ServiceNow 201:

5 - GlideSystem

Course Outline

1 Course Introduction

2 Development Overview

3 Scripting Locations

4 GlideRecord

5 GlideSystem

6 GlideForm & GlideUser

7 GlideAjax

8 Exploring Other APIs

9 Becoming A Scripting Master

10 Creating A Custom App

Section Outline

1. GlideSystem Introduction
2. Show Me The Code!
3. GlideSystem API Overview
4. Common GlideSystem Methods
5. Message & Log Types
6. getMessage, getProperty, & nil Methods
7. Concept: Events
8. eventQueue Method
9. GlideSystem Demo
10. Section Recap

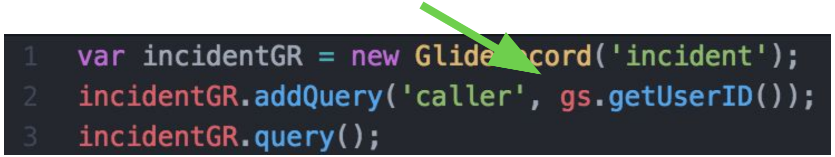
GlideSystem Introduction

- Server-side
- System and user data
- Referenced by gs
- Doesn’t need to be invoked
- Helper methods

> *The GlideSystem API provides a number of convenient methods to get information about the system, the current logged in user, etc.*
>
> ServiceNow Docs

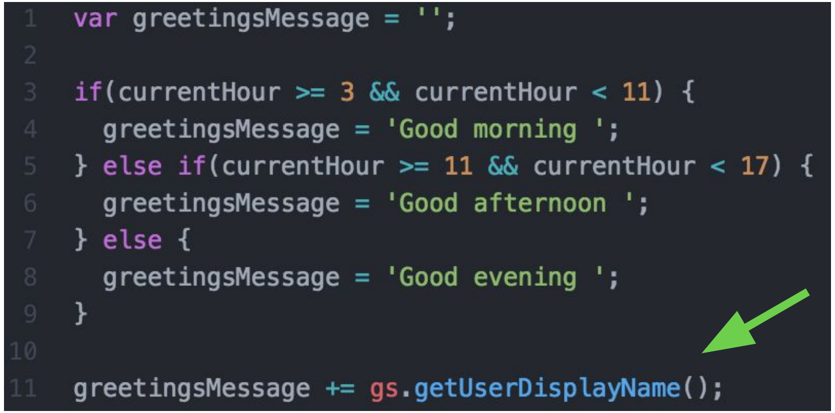
Show Me The Code!

- Print a list of all incidents where the caller is the current user



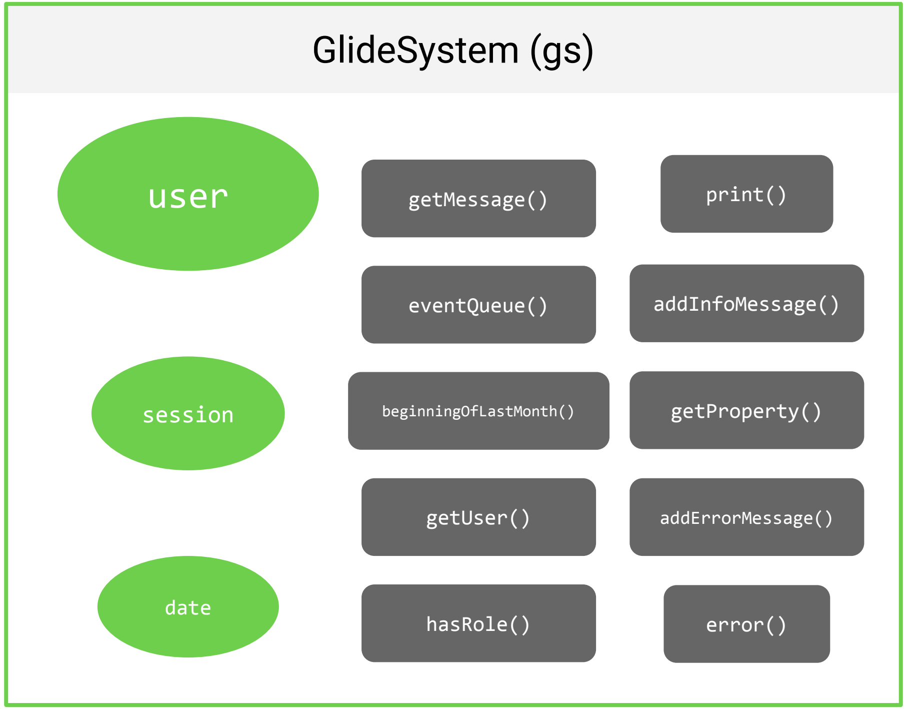
```
var incidentGR = new GlideRecord('incident');
incidentGR.addQuery('caller', gs.getUserID());
incidentGR.query();
```

- Create a user greetings



```
var greetingsMessage = '';

if(currentHour >= 3 && currentHour < 11) {
  greetingsMessage = 'Good morning ';
} else if(currentHour >= 11 && currentHour < 17) {
  greetingsMessage = 'Good afternoon ';
} else {
  greetingsMessage = 'Good evening ';
}

greetingsMessage += gs.getUserDisplayName();
```

GlideSystem API Diagram

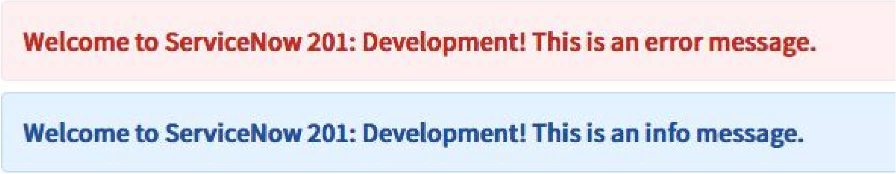


Common GlideSystem Methods

- addErrorMessage()
- addInfoMessage()
- eventQueue()
- getMessage()
- getProperty()
- getSession()
- print()
- error()
- log()
- debug()
- info()
- warn()
- generateGUID()
- getUser()
- hasRole()
- isLoggedIn()
- isMobile()
- nil()
- setRedirect()
- setProperty()

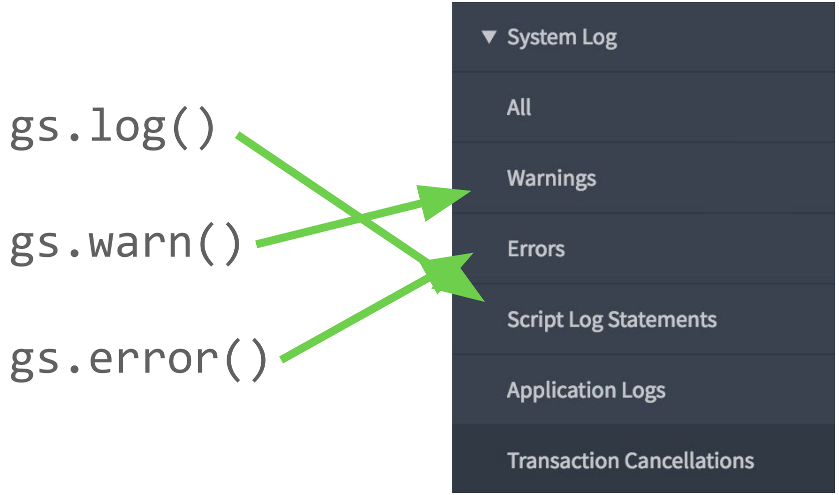
Message & Log Types

```
gs.addErrorMessage()
gs.addInfoMessage()
```

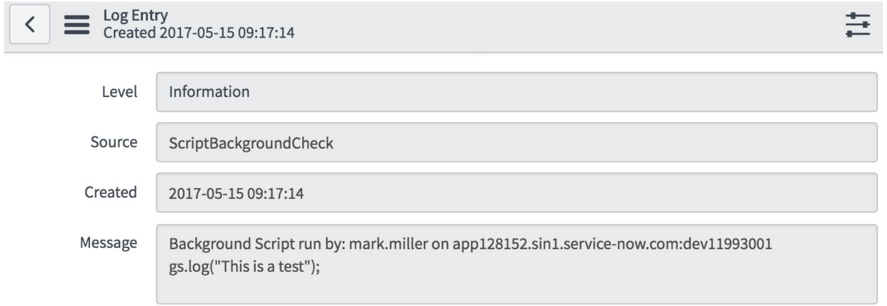


Welcome to ServiceNow 201: Development! This is an error message.

Welcome to ServiceNow 201: Development! This is an info message.



```
gs.log()
gs.warn()
gs.error()
```

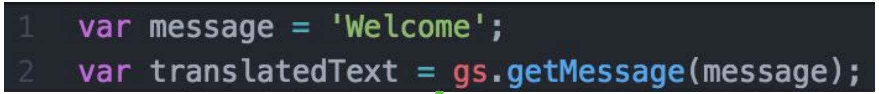


Created 2017-05-15 09:17:14

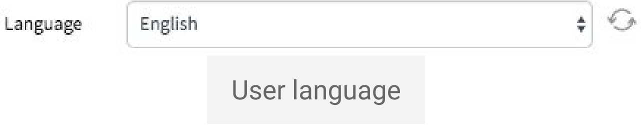
Level	Information
Source	ScriptBackgroundCheck
Created	2017-05-15 09:17:14
Message	Background Script run by: mark.miller on app128152.sin1.service-now.com:dev11993001 gs.log("This is a test");

GlideSystem getMessage() Method

- Provides a way to translate text easily
- Uses the sys_ui_message table
- First argument is the string of the message
- System uses current user language as another parameter

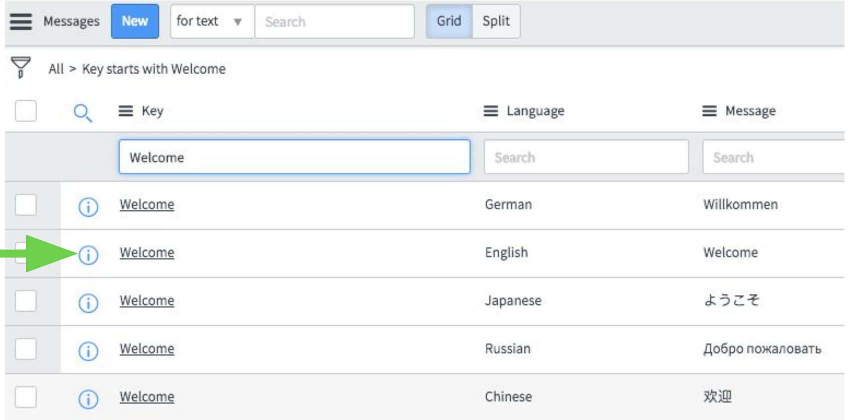


```
var message = 'Welcome';
var translatedText = gs.getMessage(message);
```



Language: English

User language

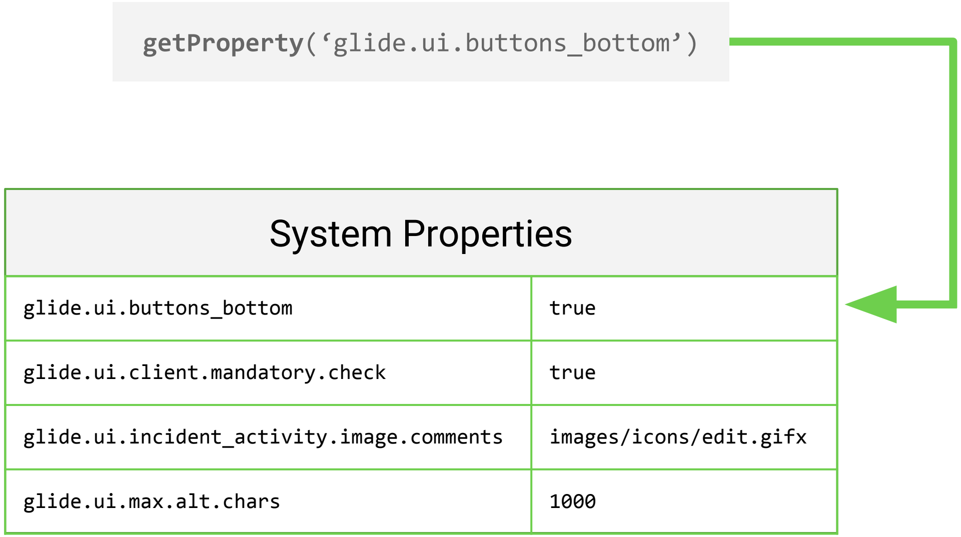


All > Key starts with Welcome

Key	Language	Message
Welcome	German	Willkommen
Welcome	English	Welcome
Welcome	Japanese	ようこそ
Welcome	Russian	Добро пожаловать
Welcome	Chinese	欢迎

GlideSystem `getProperty()` Method

- Get a system property
- Best practice instead of hard-coding

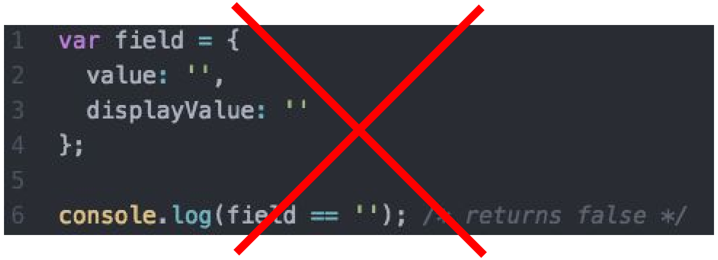


```
getProperty(‘glide.ui.buttons_bottom’)
```

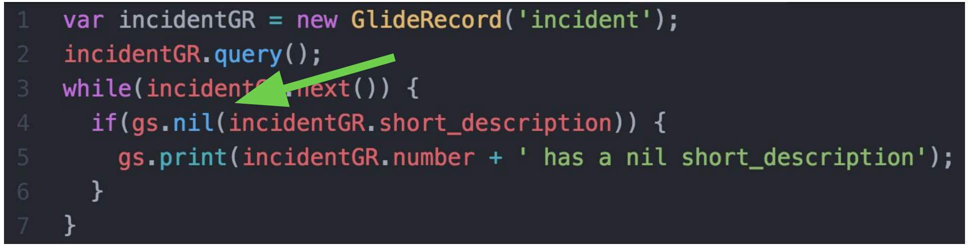
System Properties	
glide.ui.buttons_bottom	true
glide.ui.client.mandatory.check	true
glide.ui.incident_activity.image.comments	images/icons/edit.gifx
glide.ui.max.alt.chars	1000

GlideSystem `nil()` Method

- Every field in a GlideRecord is an object
- Empty string comparison doesn't always work



```
var field = {
  value: '',
  displayValue: ''
};

console.log(field == ''); /* returns false */
```



```
var incidentGR = new GlideRecord('incident');
incidentGR.query();
while(incidentGR.next()) {
  if(gs.nil(incidentGR.short_description)) {
    gs.print(incidentGR.number + ' has a nil short_description');
  }
}
```

Concept: Events

User Events

- Mouse click
- Mouse movement
- Key press

System Events

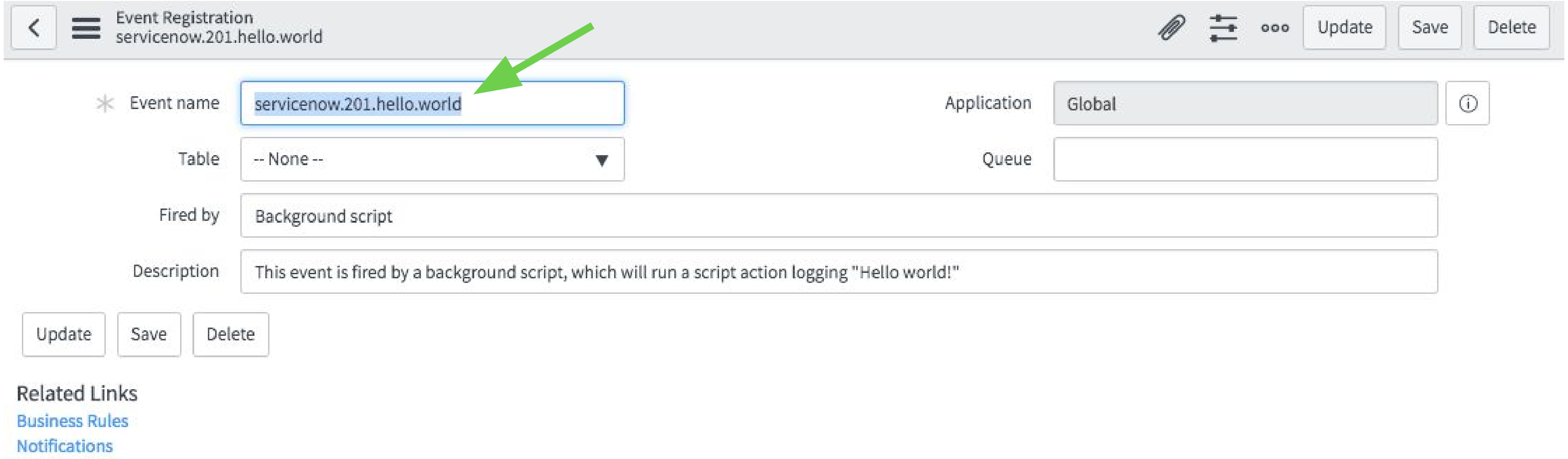
- Time
- Database operation
- Network activity

Events in ServiceNow

- Can be triggered by:
  - CRUD operations on tables (Business Rules)
  - Workflow activities
  - Scripts (Scheduled Jobs, Web Services, etc.)
- Must be registered
- Event components:
  - Registry
  - Event action (Script Action, Notification)
  - Event log

101 This topic was discussed in ServiceNow 101: Administration

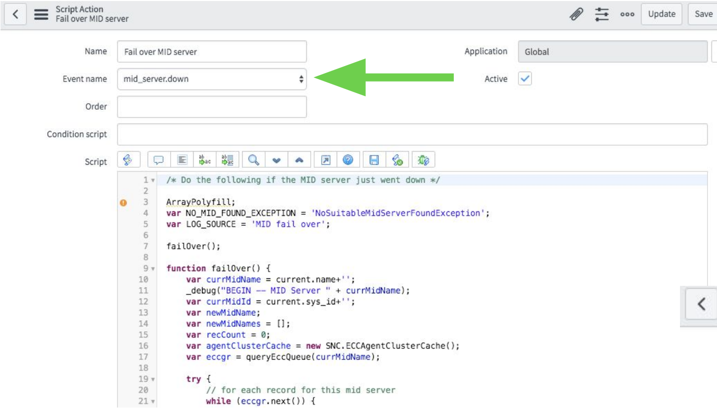
Event Component: Registry



✱ Event name	servicenow.201.hello.world	Application	Global
Table	-- None --	Queue	
Fired by	Background script		
Description	This event is fired by a background script, which will run a script action logging "Hello world!"		

Event Component: Event Action

Script Action



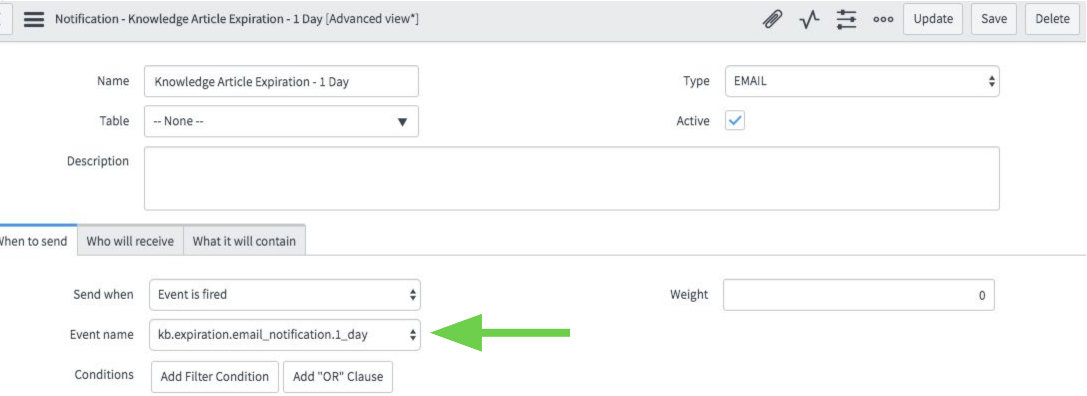
Script Action
Fail over MID server

Name	Fail over MID server	Application	Global
Event name	mid_server.down	Active	☑
Order			
Condition script			

Script

```
/* Do the following if the MID server just went down */

ArrayPolyfill;
var NO_MID_FOUND_EXCEPTION = 'NoSuitableMidServerFoundException';
var LOG_SOURCE = 'MID fail over';

failOver();

function failOver() {
    var currMidName = current.name+'';
    _debug("BEGIN -- MID Server " + currMidName);
    var currMidId = current.sys_id+'';
    var newMidName;
    var newMidNames = [];
    var recCount = 0;
    var agentClusterCache = new SNC.ECCAgentClusterCache();
    var eccgr = queryEccQueue(currMidName);

    try {
        // for each record for this mid server
        while (eccgr.next()) {
```

Notification

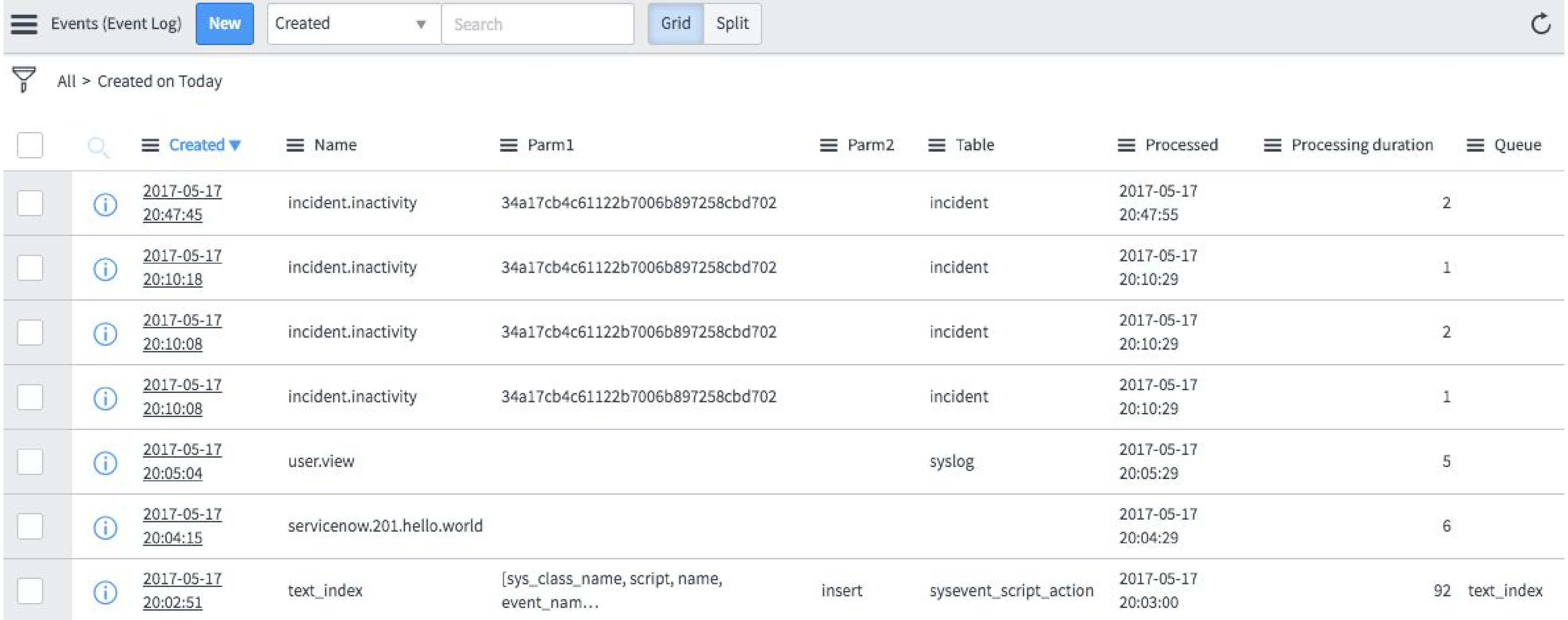


Notification - Knowledge Article Expiration - 1 Day [Advanced view*]

Name	Knowledge Article Expiration - 1 Day	Type	EMAIL
Table	-- None --	Active	☑
Description			

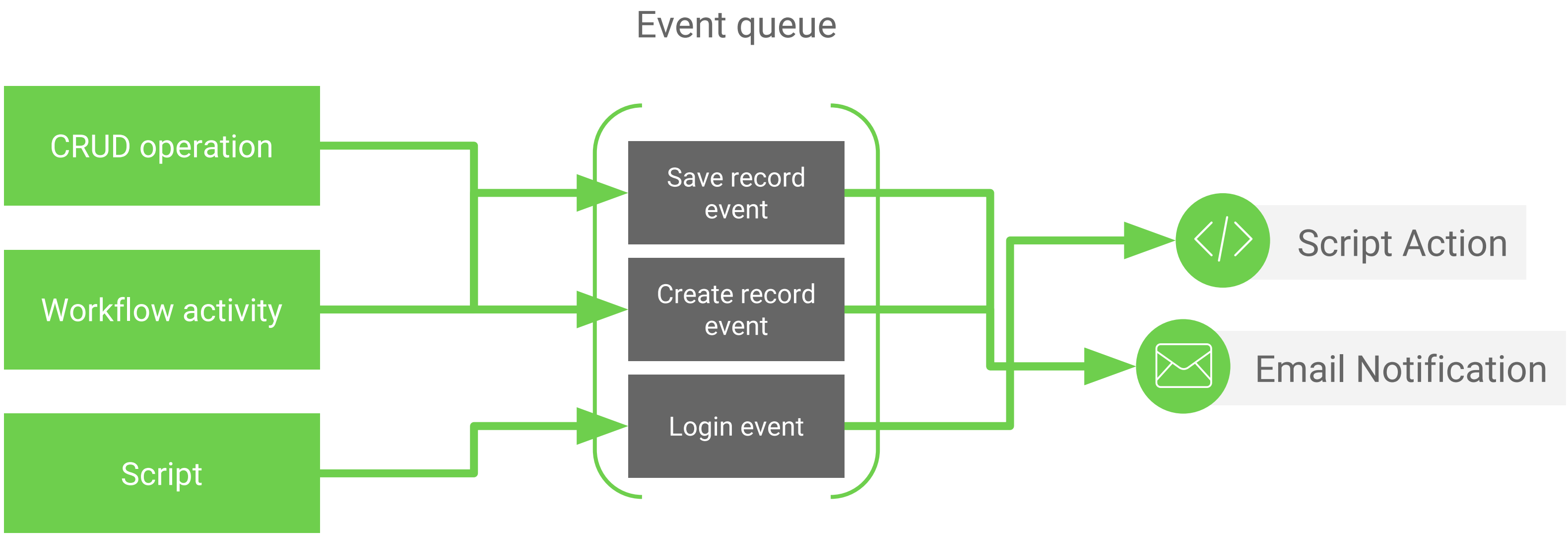
Send when	Event is fired	Weight	0
Event name	kb.expiration.email_notification.1_day		
Conditions	Add Filter Condition	Add "OR" Clause	

Event Component: Event Log



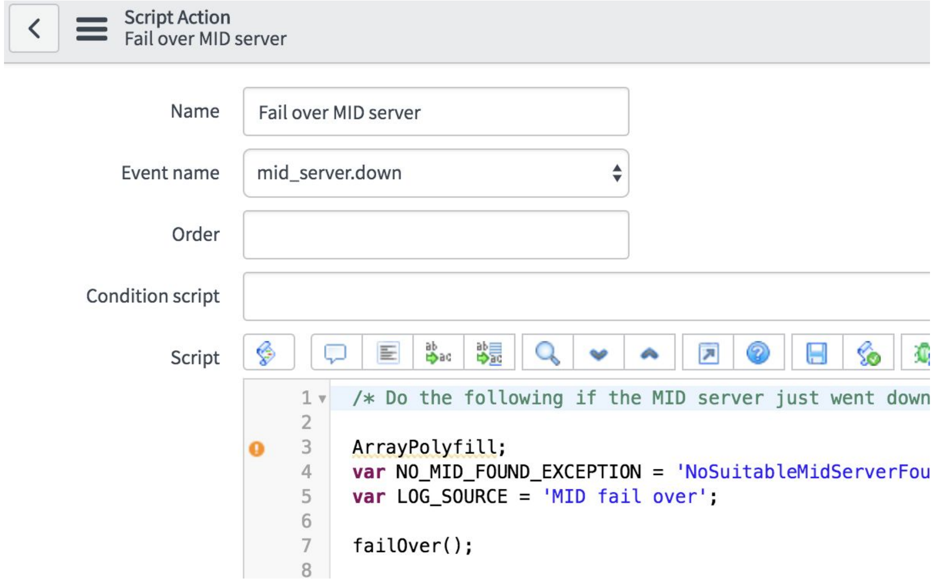
Created	Name	Parm1	Parm2	Table	Processed	Processing duration	Queue
2017-05-17 20:47:45	incident.inactivity	34a17cb4c61122b7006b897258cbd702		incident	2017-05-17 20:47:55	2	
2017-05-17 20:10:18	incident.inactivity	34a17cb4c61122b7006b897258cbd702		incident	2017-05-17 20:10:29	1	
2017-05-17 20:10:08	incident.inactivity	34a17cb4c61122b7006b897258cbd702		incident	2017-05-17 20:10:29	2	
2017-05-17 20:10:08	incident.inactivity	34a17cb4c61122b7006b897258cbd702		incident	2017-05-17 20:10:29	1	
2017-05-17 20:05:04	user.view			syslog	2017-05-17 20:05:29	5	
2017-05-17 20:04:15	servicenow.201.hello.world				2017-05-17 20:04:29	6	
2017-05-17 20:02:51	text_index	[sys_class_name, script, name, event_nam...	insert	sysevent_script_action	2017-05-17 20:03:00	92	text_index

Process: Event Queue



Script Actions

- Scripts invoked via registered events
- Server side
- Scope
  - current
  - event
    - parm1
    - parm2



Script Action
Fail over MID server

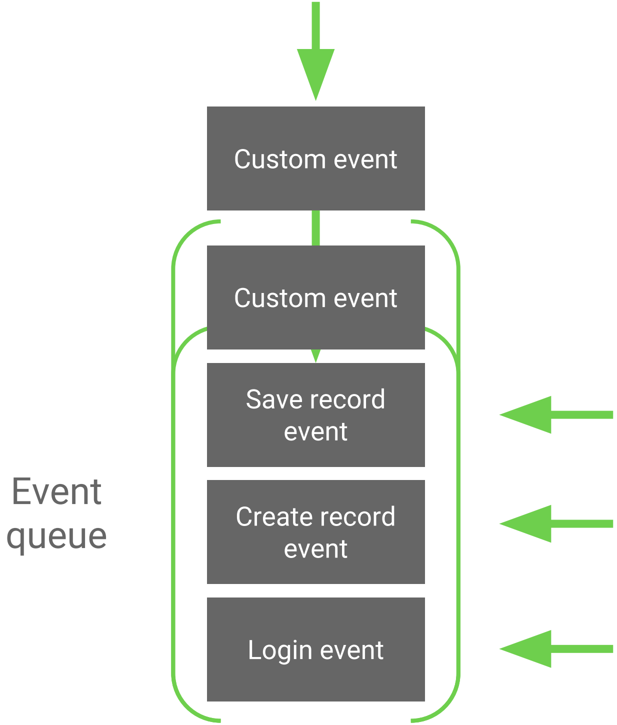
Name	Fail over MID server
Event name	mid_server.down
Order	
Condition script	
Script	

```
/* Do the following if the MID server just went down

ArrayPolyfill;
var NO_MID_FOUND_EXCEPTION = 'NoSuitableMidServerFou
var LOG_SOURCE = 'MID fail over';

failOver();

```

GlideSystem `eventQueue()` Method

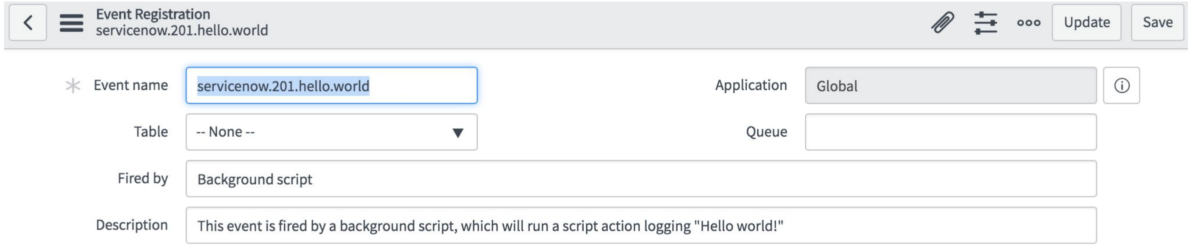
- Add events to the event queue
- Parameters
    - Event name
    - GlideRecord object
    - Optional parameter
    - Optional parameter

```
eventQueue('event.name', current, parm1, parm2)
```



Demo: Log *Hello world!* Using Events

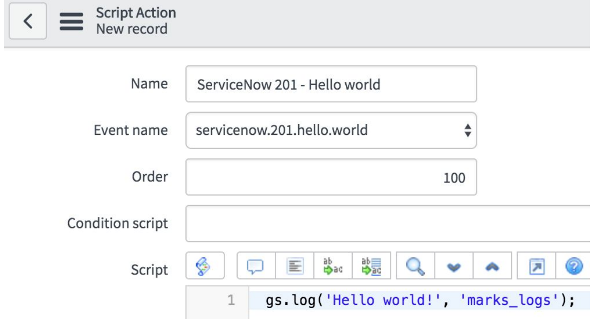
1 Create event registry



Event Registration
servicenow.201.hello.world

Event name	servicenow.201.hello.world	Application	Global
Table	-- None --	Queue	
Fired by	Background script		
Description	This event is fired by a background script, which will run a script action logging "Hello world!"		

2 Create Script Action

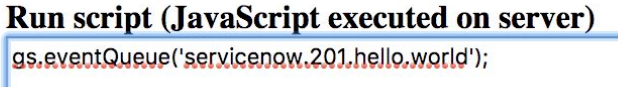


Script Action
New record

Name	ServiceNow 201 - Hello world
Event name	servicenow.201.hello.world
Order	100
Condition script	
Script	

```
gs.log('Hello world!', 'marks_logs');
```

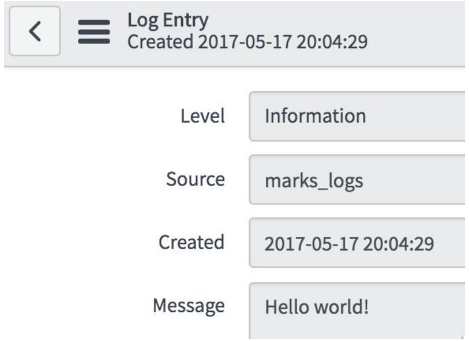
3 Run Background script



Run script (JavaScript executed on server)

```
gs.eventQueue('servicenow.201.hello.world');
```

4 Check logs



Log Entry
Created 2017-05-17 20:04:29

Level	Information
Source	marks_logs
Created	2017-05-17 20:04:29
Message	Hello world!

Where Can I Use This?

Client-Side

- GlideForm
- GlideUser
- GlideAjax

- Client Scripts
- UI Actions
- UI Policies
- Service Portal

Server-Side

- GlideRecord
- GlideSystem
- GlideDateTime

- Business Rules
- UI Actions
- Script Includes
- Scheduled Jobs
- Service Portal
- Web Services
- Workflows

Section Recap

- Use the GlideSystem API to access helper methods and data about the current user and session
- Reference GlideSystem in any server-side script by using `gs`
- Events in ServiceNow have 3 main components
  - Event registry
  - Event action (Script Action, Notification)
  - Event log
- Use the `eventQueue()` method to add registered events to the event queue
- Use the `nil()` method when checking if a JavaScript value is empty