SuiteScript

The Basics

What is SuiteScript?

SuiteScript is a JavaScript-based API that gives developers the ability to extend NetSuite beyond the capabilities provided through SuiteBuilder point-and-click customization.

The goal of the course

At the end of the course, the goal is to have better understanding of SuiteScript and to be able to create simple SuiteScript customizations.

Outline

- Script Type Overview
- Using Suitescript you can...
- How do I run a Script in NetSuite?
- Creating a simple Client-Side Suitescript
- Debugging a Script
- SuiteScript API

Outline

- SuiteScript Objects
- Reference
- Questions

Script Types Overview

User Event Scripts:

User Event scripts are triggered when users work with records and data changes in NetSuite as they create, open, update, or save records.
User Event scripts are useful for customizing the workflow and association between your NetSuite entry forms. These scripts can also be used for doing additional processing before records are entered or for validating entries based on other data in the system.

Suitelets:

Suitelets enable the creation of dynamic web content.
Suitelets can be used to implement custom front and backends.
Through API support for scripting forms and lists, these Suitelets can also be used to build NetSuite-looking pages.

Script Types Overview

RESTlets:

RESTlets are server-side scripts that can be used to define custom, RESTful integrations to NetSuite.

RESTlets follow the principles of the REST architectural style and use HTTP verbs, HTTP headers, HTTP status codes, URLs, and standard data formats. They operate in a request-response model, and an HTTP request to a system-generated URL invokes each RESTlet.

Script Types Overview

Client Scripts:

Client scripts are executed on the client. These scripts can be attached to and run on individual forms, or they can be deployed globally and executed on entity and transaction record types. Global client scripts enable centralized management of scripts that can be applied to an entire record type.

Portlet Scripts:

Portlet scripts are used to create custom dashboard portlets. For example, you can use SuiteScript to create a portlet that is populated on-the-fly with company messages based on data within the system.

Scheduled Scripts:

Scheduled scripts are executed on-demand in real-time or via a user-configurable schedule. Scheduled scripts are useful for batch processing of records.

Script Types Overview

Mass Update Scripts: Mass update scripts allows you to programmatically perform custom mass updates to update fields that are not available through general mass updates. You can also use action scripts to run complex calculations, as defined in your script, across many records.

Workflow Action Scripts: Workflow action scripts allow you to create custom actions that are defined on a record in a workflow.

Bundle Installation Scripts: Bundle installation scripts fire triggers that execute as part of bundle installation, update, or uninstall. Trigger execution can occur either before install, after install, before update, after update, or before uninstall. These triggers automatically complete required setup, configuration, and data management tasks for the bundle.

Using SuiteScript you can:

The following are some of the uses of SuiteScript.

- Perform custom business processing when NetSuite records are updated, created, deleted (using User Event Scripts).
- Perform custom validations and calculations in the browser client (using Client Scripts).
- Create custom user interfaces (using script types such as Suitelets or User Event Scripts and UI Objects).
- Run batch processes (using Scheduled Scripts).
- Execute NetSuite searches (using script types such as User Event Scripts or Scheduled Scripts).
- Perform various utility processing such as sending email and faxes, creating and uploading files, or working with XML documents (using script types such as User Event Scripts orSuitelets).
- Create custom dashboard portlets (using Portlet Scripts).

How do I run a Script in NetSuite?

This process includes:

1. Creating a JavaScript file for your script.
2. Uploading the file into NetSuite File Cabinet.
3. Creating a NetSuite “Script” record for your script.
4. Defining script runtime options on a NetSuite Script Deployment page.

Creating a simple Client-Side SuiteScript

Customizing the Form.

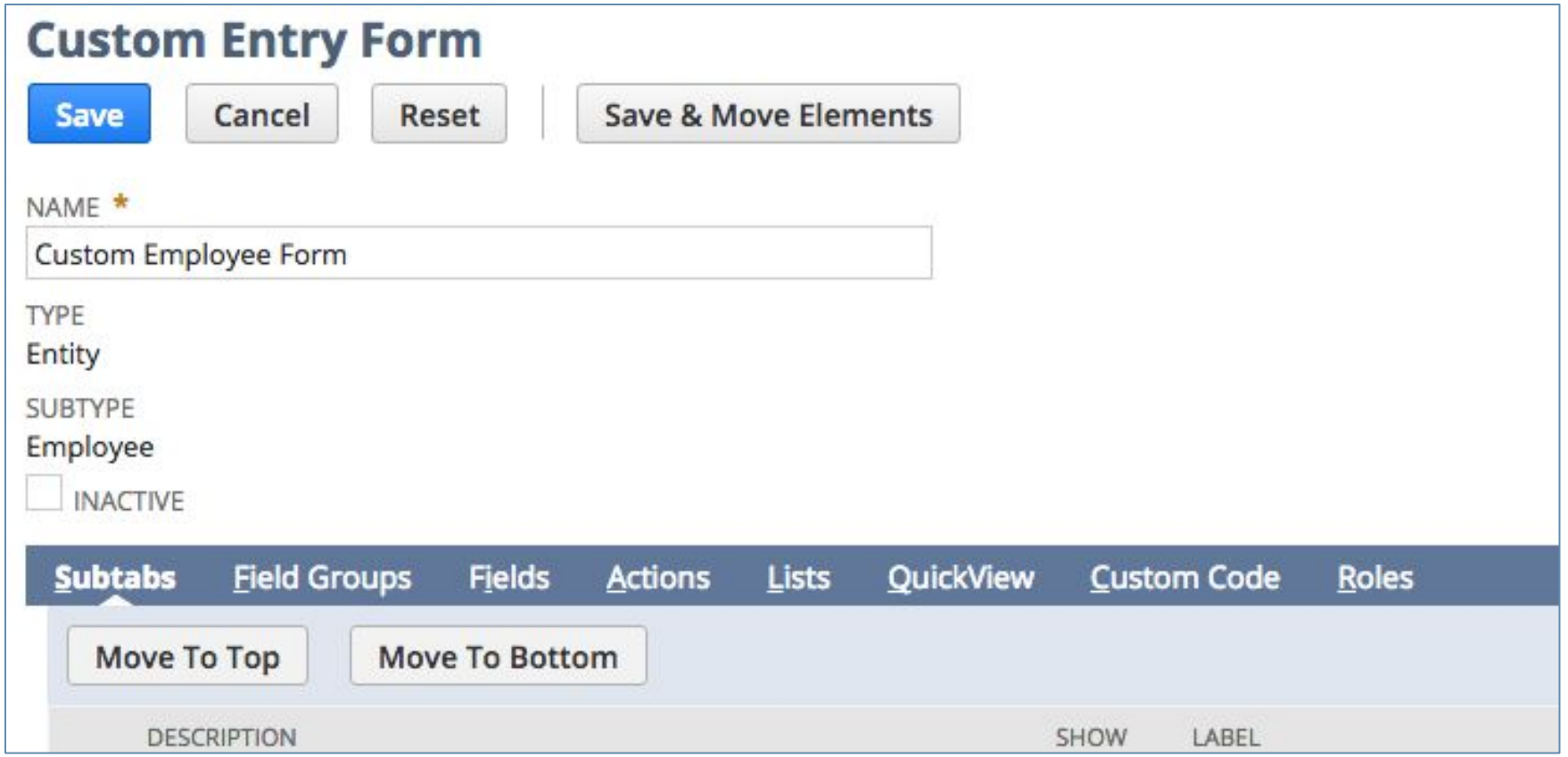
1. How to access the custom entity form screen
2. Form Customization

1. How to access the custom entity form screen

Go to Customization>Forms>Entry Forms>Click Customise

Go to the record. Click Edit. Click Customise Form

2. Form Customization



Creating a client-side Suitescript using Eclipse

1. Creating a project in Eclipse
2. Add a script file in Eclipse
3. Creating a client event function
4. Upload to a file cabinet
5. Apply the script to the form
6. Test

1. Creating a project in Eclipse

2. Add a script file in Eclipse

3. Create a Client Event Function

4. Upload to the File Cabinet

5. Apply the Script to the Form

6. Test

Client-Side Script Debugger

Use Firefox's Firebug, troubleshooting of client-side script

Use the script debugger of Internet Explorer, troubleshooting of client-side script

Debugging in Firefox

Debugging in Internet Explorer

Server-Side Script Debugger

1. **Log in to the debugger domain**

   URL https://debugger.netsuite.com

Server-Side Script Debugger

2. Starting the script debugger

Customization > Scripting > Script Debugger

3. Select the Script File

4. Start of User Operation

Server-Side Script Debugger

User-Event Function

Due to the operation of the end-user function, the-start-up of the script

5. Script Debugging

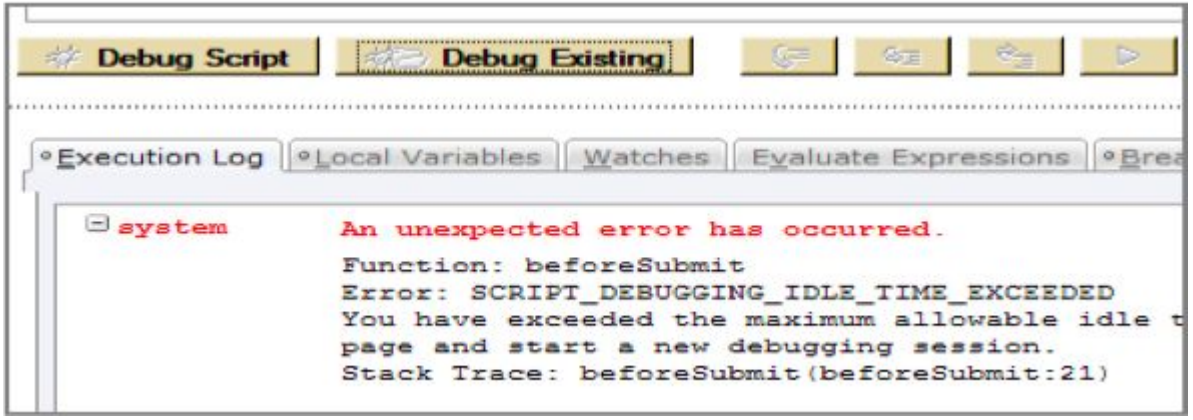
Debugging Tips

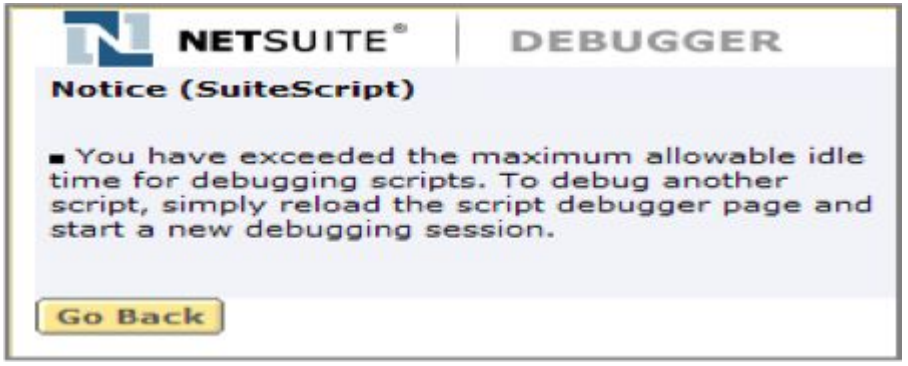
- Script you are running event in debug mode, will affect the actual data

- Log of nlapiLogExecution that were performed during the debugging run, rather than the script deploy screen, will be recorded in the debugger execution log tab

Debugging Tips

- If the execution of the script is not performed over a long period of time , you will see a time-out message





SuiteScript API

SuiteScript is a JavaScript API that enables you to programmatically access most NetSuite records, custom records, and event/trigger points. This API is a standard NetSuite feature.

DOM and SuiteScript API

DOM and SuiteScript API

Get and Set the value of a field

Enable / Disable Field

SuiteScript Objects

SuiteScript objects are classified into the following two categories

Standard Objects:

Standard objects are used more for manipulating backend data and to handle form GET and POST processing.

UI Objects:

UI Objects are a collection of objects that can be used as a UI toolkit for server scripts such as Suitelets and user event scripts. UI objects encapsulate scriptable user interface components such as NetSuite portlets, forms, fields, lists, sublists, tabs, and columns. They can also encapsulate all components necessary for building a custom NetSuite-looking assistant wizard.

Reference

SuiteScript Developer and Reference Guide

https://system.na1.netsuite.com/help/helpcenter/en_US/PDF/SSDevAndRefGuide.pdf

Questions?