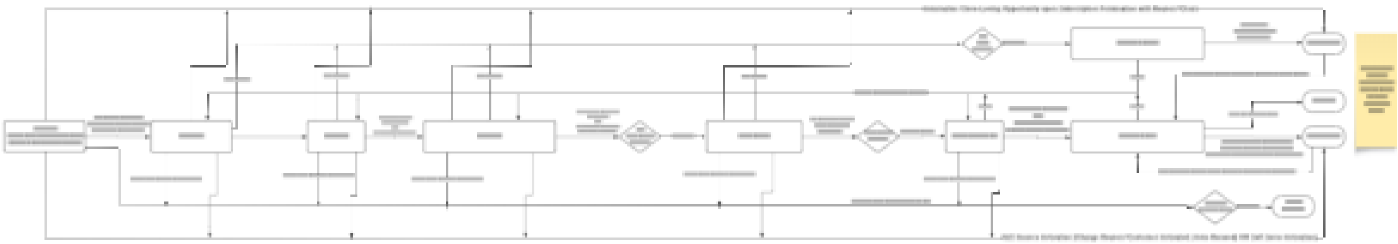
Renewal Automation

State Transition Flow (Link to Lucid Chart)



Legend:

Already In Place	
Proposed	
Formula Fields	

Opportunity Transition QBs

Transition to Finalizing Stage (Won Scenarios)

Fields.	**These conditions must be fulfilled for the automation to move the opportunity to the finalizing stage.**		
Win Type.	**Quote Signed.**	**Auto-Renew.**	**PO Received.**
Current Stage	Quote Follow-Up	Quote Follow-Up	Quote Follow-Up
Primary Quote	Quote Link	Quote Link	Quote Link
Primary Quote. Status	Signed	Out for Signature	Out for Signature
Primary Quote.NetSuite ID	Set	Set	
Primary Quote.O2C Ticket Link	Set	Set	
Has Auto Renewal Clause	Set	Yes	

Previous Quote with AR Clause		Doc URL IF “Has Auto Renewal Clause” = Yes	
Signed Quote	Doc URL		
Purchase Order	Required | Not Attached	Empty	Attached
Purchase Order Link	Doc URL IF “Purchase Order”=Required		Doc URL
Ready for Finalizing	True	True	True

Requirements to Raise Invoice Request Ticket

Fields.	**These conditions must be fulfilled for the automation to raise Invoice-Ticket for each win type:**		
Win Type	**Quote Signed**	**Auto-Renew**	**PO Received**
Stage	Finalizing	Finalizing	Finalizing
Purchase Order	(Attached | Not Required)		
Purchase Order Link	Doc URL IF “Purchase Order”=Attached		
Manual O2C	False	False	False
Ready for Invoice	True	True	True

Note: Automation doesn’t generate an Invoice Request Ticket for opportunities with Win Type Self-Serve, Co-Term, Other, or Not Set.

Transition to Finalizing Stage (LOST Scenario)

Fields.	**These conditions must be fulfilled to move an Opportunity to the Finalizing-Lost state**	
	LOST	**Won’t Process**
Loss Reason	Set and NOT IN (Data Quality, Duplicate, Handled by the BU,)	Duplicate, Prime, Data Quality, or Handled by the BU
Has Auto-Renewal Clause	True | False	

Cancellation Notice	Set IF “Has Auto-Renewal Clause” = True	
Win Type	Empty	Empty

Close Opportunity

Fields.	These conditions must be fulfilled for the automation to Close Opportunity.			
Stage	**Close-Win**	**Close-Lose**	**Won't Process**	**Co-Term (Win Type = Co-Term)**
Stage	Finalizing	Any	Any	Any
Type	Renewal	Renewal	Renewal	Renewal
Final O2C Record	Ticket URL required when is a NS product and win type is not Self-Serve	Set Only when NS ID is empty and NetSuite Product=true		
Win Type	Quote Signed | AR | PO Only			
Purchase Order	Attached | Not Required			
Purchase Order Link	Document URL IF “Purchase Order” = Attached			
Primary Quote	Not Null			
Primary Quote Status	Accepted, Generated or Signed or (‘Out for Signature’ when win type is Auto-Renew or Po Received)			
Loss Reason		Other than the	Data Quality |	

		reasons for Won't Process	Duplicate | Finance Action | Handled by the BU | Handled by Central Renewal | Prime	
Renewal Subscription Status	Activated only for Netsuite Products	Terminated - Churn | Draft | Empty (For non-NetSuite Products)	Terminated - Invalid Subscription | Terminated - Finance Action	Terminated - *
Provisioning Ticket Required	True | False	True | False		
Provisioning Ticket	'Ticket URL' IF "Provisioning Ticket Required" = True	'Ticket URL' IF "Provisioning Ticket Required" = True		
Customer Business Value	Set	Set		
Customer Moving Off Product	Set	Set		
Customer Sentiment Details	Set	Set		
AR Scenario	Not Empty IF WIN TYPE=AR			
Has Auto-Renewal Clause	Yes | No			
Previous Quote with AR Clause	URL IF "Has Auto-Renewal Clause" = Yes			
Customer Termination Notice Period	Number URL IF "Has Auto-Renewal Clause" = Yes			
Co Term To				Set
Ready to Close	True	True	Should not matter	True
All Tasks Completed	True	True	Any	True

Ref:
Opp cannot be closed unless Ready For Close flag is true, here is the formula for the field

((TEXT(Renewal_Subscription_Status__c) = 'Terminated - Churn' || TEXT(Renewal_Subscription_Status__c) = 'Activated') && All_Tasks_Completed__c && (!Provisioning_Ticket_Required__c || !ISBLANK(Provisioning_Ticket__c)) && !ISBLANK(TEXT(Customer_Business_Value__c)) && !ISBLANK(TEXT(Customer_Moving_Off_Product__c)) && !ISBLANK(Customer_Sentiment_Details__c) && (!ISPICKVAL(Win_Type__c,'Auto-Renew') || !ISBLANK(TEXT(AR_Scenario__c)))) ||
TEXT(Renewal_Subscription_Status__c) = 'Terminated - Invalid Subscription' || TEXT(Renewal_Subscription_Status__c) = 'Terminated - Finance Action'

VR that prevents closing (won/lost) an opp:

OR(Text(StageName) == 'Closed Won',Text(StageName) == 'Closed Lost') && !Ready_To_Close__c && Product_Configuration__r.NetSuite_Product__c && !ISPICKVAL(Type,'CSM') && NOT($Setup.Quality_Bars_Configuration__c.Disabled_Opportunity_QBs__c)

Re-Open Opportunity (Proposal)

Fields.	**The following values should be set/unset when an Opportunity is Re-Opened to the respective Stage.**	
Win Type	**Proposal**	**Finalizing**
All Tasks Completed	False	False
Ready to Close	False	False
Ready for Invoice	False	False
Win Type	empty	empty
Loss Reason	empty	empty
Final O2C Record	empty	empty
Provisioning Ticket	empty	empty

Note: The only reason to send an Opportunity to the proposal is when a change is needed in the Quote after customer negotiation.

Renewal Automation Flows

Sync Draft Renewal Subscription from NetSuite to Salesforce Opportunity [Link]

ITDs
1. **NetSuite Subscriptions (creation/updates) will be synced to NetSuite via a daily scheduled job.** a. The reasoning for choosing a scheduled job instead of an instant webhook trigger is to not sync the incomplete changes (work-in-progress by O2C on subscriptions) to Salesforce. 2. **Salesforce Objects that are part of the integration will store the NS record ID of their NetSuite counterpart instead of NetSuite keeping references to Salesforce.** a. NetSuite must be able to operate without Salesforce and not incur any modifications on behalf of Sales (per the integration strategy). b. All SF objects involved in the integration will have a NetSuite ID field (API name NetSuite_ID__c) added to the Org. c. The field must have SF "External ID" and "Unique" properties set so SF can index and enforce uniqueness (to ensure consistency between systems). d. Integration flows will use the NetSuite ID field to locate respective objects in NetSuite and Salesforce. e. Referential integrity is maintained by: i. We will not allow the deletion of activated Salesforce Objects (i.e., after the respective NS record has been created). ii. Activated NetSuite Subscriptions cannot be deleted and can only be deactivated and closed. 3. **We will not use any of Salesforce's internal automation for creating renewal opportunities and instead create the Opp from scratch to mirror NetSuite renewal Subscription.** a. While it would be more convenient to rely on SF to create and populate the renewal Opportunity based on the renewed Contract, Salesforce will use a set of renewal rules that must be configured to MATCH the rules used by NetSuite. b. These rules are spread across a number of objects, and it's difficult to ensure a 100% match between NetSuite and Salesforce.
High-Level Execution Flow:
1. A NetSuite Draft Renewal Subscription is created/updated 2. A daily TrayIO scheduled job is run a. It queries all the supported classes from Salesforce Product Configurations to filter the core-renewal subscriptions to be imported b. It queries all the applicable subscriptions created/updated after the last job execution time. c. It logs the job execution start time in the log file d. Loop through the queried subscriptions. e. Fetch the corresponding Salesforce Opportunities if exist f. Invoke the sub-flow to sync the draft renewal subscription to Salesforce Opportunity 3. Sync is skipped if the subscription is detected to be generated by an E2E (E2E's are only run in Sandbox, but this is an extra protection) 4. If Renewal Logs are enabled, an entry is created to the GSheet log file. 5. Parent subscription data is fetched to query the SF Opportunity by Parent ID and NS IS of the

subscription.

6. Subscription Plan data is fetched from the subscription
    a. If the subscription is created on a legacy plan, the opportunity products are not synced
        i. A message is added to the opportunity description to manually add the products to the opportunity
    b. If the subscription is created on the standardized non-legacy subscription plan, then the subscription line items are iterated along with price intervals to be synced to Salesforce Opportunity products.
7. If Opportunity is found
    a. And if the Opportunity is not closed, then the current * fields and NS ID is updated.
    b. If Opportunity is in Pending Stage
        i. All the fields, including current * fields, are updated
        ii. Opportunity products are updated if the Subscription is not legacy
    c. If Opportunity is closed
        i. Nothing is updated.
8. If Opportunity is not found.
    a. The opportunity is created in Salesforce
9. The current * fields on the opportunity are synced from the NetSuite Parent Subscription data.
10. An update message is prepended to the Opportunity description.

Convert an Accepted Salesforce Primary Quote to NetSuite Quote and Subscription [Link]

ITDs

1. **A Salesforce Opportunity Account/Partner Account and its contacts should only be synced to NetSuite when an SF quote is accepted.**
    a. There is an “Addressed validated” checkbox on the opportunity that is mandatory to check before sending an opportunity to the Final Quote stage.
    b. By checking that ISR confirms that the billing/shipping addresses on the SF Account are correct and are safe to be synced to NetSuite.
    c. This is primarily not to sync invalid/corrupt data to NetSuite.
2. **Automation deletes the old draft renewal and generates a new draft renewal subscription using the SF Quote product, price, and billing parameters.**
    a. Automation will then use the draft renewal subscription as input to generate NS Quote.
    b. The NS Quote will be linked to the renewal subscription as a transaction record.
3. **For the SF Quotes with Partners, it’s mandatory to populate the Reseller Agreement DS on the opportunity for the service provider set on the SF Quote.**
    a. For resellers, we need to generate the quotes without full T&Cs, assuming a reseller agreement is already signed with the resellers/distributors.
4. **The Quote Expiry date is taken from SF Quote and set on the NS Quote.**
5. **The Payment Term and Billing Frequency are taken from SF Quote, and the same values are set on the NS quote.**
    a. All the business logic to calculate quote expiry, Payment Term, and billing frequency should reside in Salesforce.
6. **Automation will only generate the NSQuote and O2C Quote Request tickets when Opportunity.Manual O2C flag is false.**
    a. There are some scenarios not handled by automation, e.g., more than one bundle product in the primary quote, co-term opportunities, non-standard term, etc., that cause the manual O2C flag to be set to true.

b. Automation will also not trigger for Non-NetSuite products determined through Product Configuration.NetSuite Product flag. **7. The reseller agreement should be taken from Salesforce Opportunity and attached as a PDF file to the Quote Request Jira ticket (not as a Salesforce link)** a. O2C does not have access to Salesforce and can not visit and view the document in Salesforce.
High-Level Execution Flow:
1. A SF primary quote is accepted by the Sales VP and the opportunity is in the Final Quote stage 2. Automation will trigger to generate NS Quote and O2C Quote Request Jira ticket when: a. Primary Quote status is accepted b. Opportunity is in the Final Quote stage c. Opportunity.Manual O2C flag is false d. Opportunity.Product Configuration.NetSuite Product flag is True 3. The Opportunity Account/Partner and its contacts will be synced to NetSuite before generating the quote. 4. Automation will take all the quote parameters, including products, quantities, billing frequency, term, final amount, payment terms, account/partner, expiry date, etc., and use them to generate a new draft renewal subscription and send them to TSI API for Draft Renewal Subscription and NS quote generation. a. Previous draft renewal subscription will be deleted, and a new one will be created exactly reflecting the quote's prices, items, and billing parameters. b. The draft renewal subscription is used as input to generate NS Quote and agreement. c. The logic to use the correct quote form and T&Cs is encoded in NetSuite. 5. Upon successful generation of NS Quote, a Quote Request Jira ticket will be created in O2C Jira with NS Quote details and instructions on how to QC it. 6. The O2C Jira Ticket Link and NetSuite ID of the NS Quote will be populated to SF primary quote. 7. The SF Quote status will be moved to Generated.

A NetSuite Quote (Agreement) is Signed or sent Out for Signature, Update the Salesforce Quote status [Link]

ITDs
1. When the NetSuite Agreement status is changed to 'Out for Signature' or `Signed', the automation will reflect that status to the Salesforce quote matched by the NetSuite ID field. a. The transition of the Opportunity to Quote follow-up or Finalizing will be done through Salesforce automation. **2. The counter signator signatures for the service provider will be prefilled to the quote as an image to expedite the sales process as soon as the customer signs a quote.**
High-Level Execution Flow:

Raise an Invoice Request Ticket for a Salesforce Opportunity [Link]

ITDs

1. **Once the opportunity is moved to the Finalizing stage with the correct win type (Auto-Renew or Quote Signed), automation waits for all the QBs to be met to raise the Invoice Request O2C Jira ticket**.
    a. Invoice request ticket is raised when the “Ready for Invoice” flag is set to true.
    b. QB’s to mark the “Ready for Invoice” is documented here.
2. **A new subscription is created.**
    a. If a draft subscription already exists for this opportunity, this subscription is deleted.
    b. The new subscription is created based on the primary quote, and the subscription is soft-linked to the quote. Also, if a subscription was deleted and had soft-linked quotes, the quotes will be soft-linked to the new subscription.
3. **An Invoice Request ticket should only be raised when the Manual O2C flag is false.**
    a. In the case of manual O2C=true, automation has not generated the latest draft renewal corresponding to the primary quote and can not be used to issue an invoice.
    b. That’s why O2C needs to manually delete the old daft renewal subscription and generate a new one to activate and invoice.

High-Level Execution Flow:

A Subscription in NetSuite is Activated/Terminated, Update the Salesforce Opportunity status [Link]

NetSuite Subscription is Activated (Customer Activation \| AR \| Finance Action) [Link]

NetSuite Subscription is Terminated (Customer Cancellation \| Finance Action) [Link]

ITDs

1. **Activated/Terminated subscriptions in NetSuite are processed and synced to Salesforce via a daily scheduled job.**
    a. The reason for using batch jobs is to prevent work-in-progress changes from being synced to Salesforce and causing inconsistent Salesforce data.
2. **When a subscription is terminated in NetSuite and a replacement subscription has been created, automation updates the NetSuite ID on the open opportunity.**
    a. This process is done prior the processing of any change orders.
    b. The termination order on the replaced subscription will be processed but not synced to the opportunity.
    c. Any change order on the replacement subscription will be processed against the updated opportunity.
3. **When a subscription is terminated in NetSuite, automation sets the ‘Renewal**

Subscription Status' field on the Salesforce opportunity a. The value is selected according to the "Change Reason" field on the subscription change order (Terminated - Churn, Terminated - Finance Action, Terminated - Invalid Subscription). **4. When a subscription is activated in NetSuite, automation sets the 'Renewal Subscription Status' field to "Activated" on the Salesforce opportunity.** **5. How is the opportunity searched in Salesforce?** a. The search is first performed by NS ID; if the opportunity is not found by NS ID, then a search is conducted by Parent Subscription ID and empty NS ID. b. If an Opportunity is not found, and the product integration record for this product exists, then a bug ticket is automatically created. **6. The logic to close-win or close-lose an opportunity is encapsulated in Salesforce automation.** a. Salesforce automation uses a number of QBs to determine whether an opportunity is ready to close or not. b. All the opportunities close QBs are documented here. c. Once all the QBs are met, the "Ready to Close" flag on SF Opportunity is set to true, and automation closes-win or close-lose the opportunity.
High-Level Execution Flow:

Actions to be taken depending upon the Change Reason for a Subscription Change Orders

Order Type	Change Reason	Action to be taken
Termination	Finance Action	
Termination	Invalid Subscription	
Termination	Churn	
Activation	Finaction Action	
Activation	Customer Activated	
Activation	Auto Renewed	

A Draft Renewal Subscription in NetSuite is Deleted, Update the Salesforce Opportunity

ITDs
1. **When a draft renewal subscription is deleted in NetSuite, an event is triggered that sets the NetSuite ID on the corresponding Opportunity to empty.** a. The NS ID on opportunity is emptied because the subscription no longer exists in NetSuite, and the opportunity should not point to the non-existing entity. b. In subsequent steps, when a new draft renewal subscription is created for the same parent, then automation will search the opportunity by Parent ID and empty NS ID combination.
High-Level Execution Flow:

When a NetSuite Subscription is Activated/Terminated, Raise Zendesk Provisioning/De-provisioning ticket [Link]

ITDs
1. **The decision to raise a provisioning or de-provisioning ticket completely relies on NetSuite data and does not consult Salesforce in decision-making.** a. The Product Integration Records in NetSuite contain the configuration for each product that needs a Provisioning/Deprovisioning or both tickets to activate/terminated licenses/subscriptions. 2. **Once the Zendesk ticket is raised, if it's a core-renewal product, the corresponding Salesforce opportunity is searched, and the Provisioning Ticket URL field on the opportunity is populated with the link.** 3. **If Automation fails to raise a Zendesk ticket where it was necessary, a task for the ISR is created on the opportunity for him to manually create a provisioning/de-provisioning ticket.**
High-Level Execution Flow:

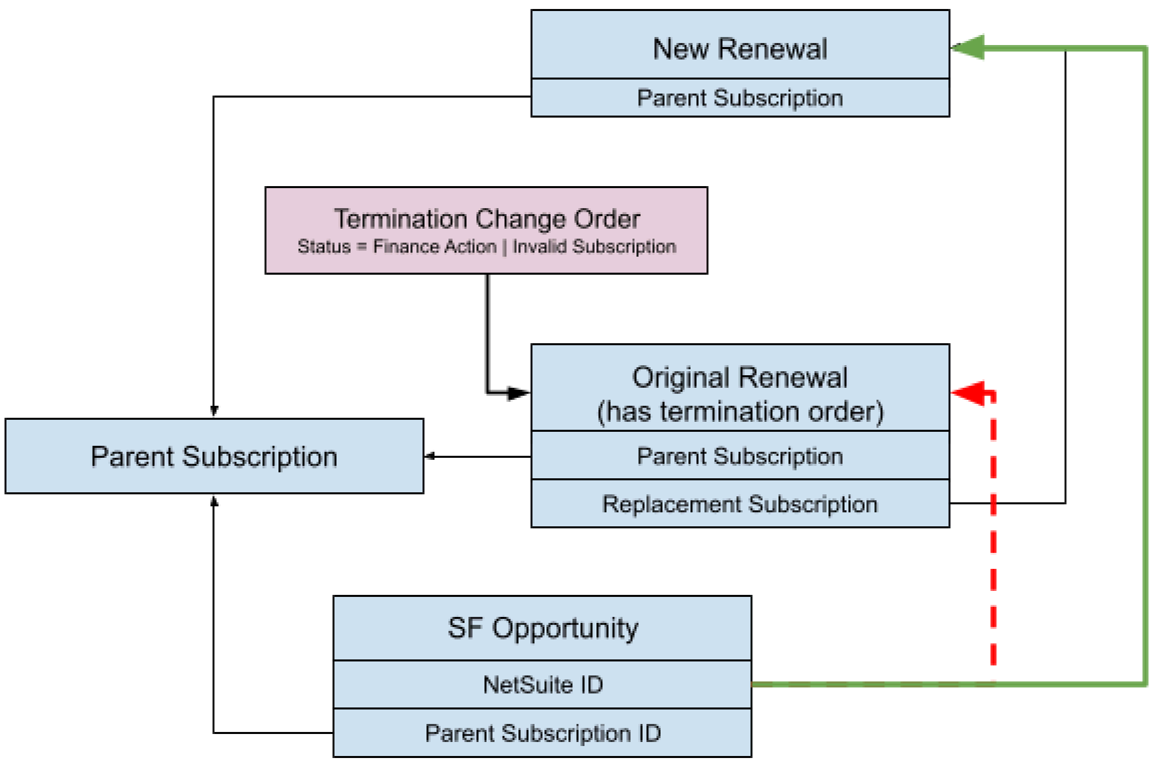
When a NetSuite Subscription is terminated and is replaced, fix NetSuite ID on Salesforce Opportunity [Link]

ITDs
1. **When a draft renewal subscription is terminated and undergoes replacement, the**

associated NetSuite ID on the synchronized opportunity is adjusted to correspond with the replacement subscription.

a. A subscription is properly replaced when a replacement subscription is designated for the same parent subscription.
b. The termination order should be attributed to the reasons "Invalid Subscription" or "Finance Action".
c. The relevant opportunity is identified matching the NetSuite ID and the replaced subscription ID, and must be in an open stage.

Schema



High-Level Execution Flow:

This subflow is invoked for a termination order if a replacement subscription has been assigned to the terminated subscription.

1. If any of the following conditions are met, skip the process without triggering an error:
    a. The termination order's reason is not "Invalid Subscription" or "Finance Action".
    b. The replaced subscription and the replacement subscription belong to distinct parent subscriptions.
    c. No open opportunity is synchronized with the replaced subscription.
2. Update the open opportunity to reference the replacement subscription in place of the replaced subscription.

When a Salesforce Opportunity is Closed [SF Flow]

ITDs
1. **Salesforce automation will automatically close-win or close-lose the opportunity.** a. All the QBs for ready to close are documented here
High-Level Execution Flow:

SF Opportunity is recalled back from the Finalizing Stage [Link]

ITDs
1. **When an opportunity is recalled from the Finalizing stage to the previous stage, the O2C Invoice Request ticket is canceled if it's in the Input QC stage.** a. If the Invoice Request ticket is past the Input QC stage, then a comment on the ticket is added to not issue an invoice and wait for a new invoice request ticket. 2. **When an opportunity is recalled from the Final Quote stage, the Out for Signature agreement is canceled.** a. A quote request Jira ticket is also canceled if it's in the Input QC stage.
High-Level Execution Flow:

NetSuite Quote Agreement is canceled [Link]

ITDs
1. **When a NetSuite Quote Agreement is canceled, the automation triggers and updates the corresponding Salesforce Quote status to cancel.** a. If the canceled Salesforce quote is a primary quote, then the opportunity is moved to the Proposal stage by a Salesforce automation.
High-Level Execution Flow:

SF Opportunity Stage is Changed [Link]

ITDs

High-Level Execution Flow:

An Account in Salesforce is Created/Updated, Create Update NetSuite Customer [Link]

ITDs
1. **Accounts created/updated in Salesforce are not synced to NetSuite instantly.** a. Accounts are only synced to Salesforce when an Opportunity is moved to the Final Quote stage and the primary quote is accepted. b. All the contacts related to the account are also synced to NetSuite at this stage.
High-Level Execution Flow:

A Customer in NetSuite is Created/Updated, Create Update Salesforce Contact [Link]

ITDs
1. **A customer update in NetSuite is instantly synced to Salesforce Account, along with all of its related contacts.**
High-Level Execution Flow:

A Contact in Salesforce is Created/Updated [Link]

ITDs
1. **A contact in Salesforce should only be synced to NetSuite when the contact is acting as a primary contact to an accepted Salesforce quote.**
High-Level Execution Flow:

A Contact in NetSuite is Created/Updated [Link]

ITDs
1. **When a contact in NetSuite is updated, its related customer and all contacts are synced to Salesforce.**
High-Level Execution Flow:

Product Catalog Sync

On NetSuite Subscription Plan Created/Updated, Create/Update SF Product [Link]

ITDs
1. **A daily scheduled job syncs the updated/created NetSuite Subscription Plans to Salesforce that were updated after the last job execution.** 2. **Every update to Subscription Plan will generate a new version of the Salesforce product and deactivate the old version in Salesforce.** a. The Salesforce product bundles contain the addons as sub-bundles (bundles with bundle) b. Use the SKU field on the SF product to store the versioned code, i.e., -XXX c. Automation deactivates the previous version of the bundle after successfully creating the new version d. The related products in the SF product bundles are ignored by automation.
High-Level Execution Flow:

Legal Review

Project Link (sandbox)

Create Contract Legal Review [Link]

ITDs

• Contract Review will be initiated from Salesforce, and we will use Salesforce as a store of results and for automating subsequent actions.
  ○ Contract Review is a part of the Renewal process which is managed in Salesforce. The data required to automate contract termination action has to be in Salesforce.
  ○ Contract Review's purpose is to identify contracts that don't align with our business rules and to correct them. After the renewal process, the newly signed contract will conform to our business rules, so there is no need to store any contract parameters in NetSuite.
• We will start with high-value opportunities as identified by the "High Value" field on the opportunity record in Salesforce.
  ○ Currently, it identifies opportunities > $100K ARR, but it may change in the future.
• We will review the contracts for 6 months (using "months" to align on the month boundary) before renewal to give ample time to act on the termination if needed.
• Legal Contract Review work unit will aggregate multiple contracts together instead of sending them for review one by one.
  ○ We will aggregate 1 month-worth of contracts into a single ticket on a month boundary 6 months into the future.
  ○ I.e., on Jan 1, we will collect all high-value contracts with renewal dates July 1 - July 31 and send them for review.

High-Level Execution Flow:

Flows related with Contract Legal Review are in the Contract Review Project (sandbox):
• **Create Contract Legal Review** (sandbox) - This is the main flow that is executed for each opportunity we want to add data in the spreadsheet for Legal Review. This flow:
  ○ Gets all data from netsuite going through the subscription chain.
  ○ Creates a folder for this opportunity inside the Contract Review folder.
  ○ Gets from netsuite all files attached to the contractual documents for the subscriptions in the subscription chain and uploads all files to the opportunity folder.
  ○ If it's a subscription sold through reseller also uploads to the opportunity folder the reseller agreement if it's attached in salesforce.
  ○ Inserts the data needed for Legal Review in a new column in the Contract Review spreadsheet.
• **Scheduled Create Contract Legal Review** (only in production) - This flow is schedule to execute in the 1st of each month and:
  ○ Gets all high value opportunities that are due to renew in 7 months.
  ○ Creates a folder named Contract Review for MM/YYYY (Created at YYYY-MM-DD)
  ○ Copies the spreadsheet Contract Review Template and places it inside the new folder.
  ○ Loops all high value opportunities calling the Create Contract Legal Review flow to process and get the documents and data for each opportunity.
• **Trigger Manually Create Contract Legal Review** (sandbox) - This flow does the same thing that the schedule flow, but it can be triggered manually when desired.
  ○ Change the number of months in this step to execute for the desired month.

Sync Contract Review NS -> SF [Link]

ITDs
• The data analyzed by Legal will be inserted in netsuite. • We need to sync that data from NS to SF.
High-Level Execution Flow:
• Netsuite Contract Review is synced to salesforce when created/updated/deleted. (sandbox link)

Outreach

Opportunity is moved to Outreach stage [Link]

- Sub Automation: Start Outreach Sequence for an Opportunity [Link]

ITDs
High-Level Execution Flow:

TSI AI Automations [Link]

Get Call Transcript [TrayIO Link]

Salesforce Flows

- On Call Transcript Ready, Generate Call Summary [SF Flow]
- On Call Transcript Ready, Generate Followup Email [SF Flow]
- On Call Transcript Ready, Extract Platinum Offer Feedback [SF Flow]

TSI AI API [TrayIO Link]

OpenAI Prompt Execution [TrayIO Link]