High Level Review of Netsuite's Workflow (Suiteflow) Solution

Please be aware that this document is a gathering of knowledge with respect to Netsuite's Suiteflow Functionality – To access the links within this document you will need to have access to Netsuite itself

Workflows

Pros –

- Relatively easy to build without deep technical knowledge, provide visualization, and easier to review, update and manage than scripts.
- Great for organizations without heavy IT staff.
- Great for manual review processes and data validation/enforcement.
- Can be combined with workflow action scripts to provide more customized functionality.

Cons –

- Function very linearly, so may run more slowly and less elegantly than scripts.
- Not always entirely intuitive and not all actions are available for all record types.
- Require a fair amount of computational thinking to setup properly and can require lots of additional actions/transitions.
- Provide limited functionality for line level data manipulation.

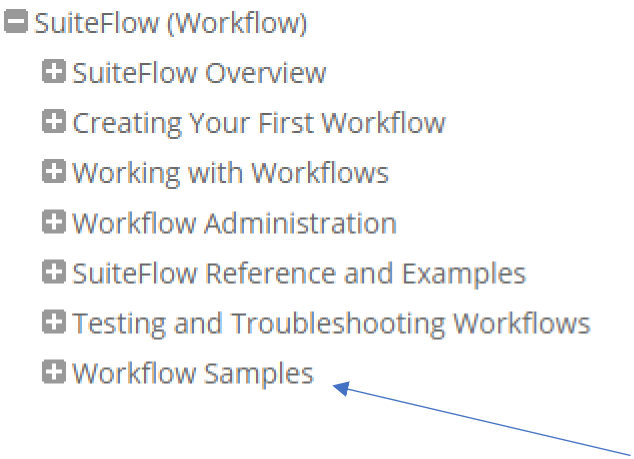
Note that Sublists are not available in a workflow. You can only manipulate and touch body fields of a record. However, there are some ways of using saved searches, which you can set up as a condition of a workflow state or transition. The saved search functionality can use the sublist values to define criteria for your business process.

	Workflows	Scripts
Easy to build, check & update for non-technical staff	Yes	No
Difficulty in isolating & testing specific functionality	Easy	Hard if script is complex
Speed	Relatively Fast, but linear	May run Faster / Slower depending on complexity
Allows for modification/control of line-level data?	*Yes – this is new but limited functionality in sub-list action groups	Yes
Uses workflow states, which can be identified and used for saved search criteria	Yes	No – can only manipulate custom fields
Can utilize criteria from related records (one-record deep)	Yes	Yes
Can transform records or create new ones?	Yes	Yes

Be aware of the manual - SuiteFlow User Guide –

https://docs.oracle.com/cloud/latest/netsuitecs_gs/NSSFU/NSSFU.pdf

Log into Netsuite – click on 



This manual has several workflow examples:- “Workflow Samples”

- Lead Nurturing
- Lead that did not convert to customer within 3 days
- Estimate Approval Routing
- Travel Approval Workflow
- Welcome email sent to customers 3 days after first order

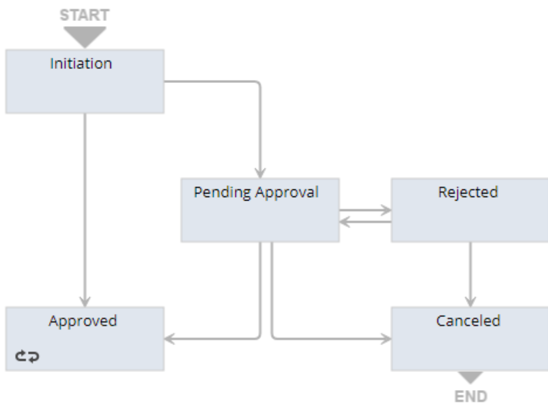
Workflow Templates provided within the Netsuite solution:-

- Journal Entry Approval
- Purchase Order Approval
- Sales Order Approval
- Lead Nurturing

When developing a workflow one needs to consider what you want to do:-

- Create workflow action on loading up of a record – aka put a button on the form itself;
- Create workflow action during data entry of a record;
- Create workflow action on saving up a record – aka sending an email as a result of record details;
- Create an approval process

Consider using flowcharts to document your business process and what you want the workflow to do



On the transition (lines connecting the states) – you indicate that an order is approved unless a certain condition is reached – and then it goes to Pending Approval. Such condition could be customer open balance is NIL, AND total order value is less than $xx, OR user has no supervisor.

Workflow Examples

Simple:-

- Default a value on a form itself
- Hide fields if a user role views those records – aka GM% on sales orders

Complex:-

Involves multiple stages, transitions and actions

- Routing Expense reports for approvals – say to finance initially who reviews receipts and then to actual manager who approves the spend knowing that the receipts have been checked
- Approving Sales Orders to ensure Sales Reps have achieved expected margins. Can consider allowing for auto approvals if margin of xx has been achieved to reduce the volume of orders that need to be reviewed
- New Customer Project is created and info is required in order to complete the creation of this project – things like manager, budget, who is authorised to work on the project etc.

Robust vendor bill approval process.

- Instead of selecting specific individuals in the workflow for each scenario, ensure it is more flexible
- Consider creating two list/record type custom fields on the native Departments record in NetSuite (using the Other Record Fields option) and called them "Department Head" and "Executive Approver".
- Set the criteria in the workflow to look at the Department field on the transaction record and join to the Department Head and/or Executive approver based on our dollar thresholds.

This is a more scalable and flexible solution, which only requires updates to the Department record fields if there are personnel changes (rather than direct modifications to each individual employee record or the workflow itself).

Example of Custom Fields

Before creating a custom field, one needs to consider whether the field should be mandatory and what impact that has on entity and/or transaction fields.

- Customers have attributes – for example what vertical they are and as a result the sales order form may need to be different. Be aware of ***GL PlugIn*** functionality to reallocate the revenues according to customer's vertical instead of the income account on the item record.
- Items have attributes – the item code can embed knowledge about the product – but you want to do a custom search on a portion of that item code. Consider custom fields to capture this information.

These custom fields could be used on a workflow as need be.

CONCEPTS

Workflow States – These are the building blocks of workflows. Where are we in the business process? States include specific actions and are connected to other states by transitions.

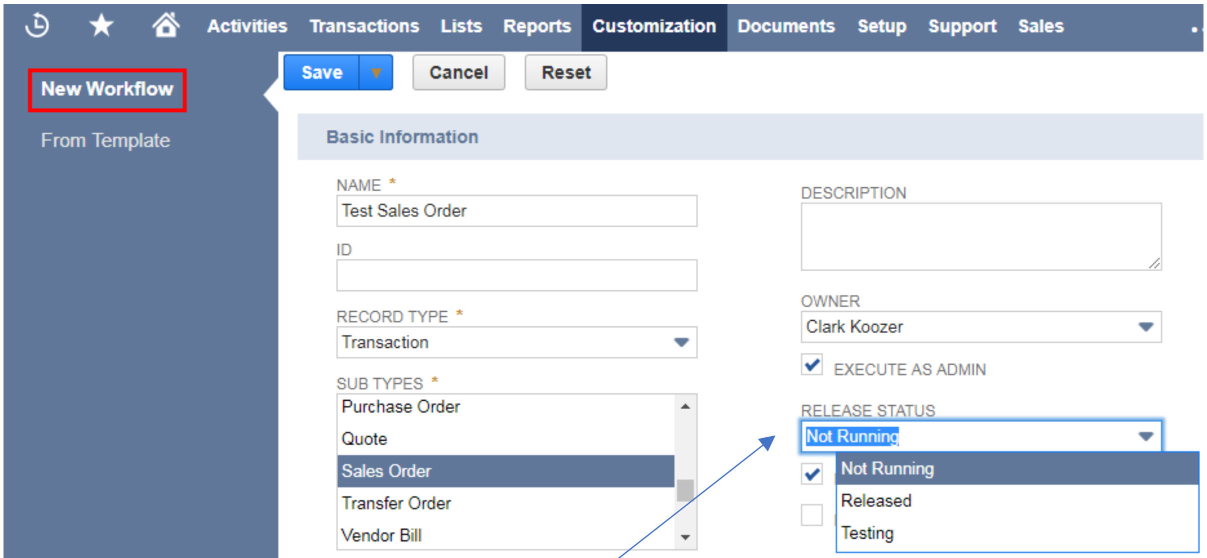
Workflow Actions – What task(s) should be performed at the current state? For example: send an email, add a button, remove a button, set a field value, go to record, return user error.

Workflow Transitions – Transitions allow for movement between states (steps) of your business process. Which state should the business process move to next? You can define conditions and trigger condition options for the most accurate process possible.

Triggers – When should a workflow be initiated, a transition occur, and/or some action be performed?

Workflow Fields – Fields work similar to other fields you would add to a record, although they live within the workflow. You can add NetSuite fields to the entire workflow or just to a particular state.

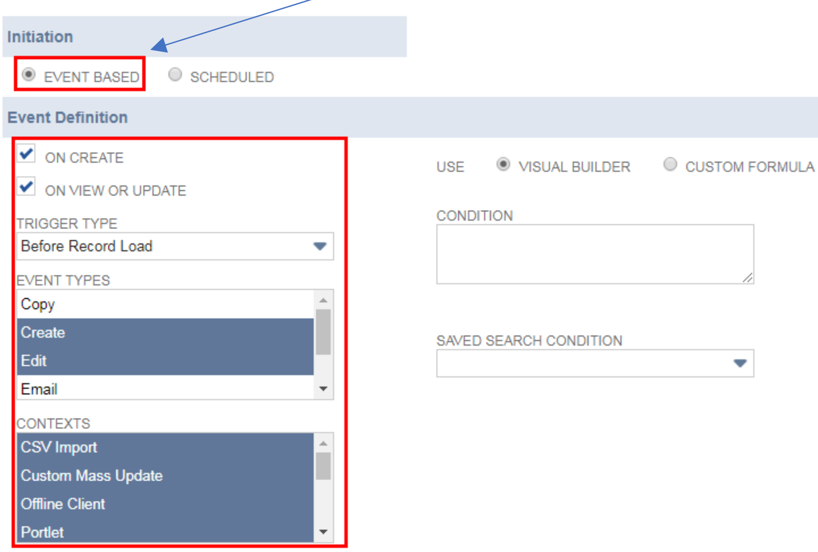
Workflows work on specific record type and can only be used to configure changes on that selected record type. Once you have selected the record type, it cannot be changed after click on save button, so it is important to make sure that you select the correct record type when initially creating your workflow.



Workflow has 3 different release status:

- **Not Running:** If release status is not running then workflow conditions cannot be triggered on that record type.
- **Released:** If release status is released then workflow conditions executes for all users.
- **Testing:** If release status is testing then workflow condition executes only for the owner of the workflow.

Initiation – Is Either Event Based or Scheduled.



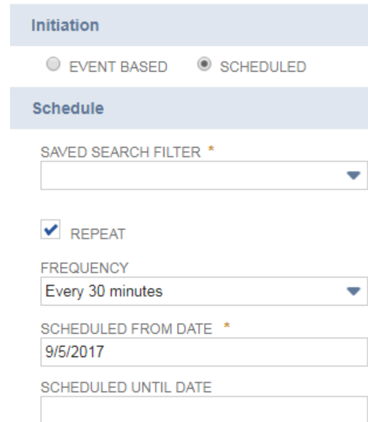
Workflows can be configured to run according to a schedule or based on user-defined criteria being met (referred to as "**Event Based**" initiation).

The above shows an event-based initiation since we want the changes to occur as the user edits the sales order record. Additionally, we'll allow the workflow to trigger both when a new sales order is created and when an existing sales order is updated using the "On Create" and "On View or Update" preferences.

Then populate the trigger type for this Workflow, for SuiteScript, we will need some scripts to check this, but in Workflow user has the ability to set them in the UI. Based on the trigger type event types will be changed.

Scheduled Based initiation:

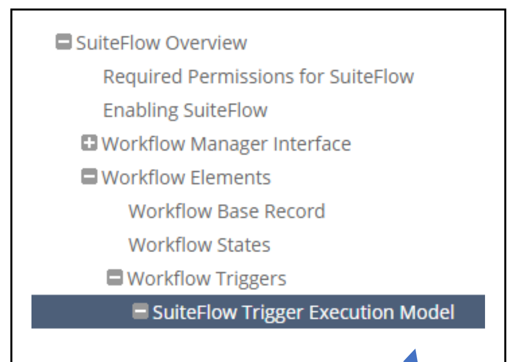
- Under initiation select the scheduled checkbox.
- Select the saved search which you want to apply as filters.
- If you want to repeat the scheduled, then check the repeat checkbox.
- Select the frequency list to run scheduled workflow, frequency list are defined as below
    - Every 30 minutes (Default)
    - Daily,
    - Weekly,
    - Monthly and
    - Yearly (the form changes based on what you select here!)
- Complete the dates as needed



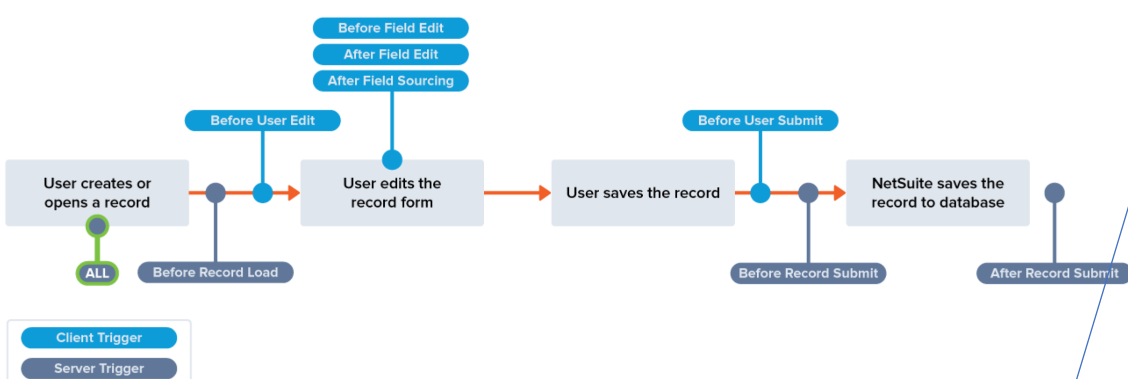
EVENT DEFINATION

<ins>**Trigger Types:-**</ins>

Triggers are based on events that occur during the processing of a record in NetSuite and dictate when the workflow should perform certain tasks. NetSuite defines triggers for when a workflow should initiate, when actions should be performed, or when a record should transition to another state for the record, based on the processing of a record in NetSuite. You direct a workflow to respond to the record processing events.

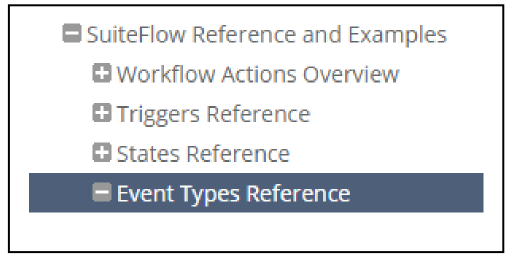


SuiteFlow includes two types of triggers, server triggers and client triggers:



<ins>**Also check out this** </ins>SuiteFlow Trigger Execution Model as per NS Help Center

Event Types:-

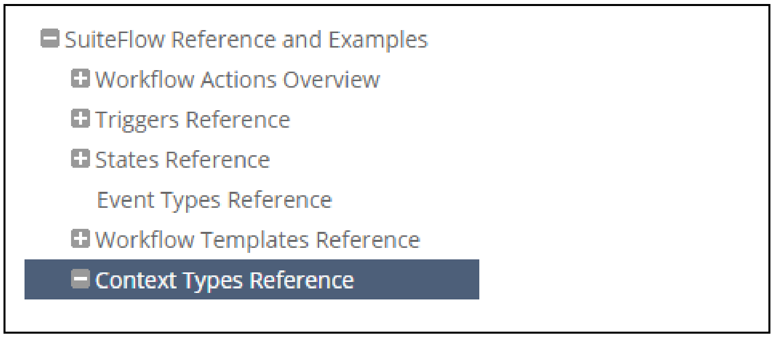


The Event Type depends on Trigger Type.

For example, approve is NOT available in the drop down when you select Before Record Load.

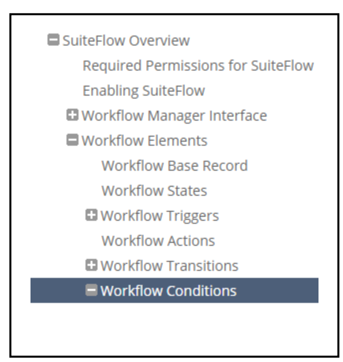
Event Type	Trigger Types (Workflow Initiation)
Approve	Before Record Submit (On Update) After Record Submit (On Update)
Cancel	Before Record Submit (On Update) After Record Submit (On Update)
Copy	Before Record Load (On Update)
Create	Before Record Load (On Create) Before Record Submit (On Create) After Record Submit (On Create)
Delete	Before Record Submit (On Update)
Drop Ship	After Record Submit (On Create)
Edit	Before Record Load (On Update) Before Record Submit (On Update) After Record Submit (On Update)
Edit Forecast	Before Record Submit (On Update)
Email	Before Record Load (On Update)
Inline Edit (XEDIT)	Before Record Submit (On Update) After Record Submit (On Update)
Mark Complete	Before Record Submit (On Update)
Order Items	After Record Submit (On Create)
Pack	Before Record Submit (On Update)
Pay Bills	After Record Submit (On Create)
Print	Before Record Load (On Update)
Reassign	Before Record Submit (On Update)
Reject	Before Record Submit (On Update)
Ship	Before Record Submit (On Update) After Record Submit (On Update)
Special Order	After Record Submit (On Create)
View	Before Record Load (On Update)

Context Types



Context	Description
CSV Import	Use CSV import to transfer data sets from other applications into NetSuite and create or update records. See CSV Imports Overview. To use this context, you must enable the Run Server SuiteScript and Trigger Workflows preference at Setup > Import/Export > CSV Import Preferences.
Custom Mass Update	Use a custom mass update SuiteScript action script to update fields that are not available through general mass updates. See Mass Update Scripts.
Portlet	Use a SuiteScript portlet script to define and publish custom dashboard content. See Portlet Scripts.
Scheduled Script	Use a SuiteScript scheduled script to create or update records. See Scheduled Scripts.
Suitelet	Use a SuiteScript Suitelet script to create or update records through a Suitelet. See Suitelets.
User Event Script	Use a SuiteScript user event script to create, load, update, copy, delete, or submit records. Note that user event scripts cannot be executed by other user event scripts. As such, workflows with the User Event Script context type cannot execute user event scripts. See User Event Scripts.
User Interface	Record created, updated, or viewed through a browser.
SOAP Web Services	Use a SOAP web service to create or update any record type supported by SOAP web services. See SOAP Web Services Supported Records. To use this context, make sure that the Disable Server SuiteScript and Workflow Triggers preference is not enabled at Setup > Integration > SOAP Web Services Preferences.
Web Store	Any records created or updated through the web store.

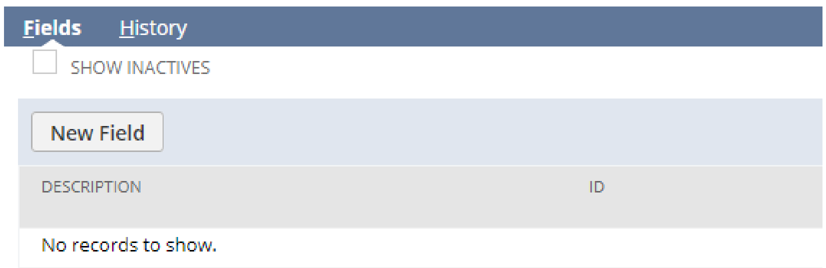
Conditions:-



You can use conditions on workflow initiation, actions, and transitions. Conditions can be basic field/value comparisons, expressions, or formulas. Use conditions to restrict for whom or when a workflow initiates or actions and transitions execute. You can use user fields in conditions to check the user, role, class, department, location, or subsidiary.

Workflow Fields:

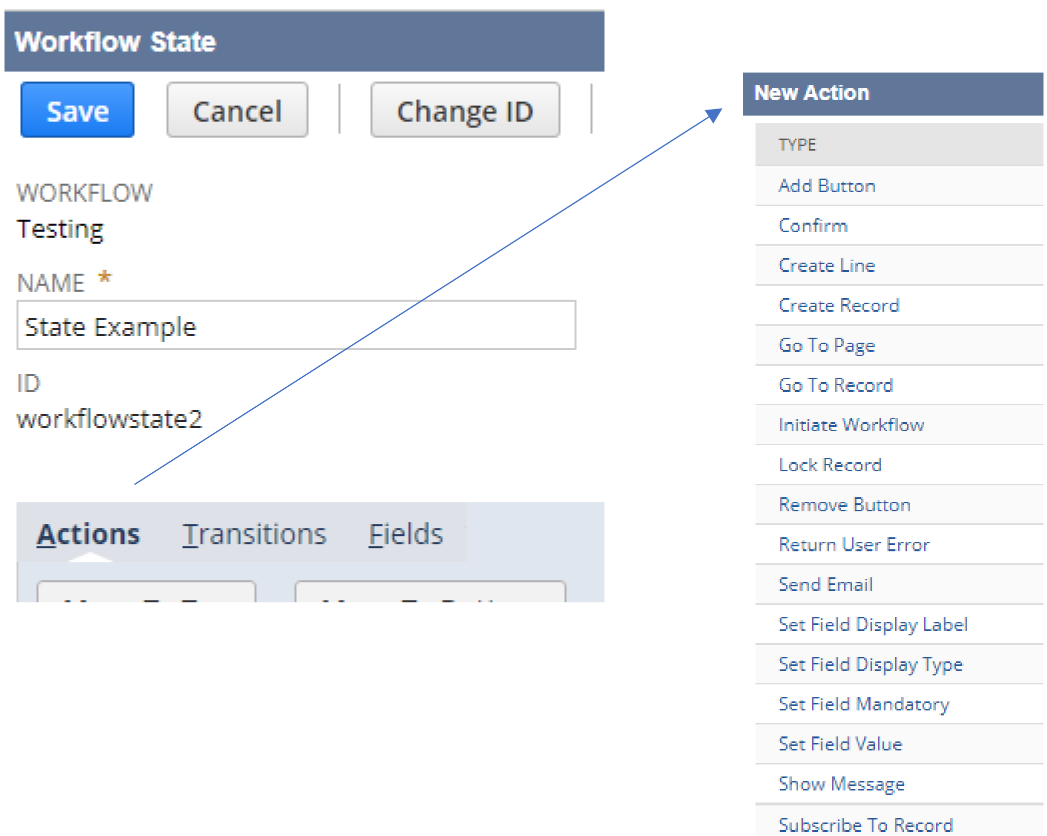
Upon saving the initial definition of a workflow, you can add fields to that workflow.



This can be done at the workflow level OR at the state level.

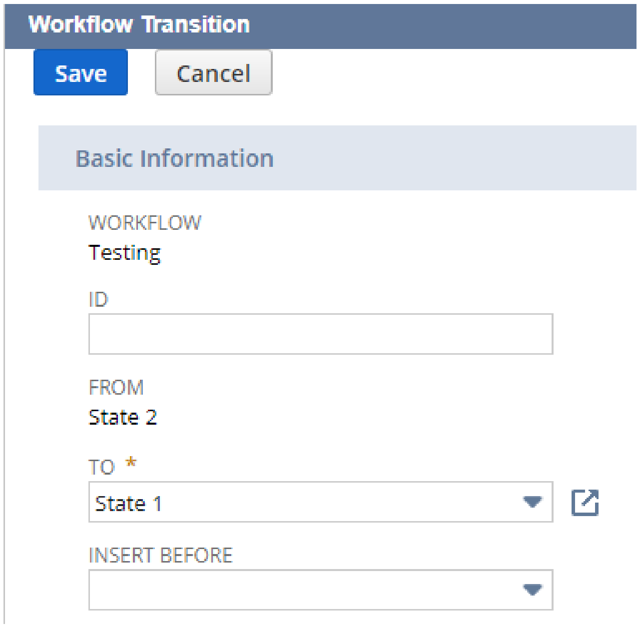
Workflow States:

These are the building blocks of workflows. A state corresponds to a stage or step in a business process. States include specific actions, and are connected to other states by transitions.



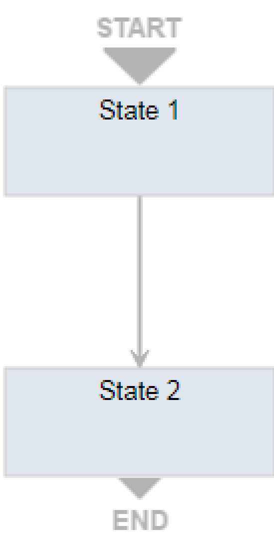
Workflow Transitions:

Transitions allow for movement between states (steps) of your business process. You can define conditions and trigger condition options for the most accurate process possible.

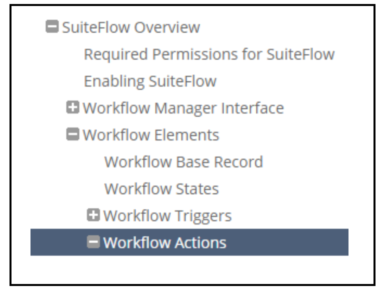


The SuiteFlow contains "States" and "Transitions",

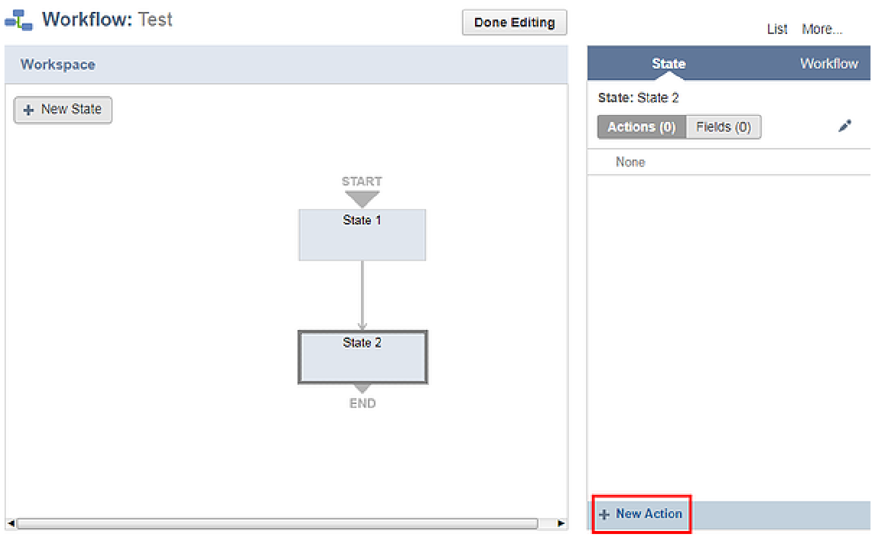
States is the Box and Transitions are the lines connect each of the states, as shown below



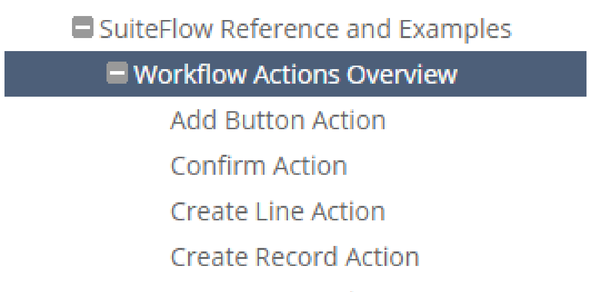
Workflow Actions:



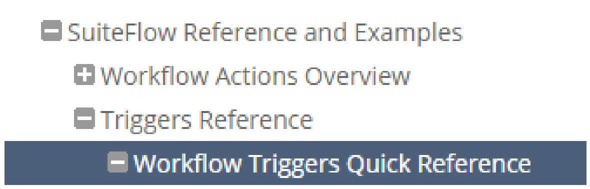
Create actions that will allow a state (steps) to execute a command. For example: send an email, add a button, remove a button, set field value, go to record, and return user error etc.



Check this out



Check this out



Conclusion:-

Suggest that one starts with a template workflow and use this as a way of educating yourself on how workflows operate. The power of Netsuite's SuiteFlow is the knowledge of what can be done.

To get anything built, it is important that this basis understanding of workflows. You need to document your business processes in a manner that allows a workflow process to be developed.