	Client Script		Client Script Retired	admin		DELETE
2018-01-19 02:15:56	Client Script		Test Client Script	admin		INSERT_OR_UPDATE
2018-01-19 02:15:34	Business Rule		Test Business Rule	admin		INSERT_OR_UPDATE

Chapter 12: Building a Custom Application Using ServiceNow Scripting




Start from scratch

Create an empty application.

Start with a blank slate when building workflow, activity or API based apps and components.

Create custom application

Create a powerful business application in two clicks.

Start with a table, menu and simple access controls. Then build from there using all of the power of the platform behind you.

Start from a service

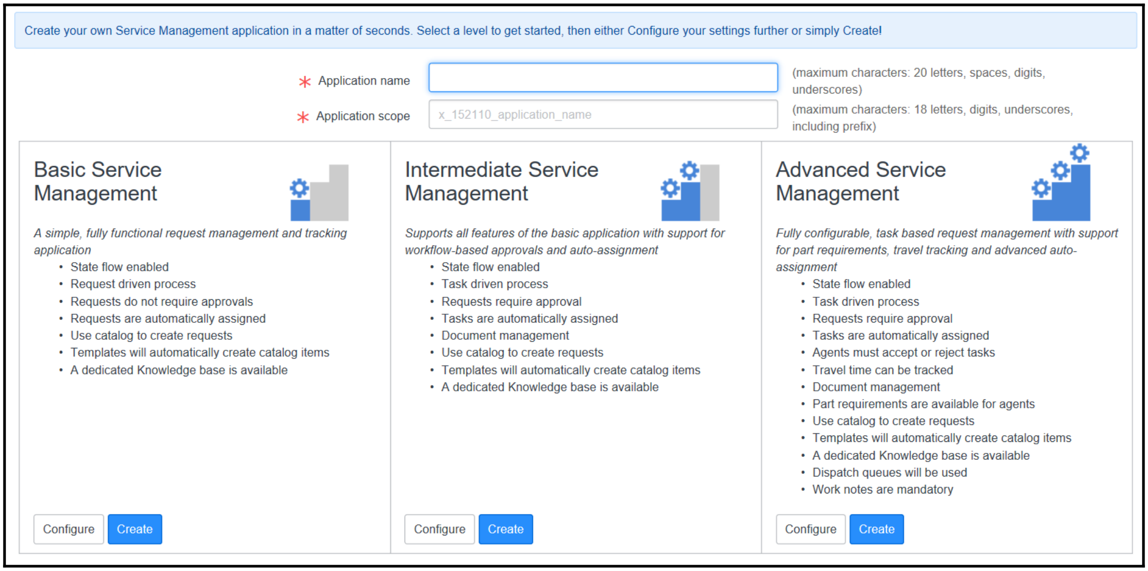
Have you used Service Creator to deploy a successful service? Create a new, more advanced enterprise application based on your winning service.

Keep the original service in production and develop your new application to replace it when you are ready.

Start from a template

Create an advanced task-oriented solution instantly by starting from a template.

Get a head start on an enterprise-grade service management solution, right now.


Create your own Service Management application in a matter of seconds. Select a level to get started, then either Configure your settings further or simply Create!

* Application name: (maximum characters: 20 letters, spaces, digits, underscores)

* Application scope: x_152110_application_name (maximum characters: 18 letters, digits, underscores, including prefix)

Basic Service Management

A simple, fully functional request management and tracking application

- State flow enabled
- Request driven process
- Requests do not require approvals
- Requests are automatically assigned
- Use catalog to create requests
- Templates will automatically create catalog items
- A dedicated Knowledge base is available

Intermediate Service Management

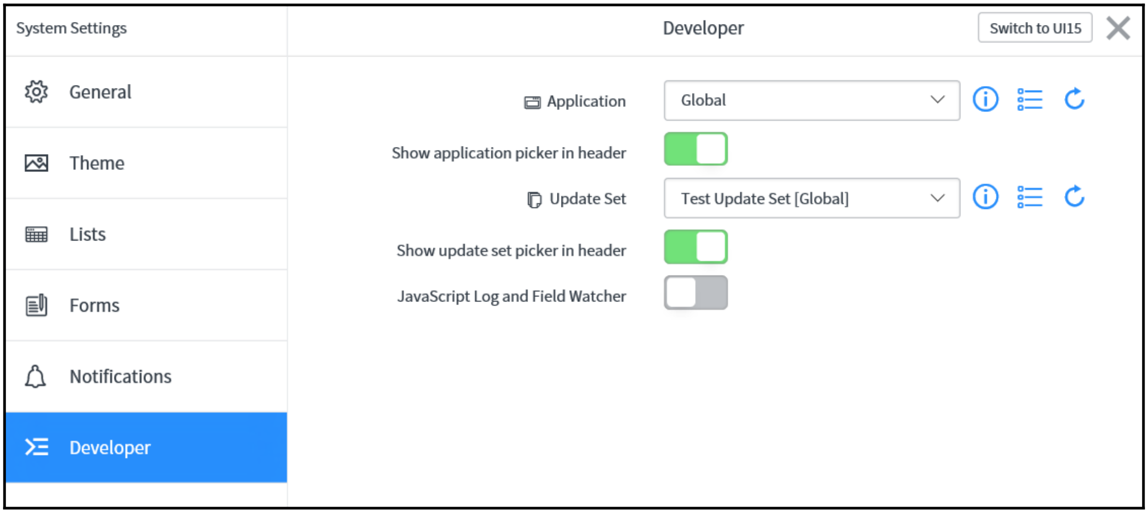
Supports all features of the basic application with support for workflow-based approvals and auto-assignment

- State flow enabled
- Task driven process
- Requests require approval
- Tasks are automatically assigned
- Document management
- Use catalog to create requests
- Templates will automatically create catalog items
- A dedicated Knowledge base is available

Advanced Service Management

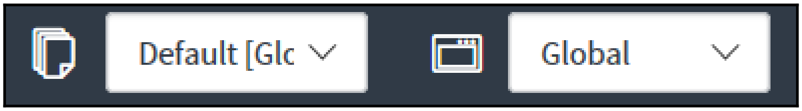
Fully configurable, task based request management with support for part requirements, travel tracking and advanced auto-assignment

- State flow enabled
- Task driven process
- Requests require approval
- Tasks are automatically assigned
- Agents must accept or reject tasks
- Travel time can be tracked
- Document management
- Part requirements are available for agents
- Use catalog to create requests
- Templates will automatically create catalog items
- A dedicated Knowledge base is available
- Dispatch queues will be used
- Work notes are mandatory

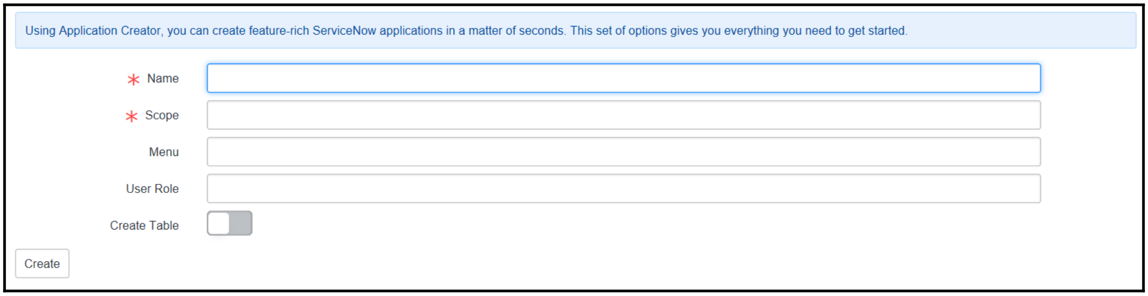


Developer

Application	Global
Show application picker in header	On
Update Set	Test Update Set [Global]
Show update set picker in header	On
JavaScript Log and Field Watcher	Off

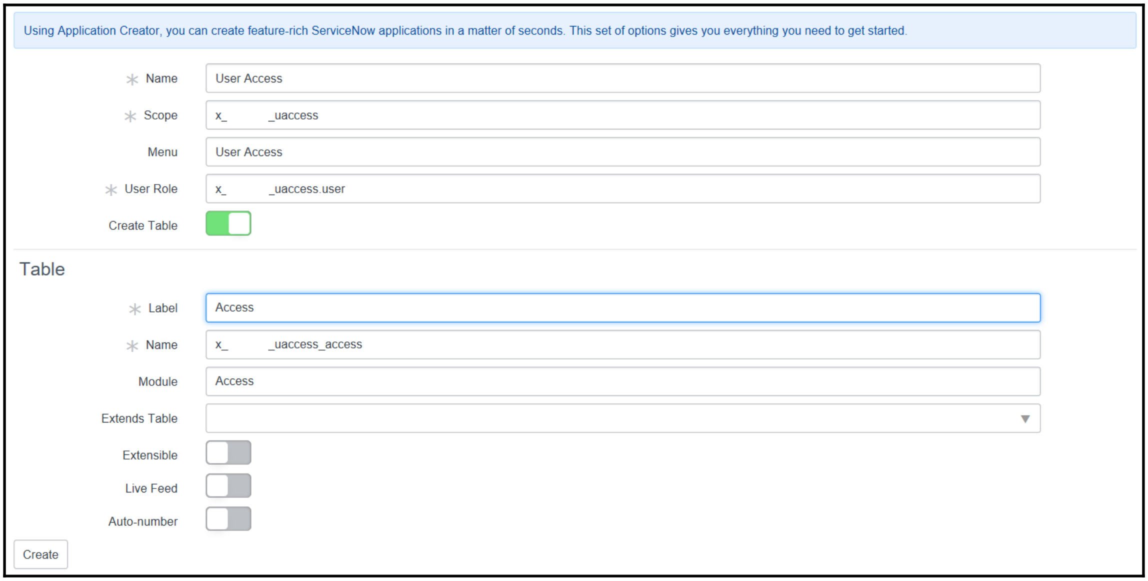


Default [Glc ⌄ Global ⌄



Using Application Creator, you can create feature-rich ServiceNow applications in a matter of seconds. This set of options gives you everything you need to get started.

* Name	
* Scope	
Menu	
User Role	
Create Table	Off

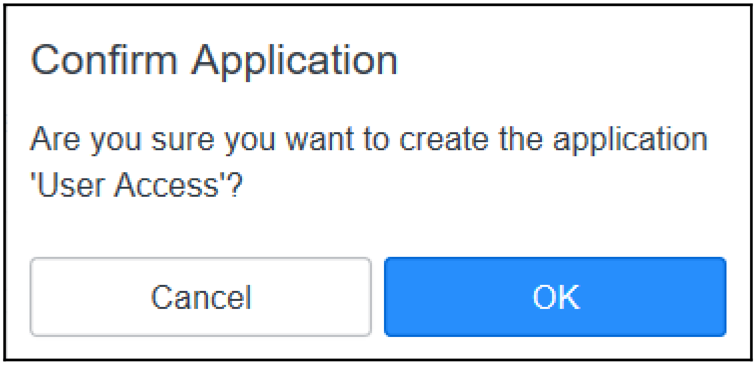


Using Application Creator, you can create feature-rich ServiceNow applications in a matter of seconds. This set of options gives you everything you need to get started.

* Name	User Access
* Scope	x_ _uaccess
Menu	User Access
* User Role	x_ _uaccess.user
Create Table	●

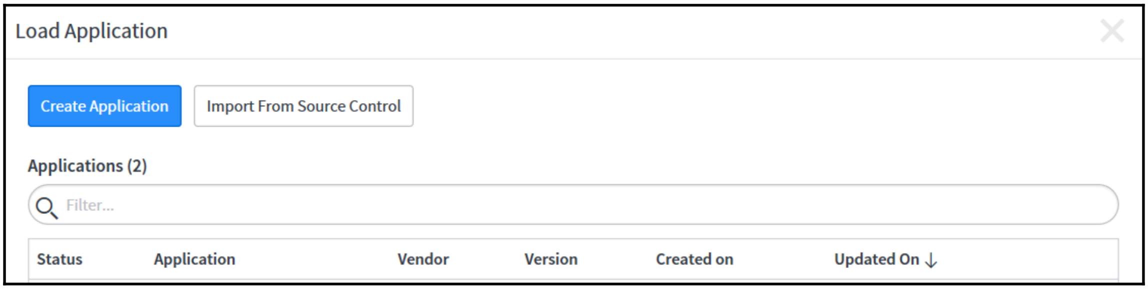
Table

* Label	Access
* Name	x_ _uaccess_access
Module	Access
Extends Table	
Extensible	○
Live Feed	○
Auto-number	○



Confirm Application

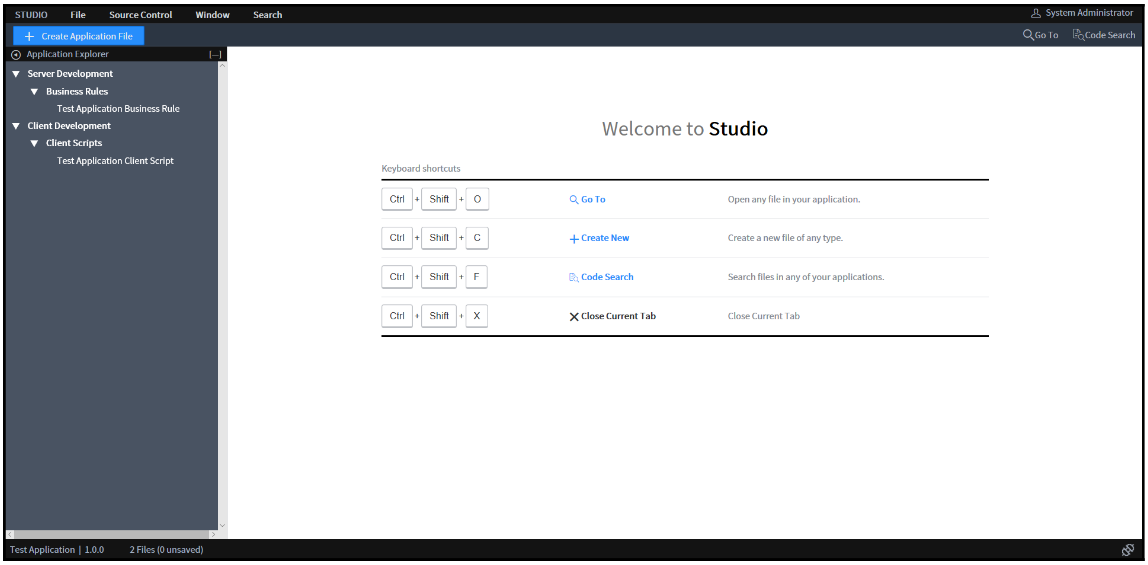
Are you sure you want to create the application 'User Access'?



Load Application

Applications (2)

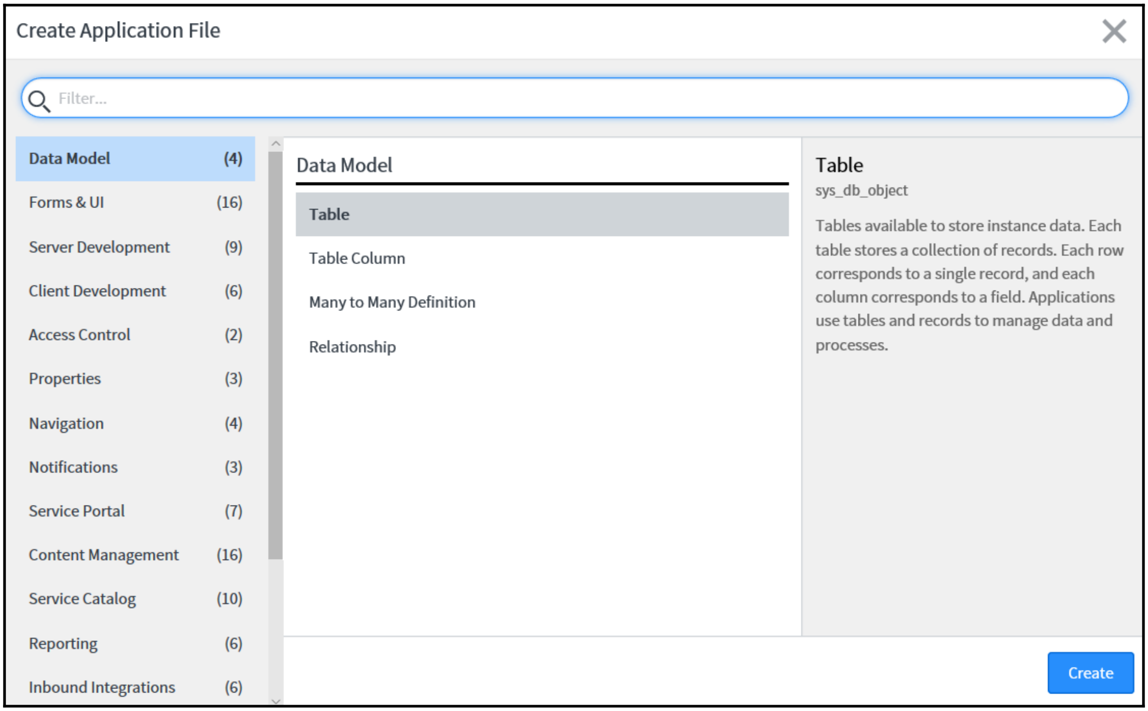
Status	Application	Vendor	Version	Created on	Updated On ↓



Welcome to **Studio**

Keyboard shortcuts

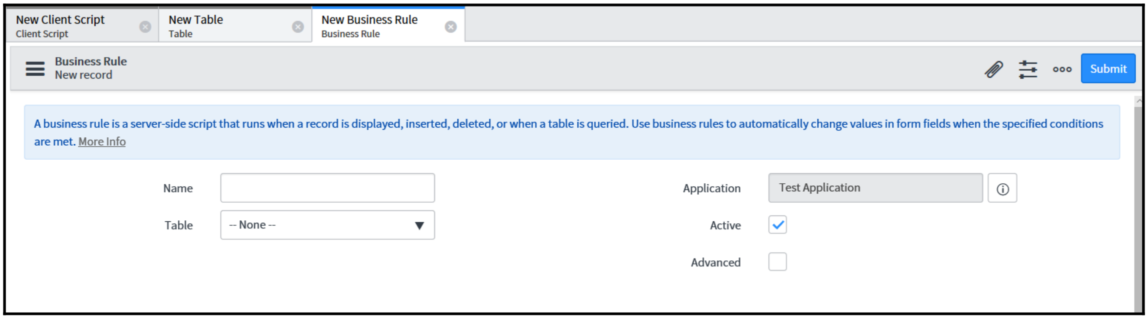
Ctrl + Shift + O	Go To	Open any file in your application.
Ctrl + Shift + C	Create New	Create a new file of any type.
Ctrl + Shift + F	Code Search	Search files in any of your applications.
Ctrl + Shift + X	Close Current Tab	Close Current Tab



Table

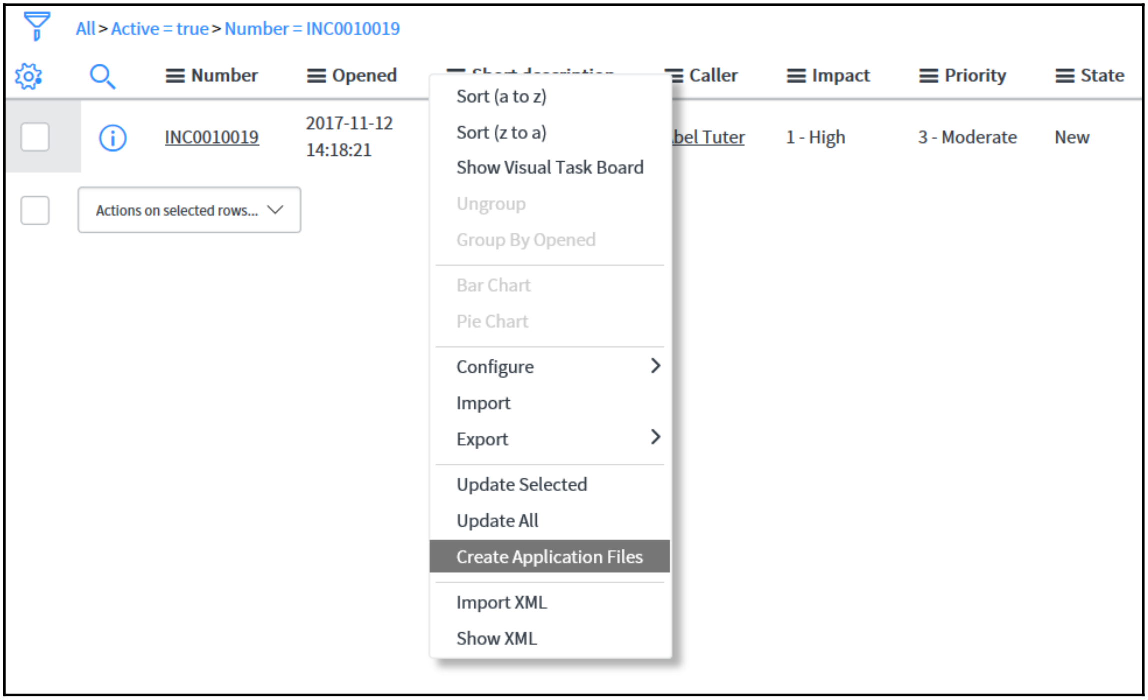
sys_db_object

Tables available to store instance data. Each table stores a collection of records. Each row corresponds to a single record, and each column corresponds to a field. Applications use tables and records to manage data and processes.

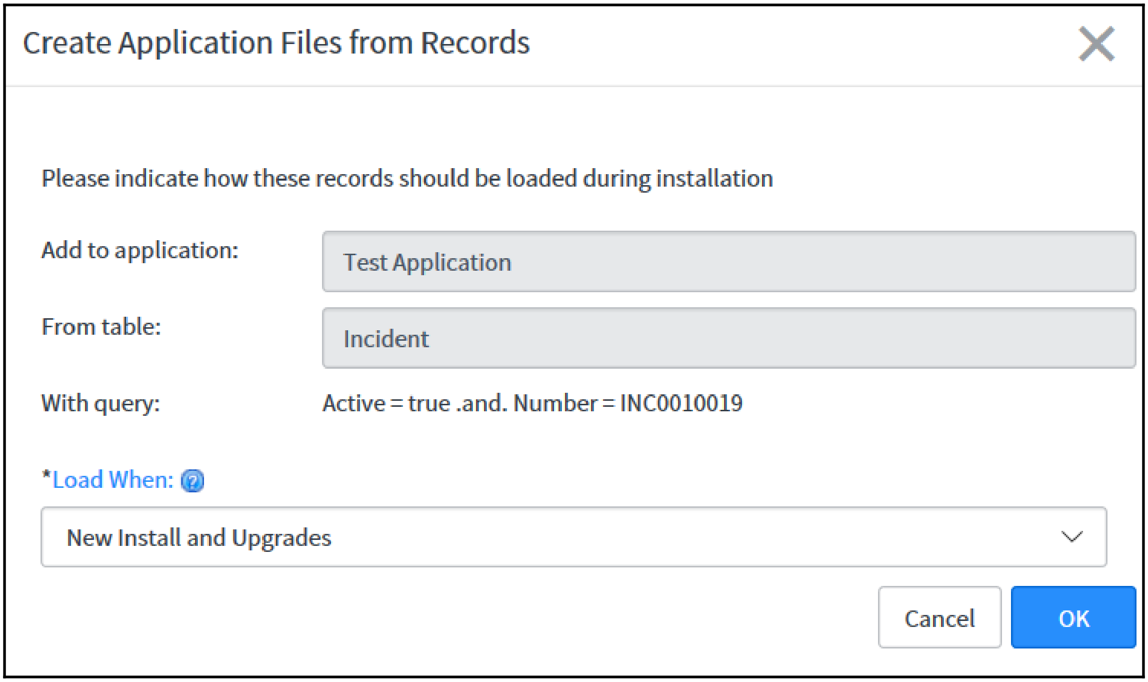


A business rule is a server-side script that runs when a record is displayed, inserted, deleted, or when a table is queried. Use business rules to automatically change values in form fields when the specified conditions are met. More Info

Name		Application	Test Application
Table	-- None --	Active	☑
		Advanced	□


All > Active = true > Number = INC0010019

Number	Opened	Caller	Impact	Priority	State
INC0010019	2017-11-12 14:18:21	bel Tuter	1 - High	3 - Moderate	New



Create Application Files from Records

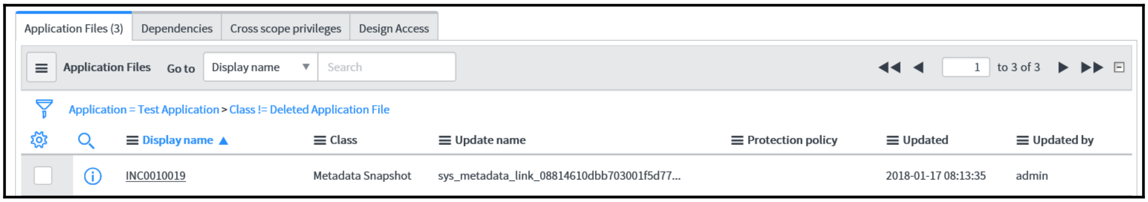
Please indicate how these records should be loaded during installation

Add to application: Test Application

From table: Incident

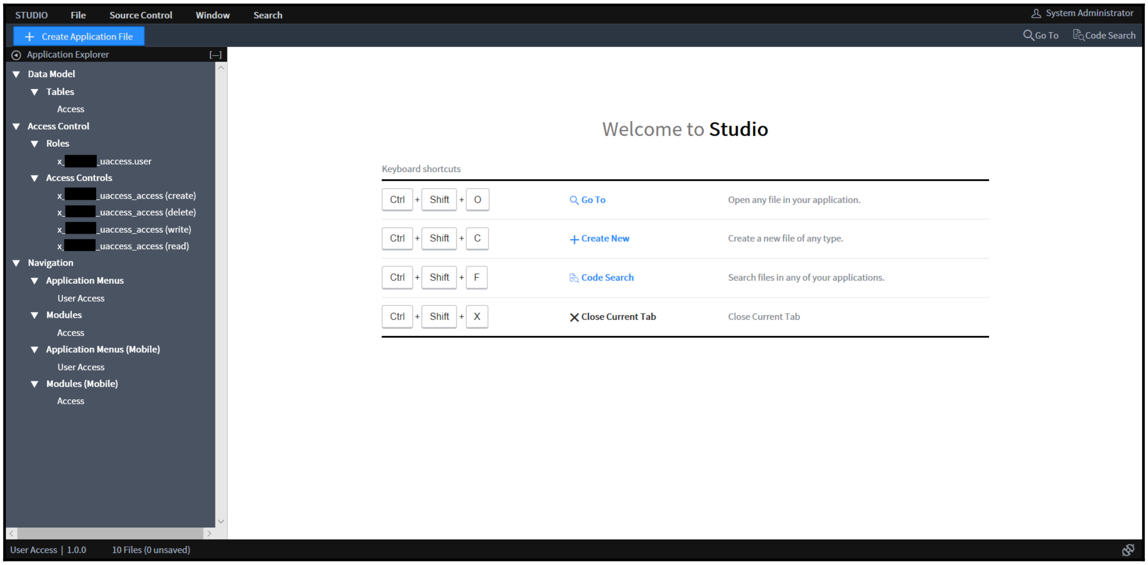
With query: Active = true .and. Number = INC0010019

*Load When: New Install and Upgrades



Application = Test Application > Class != Deleted Application File

Display name ▲	Class	Update name	Protection policy	Updated	Updated by
INC0010019	Metadata Snapshot	sys_metadata_link_08814610dbb703001f5d77...		2018-01-17 08:13:35	admin



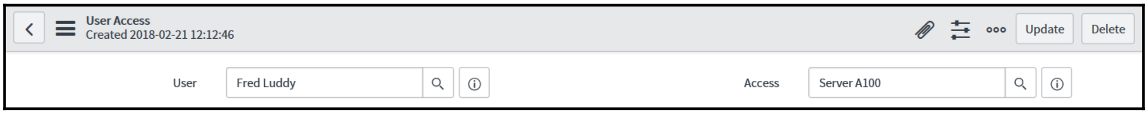
- Data Model
  - Tables
    - Access
- Access Control
  - Roles
    - x_[redacted]_uaccess.user
  - Access Controls
    - x_[redacted]_uaccess_access (create)
    - x_[redacted]_uaccess_access (delete)
    - x_[redacted]_uaccess_access (write)
    - x_[redacted]_uaccess_access (read)
- Navigation
  - Application Menus
    - User Access
  - Modules
    - Access
  - Application Menus (Mobile)
    - User Access
  - Modules (Mobile)
    - Access

Welcome to **Studio**

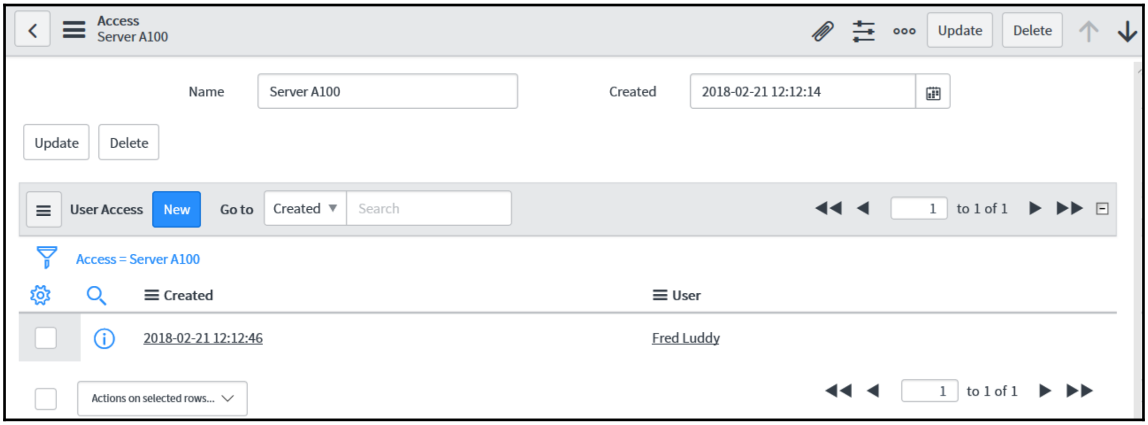
Keyboard shortcuts

Ctrl + Shift + O	Go To	Open any file in your application.
Ctrl + Shift + C	Create New	Create a new file of any type.
Ctrl + Shift + F	Code Search	Search files in any of your applications.
Ctrl + Shift + X	Close Current Tab	Close Current Tab

User Access | 1.0.0   10 Files (0 unsaved)


User Access
Created 2018-02-21 12:12:46

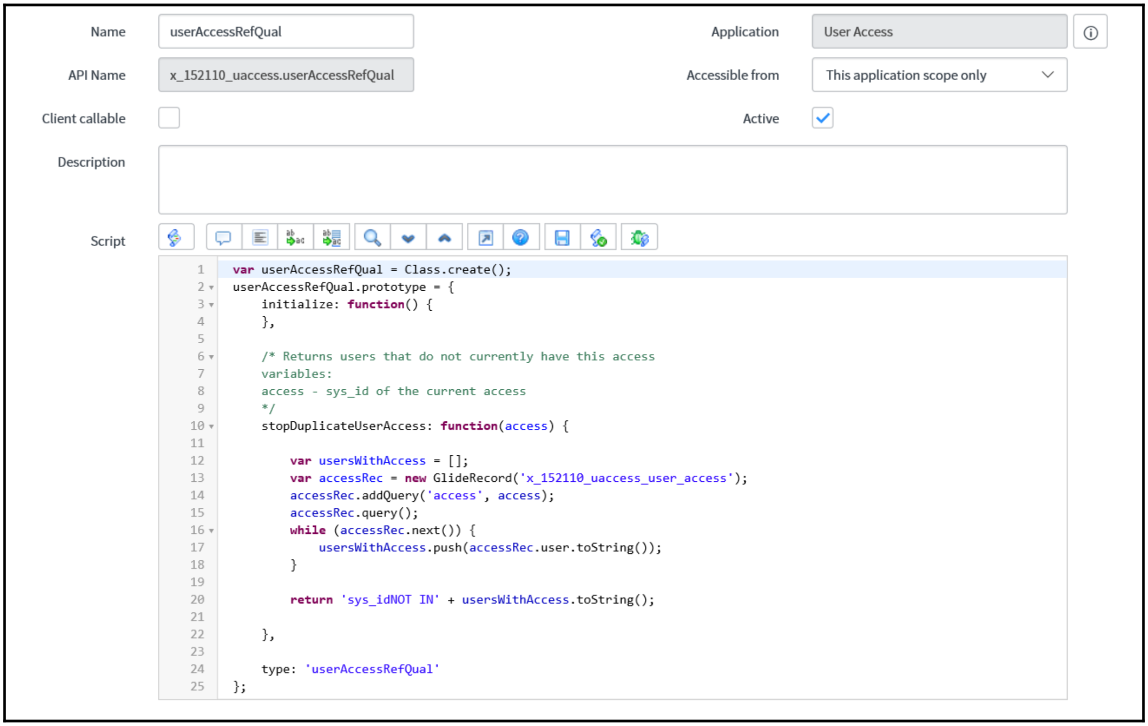
User	Fred Luddy	Access	Server A100


Access
Server A100

Name	Server A100	Created	2018-02-21 12:12:14

Access = Server A100

Created	User
2018-02-21 12:12:46	Fred Luddy



Name userAccessRefQual

API Name x_152110_uaccess.userAccessRefQual

Client callable □

Application User Access

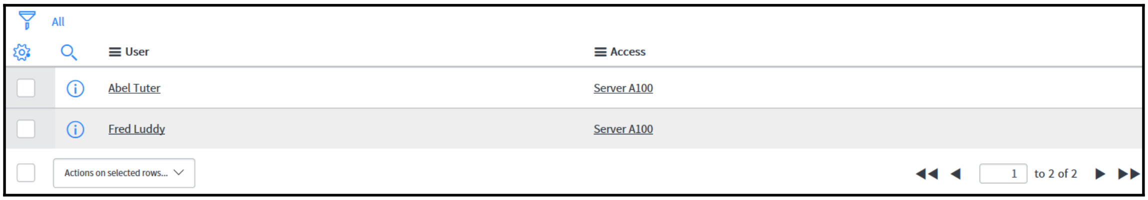
Accessible from This application scope only

Active ☑

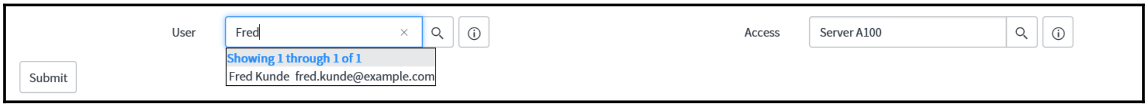
Description

Script

```
var userAccessRefQual = Class.create();
userAccessRefQual.prototype = {
    initialize: function() {
    },

    /* Returns users that do not currently have this access
    variables:
    access - sys_id of the current access
    */
    stopDuplicateUserAccess: function(access) {

        var usersWithAccess = [];
        var accessRec = new GlideRecord('x_152110_uaccess_user_access');
        accessRec.addQuery('access', access);
        accessRec.query();
        while (accessRec.next()) {
            usersWithAccess.push(accessRec.user.toString());
        }

        return 'sys_idNOT IN' + usersWithAccess.toString();

    },

    type: 'userAccessRefQual'
};
```



	User	Access
□	Abel Tuter	Server A100
□	Fred Luddy	Server A100

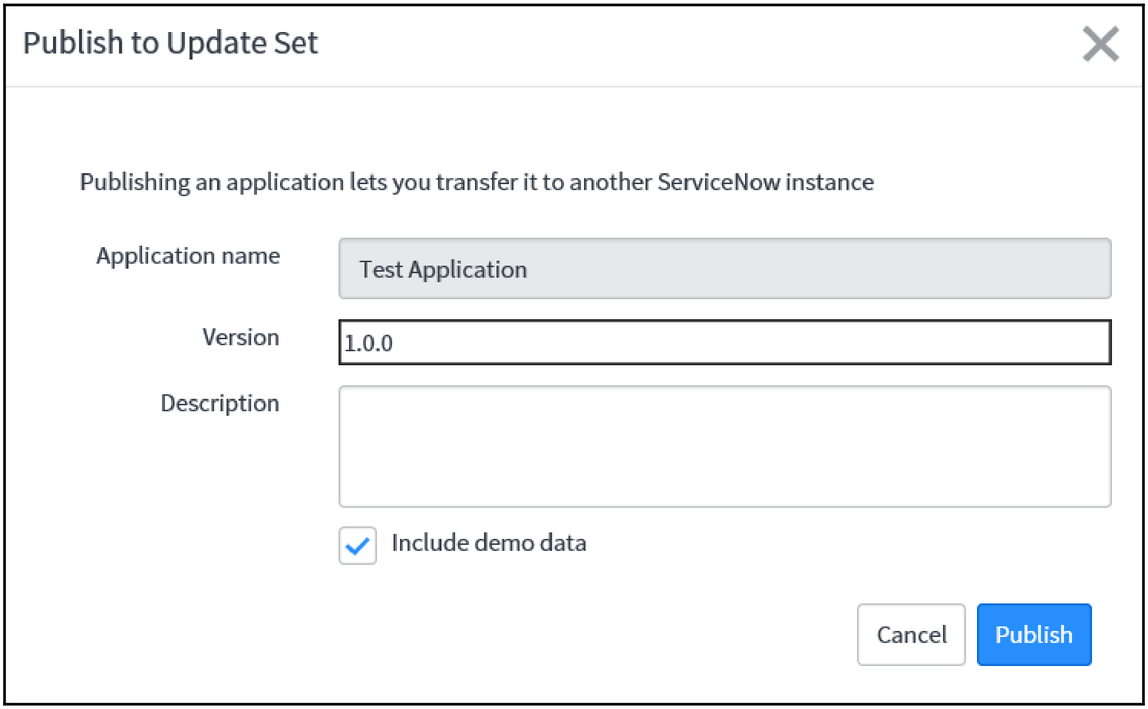


User Fred

Showing 1 through 1 of 1

Fred Kunde fred.kunde@example.com

Access Server A100


Publish to Update Set

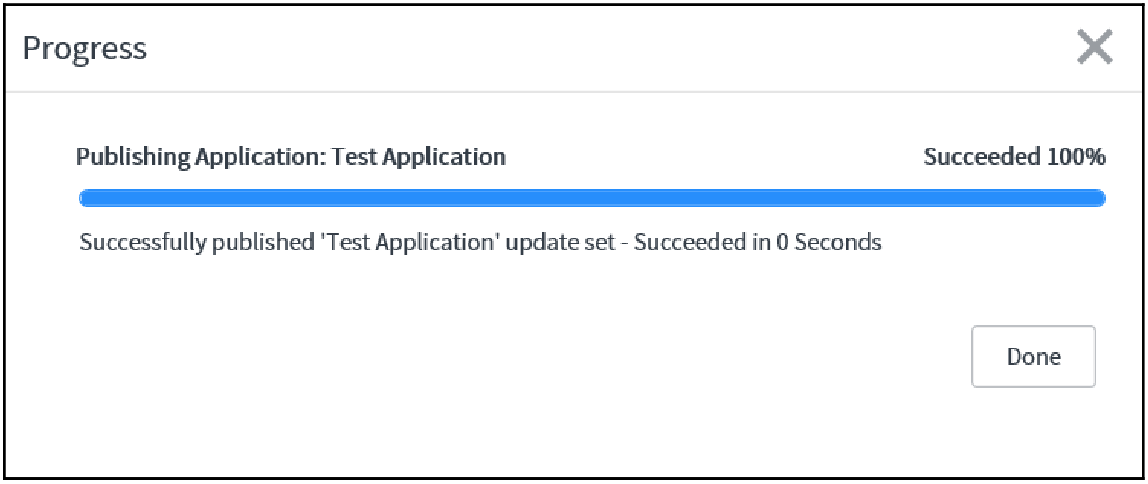
Publishing an application lets you transfer it to another ServiceNow instance

Application name: Test Application

Version: 1.0.0

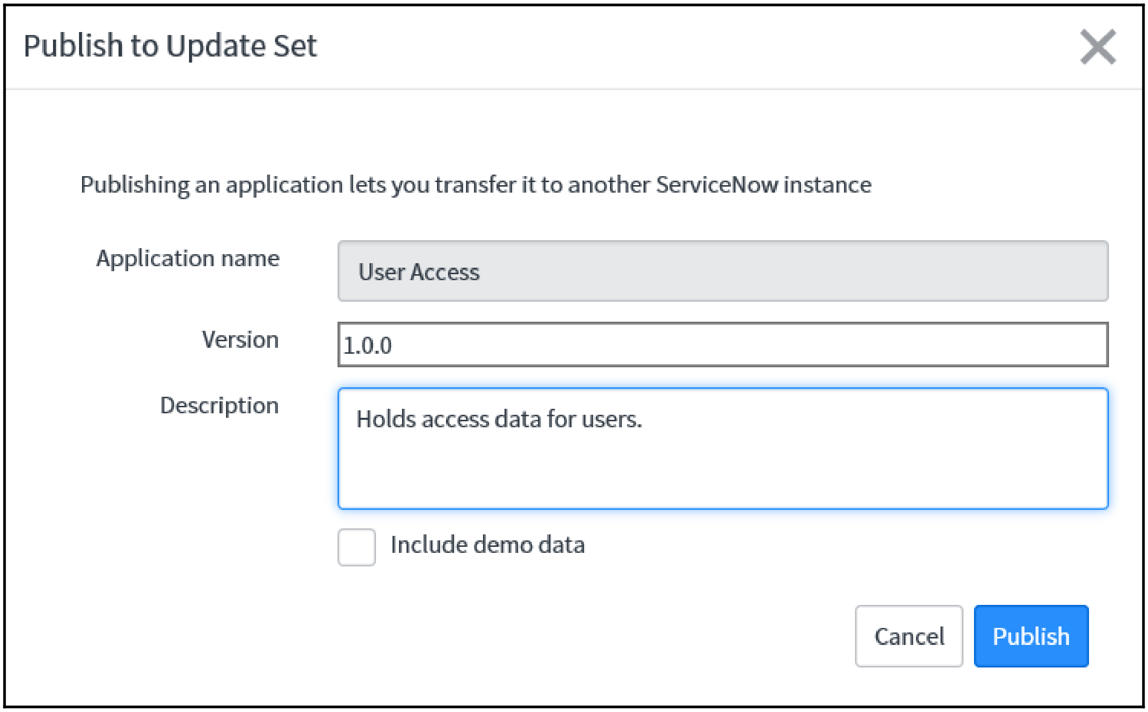
Description:

☑ Include demo data


Progress

Publishing Application: Test Application **Succeeded 100%**

Successfully published 'Test Application' update set - Succeeded in 0 Seconds

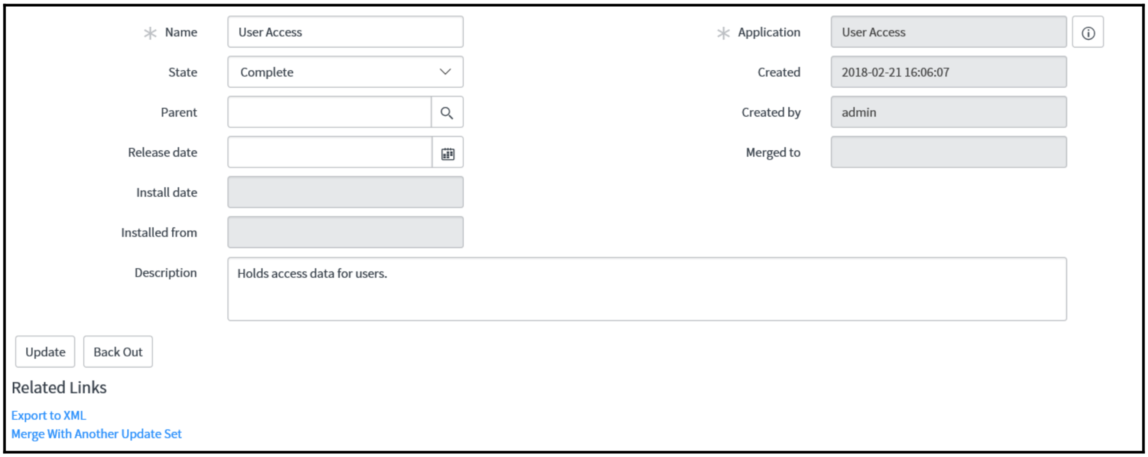


Publish to Update Set

Publishing an application lets you transfer it to another ServiceNow instance

Application name	User Access
Version	1.0.0
Description	Holds access data for users.

□ Include demo data



Name	User Access	Application	User Access
State	Complete	Created	2018-02-21 16:06:07
Parent		Created by	admin
Release date		Merged to	
Install date			
Installed from			
Description	Holds access data for users.		

Related Links

Export to XML

Merge With Another Update Set