# NetSuite Data Management

# Data Management Overview:

NetSuite Data Management process is overview of data migration best practices to be followed for smooth and error free migrations from Legacy system to NetSuite ERP.

# Subsidiary Management:

* Subsidiary management is very first and important steps in data migration, we need to understand the current business entities including the future expansions and tax reporting
* Prior to implementation, defining your company structure and confirming which entity rolls up into another, and the reporting output you require, is vitally important. Once these are defined in NetSuite it is very hard, if not impossible, to change.

There are two key elements when it comes to the subsidiary structure:

1. Consolidated Reporting
2. Intercompany transactions

## **Consolidated Reporting**

When setting up your structure NetSuite will report accordingly, getting this structure right will allow for greater clarity on reporting and the figures presented in NetSuite.

## **Intercompany transactions**

A multi-subsidiary NetSuite account allows for intercompany trading, and with this there are three things to consider when establishing your structure.

1. **Elimination –** Whenever you set up your subsidiary structure in NetSuite there is a requirement for an “Elimination subsidiary” set up for each level of the subsidiary structure. This means that when an intercompany transaction occurs, after running period-end these transactions are posted to the elimination subsidiary, opposed to presenting in the consolidated reporting. This effectively zeros out the balance on each of the subsidiary ledgers, meaning you aren’t accounting incorrectly on your financial statements for a consolidated view.
   1. NetSuite does not auto create Elimination Subsidiaries
   2. Must have same currency as Parent
   3. Only Journal Entries post to elimination subsidiary
2. Each individual entity requires intercompany customers and vendors that represent the other subsidiaries.
3. **Intercompany Purchase Order/Sales Orders –** when creating a Purchase Order using an intercompany vendor this allows for automatic creation of a Sales Order in the subsidiary you are wanting to trade with.

# Classifications:

Classifications are powerful tools that can be used to identify and categories records in your NetSuite Account.

* **Department:**

Used to designate transactions and employees as part of an internal team or cost center.

* **Class**

Often used in conjunction with departments, classes provide wider segments within your business.

* **Location**

Like with departments and classes, Locations allow you to filter and track data. To do this, you must create a record for each location you would like to track. Then, you can then associate employees and transactions to a location.

* **Custom Segment**

Another classification functionality within NetSuite is the Custom Segment Feature, allowing you to create additional custom fields like class and department. These custom segments can be assigned multiple possible values, and these segments can be added to specific record types to suit your bespoke needs.

Applications of Segmentations:

## Used in Reporting

## Restricting Access by Classifications

## Segmentation of Data at a Transactional Level

# Chart of Accounts:

The Chart of Accounts is always the first list to get uploaded in NetSuite and then you can upload employees, Customer or Vendors.

This to keep in mind as best practice for COA migrations:

* Identity all the parent and child accounts
* Identify the correct account type for each GL
* Identify the appropriate currency for each GL account
* Identify the subsidiary for each GL account
* Identify whether GL account is summary account
* To upload COA, two files are required (Parent file and Children file)

Standard Numbering for Chart of Accounts:

NetSuite suggests the below Numbering format for Chart of Accounts, but user can provide his/her own numbering format based on the business requirements:

1xxx - Asset accounts

2xxx - Liability accounts

3xxx - Equity accounts

4xxx - Income accounts

5xxx - Cost of Goods Sold (COGS) accounts

6xxx - Expense accounts

8xxx - Other Income accounts

# Vendor Master Management:

* A vendor is a company or person from whom you purchase goods and services. Vendor records track information about your vendors and enable you to view past transactions and communications with them.
* In the **Vendor ID** field, enter your vendor's name the way it should appear in all lists.
  + If you leave this field blank, this field fills with the name you enter in the **Company Name** field.
  + If you use Auto-Generated Numbering, this field fills with the vendor number or code.
* If you use NetSuite One World, in the **Primary Subsidiary** field, select the primary subsidiary to assign to this vendor. You cannot enter transactions for this vendor unless a primary subsidiary is assigned. The default primary currency for the vendor is the base currency of the primary subsidiary.
* If you use NetSuite One World, the **Subsidiaries** subtab enables you to assign the secondary subsidiaries that can share this vendor. After you save the vendor record, you can see vendor open balances at subsidiary level.
* You can assign a primary subsidiary and an unlimited number of secondary subsidiaries to a vendor record.

**Points to be remembered while using multi-currency feature:**

* Set primary and additional currencies on the customer or vendor record
* Enter Sales or Purchase transactions in multiple currencies – adding as many as required in the sublist
* Define Primary Currency as the default for sales or purchase transaction information and credit limit.
* We can change the primary currency on a Customer or Vendor record, resetting the default for transactions, credit limits and aggregate balance
* If primary currency is changed, we must re-enter the credit limit
* Cannot remove any currencies that have transactions against them
* If the consolidated payments feature is enabled, then customer and sub-customer can have different primary currencies.
* Primary Subsidiary once fixed on the customer or vendor record, it cannot be changed rather additional subsidiaries can be added.

# Customer Master Management

* The Multi-Subsidiary Customer feature permits you to share customer and sub-customer records with multiple subsidiaries, and then select those subsidiaries on core transactions.
* The Multi-Subsidiary Customer feature also enables you to view the following:
  + Customer balance information on the customer record
  + Secondary subsidiary information on core reports
  + Customer Statement by currency and by subsidiary
* The Multi-Subsidiary Customer feature enables you to create a multi-subsidiary vendor record from a multi-subsidiary customer record. This customer vendor relationship produces a single entity, which is useful when you want to use one entity to represent both a customer and a vendor on core transactions.
* **Following are limitations in the Multi-Subsidiary Customer feature:**
* Incompatible Features:
  + Consolidated Payments
  + Mini EU One Stop Shop (MOSS)
* Limited Functionality:
  + Customer specific budgets can be created only for the primary subsidiary of the customer
  + Communication subtab on the customer record
    - Messages, Activities, Files, and User Notes added to the customer record are available only for the primary subsidiary.
* Technical Limitations:
  + To ensure that dynamic mode scripts and workflows perform as expected, the selected entity or customer (depending on the transaction) determines the default subsidiary.

# Accounts Receivables:

* **Best Practice for Data Migration of Open Deferred Revenue Schedules:**

There are three approaches that we can take to bring in the orders that have outstanding deferred revenue balances. In the case of high transaction volumes, the CSV import tool can be leveraged to create the needed records, which in turn can help us speed up this important data migration process. For term licenses and support revenue that need to be recognized in a straight-line fashion, a Straight-Line Rev Rec template will need to be pre-configured in NetSuite before any import activity can commence. If this revenue needs to recognize based on a percent completion of a project, a Variable Revenue Recognition Schedule will need to be configured first.

* **Straight-line Revenue Recognition Schedule**

Imagine if one of your on-going deals is a 12-month support contract that was originally sold on February 1st, 2015 for $24,000. This contract will have an expiration date of February 31st, 2016. After 6 months, let’s assume that half of it has been recognized and the remaining $12,000 balance is left as deferred revenue waiting to be amortized. If your team is ready to transition operations onto NetSuite on August 1st, 2015, this outstanding amount will need to be imported into NetSuite. This can be achieved by using either the Cash Sale or Journal Entry record types.

**Approach 1: Leveraging the Cash Sale Record Type**

Once the Cash Sale has been successfully imported into NetSuite using the CSV Import tool or created manually via the UI, the Cash Sale will have the following attributes set:

| **ITEM** | **NAME** | **AMOUNT** | **REV REC TEMPLATE** | **START DATE** | **END DATE** |
| --- | --- | --- | --- | --- | --- |
| [Item Name] | [Customer Name] | 12,000 | Straight-Line | 8/1/2015 | 2/31/2016 |

To accomplish this, a journal entry needs to be created. The final step that is required in this approach is to reverse the GL impact of this Invoice. It is recommended to use a clearing bank account to avoid posting to the real bank account and bank reconciliation.

**Approach 2: Leveraging the Journal Entries Record Type**

Prior to creating the Journal Entry, a configuration step is required to associate the appropriate Deferred Revenue account on the Revenue account in the Chart of Accounts. Once the Journal Entry has been successfully imported into NetSuite using the CSV Import tool or created manually via the UI.

* **Variable Revenue Recognition Schedules – Based On % Completion of Projects**

Imagine if one of your on-going support projects was originally sold for $24,000. Now, if the consultants on this project have already used up 25% of the allotted hours, this would indicate that $6,000 of the service revenue has already been recognized. Based on this detail, $18,000 is left over as deferred revenue balance. (e.g. If your team is ready to transition operations onto NetSuite on August 1st, 2015, this outstanding amount will need to be imported into NetSuite and this can be achieved by using the Project record.)

**Approach 3: Leveraging the Project Record Type**

The Project will be created in NetSuite with % completion for rev rec defined. To achieve this, the Rev Rec Override Percent Complete field on the Project must be set to 25%. The Project record can either be imported using the CSV Import tool or created manually via the UI.

Once the import of the Project is completed successfully, an additional Journal Entry will need to be created to account for the remaining $18,000 that has yet been recognized. Prior to creating the Journal Entry, a configuration step is required to associate the appropriate Deferred Revenue account on the Revenue account in the Chart of Accounts.

The final step to complete this approach is to run the Revenue Recognition engine to create the two Revenue Recognition Journal Entries from the Revenue Recognition Schedule.

# Account Payables balances:

All open and unpaid bills your company owes in your legacy system prior to your changeover to Accounting Seed will need to be imported as Payables so that future vendor payments can be properly matched and paid against the respective vendor invoice due.

* Migrate the open vendor bills with generic item as “Opening Balance” and mentioned the details in the memo field to track while create the Journal entry for reversing the balances
* Once all open bills have been imported, journal entries must be created to reverse the net GL impact of those bills.
* A saved search is best to aggregate the GL impact of all bills created.  The journal entry created will be the **reverse**of the GL impact of the vendor bills imported.
* Open POs that have no related transactions should be imported with their line item details.  POs can be imported into NetSuite if there are many of them still open at the time of the cutover.
* For non-inventory purchases that have not been billed yet, the easiest method is to just enter the vendor bills as you receive them.  The vendor bill will immediately enter the payable due and post the amount to the correct expense account.

# Fixed Asset Migration: Mid-Life Assets

The NetSuite Fixed Assets Management SuiteApp simplifies the management of your fixed assets by automating these various functions. Whether you are a new customer looking to implement this feature, or an existing one planning to enable it, you will need to import your existing mid-life assets into NetSuite as part of the initial setup.

First, we recommend using the CSV import tool to create your existing fixed asset records by loading them in as the ‘FAM – Asset’ record type. At a minimum, the following mapping can be used to complete this task:

After you have imported the Mid-Life Assets, there are two additional records that need to be created in order to track Fixed Assets in NetSuite.

* **The Depreciation History – Acquisition record** which creates the original transaction amount and the beginning book value
* **The Depreciation History – Depreciation record** which creates the value of depreciation.

The first step to creating these two records is to export the existing Asset records from NetSuite. The best way to do this is to create a new FAM - Asset search by navigating to ‘Fixed Assets -> Searches -> Asset List’

When the search results load, click the ‘Export – CSV’ button to extract the results in CSV format to Excel. You can now use this resulting file to create the Acquisition and Depreciation records.

Using NetSuite’s Import Assistant tool, select Custom Records as the Import Type, FAM – Depreciation History as the Record Type, and adding the file created earlier in CSV format. Moving to the next step through the Import Assistant, set Data Handling to “ADD” and make sure the Custom Form is set to “Standard FAM – Depreciation History” under Advanced Options.

The next step in the import is the mapping.

This will create the **Depreciation History – Acquisition record**.

Next, to create the **Depreciation History – Depreciation record**, proceed through the Import Assistant a second time.

This will create the **Depreciation History – Depreciation record**.

Once the records are created you will be able to navigate to the Fixed Assets menu -> Lists -> Assets. By clicking on the View link for each line you will be able to see the Asset details. You will also see the Depreciation History under the Depreciation History subtab.

It is a good practice to tie out your assets using the Asset Summary Report through the Fixed Assets menu. This report includes information such as cost, depreciation, and net book value. You can restrict this report for a specific period and for specific asset types. You can make sure the amounts on the report match the amounts on your balance sheet. This is not a required step but is recommended for sound accounting.

# Trial Balances Migration:

1. **Use CSV Journal Entry Import:**A journal entry is needed for each trial balance period.  Use a CSV “Transactions” type with a “Journal Entry” record type.  A single file approach with all our data is generally easier than a multi-file linked approach.
2. **Import Balance “Debit”**:  Target a single column of balance data.  If the balance is positive, it will be a debit; if the balance is negative, it will be a credit.
3. **Avoid Retained Earnings Account:**  NetSuite automatically produces the retained earnings account as a derivative of income statement accounts in previous periods.  To minimize trying to explain retained earnings balances, I recommend any actual journal entries to Retained Earnings be to a separate account, such as “Retained Earning – Adjusted”.
4. **Avoid Control Accounts:**  To help explain account balances that are generally derived from subledgers, such as Accounts Payable or Accounts Receivable, I recommend avoiding posting directly to these accounts.  Just like Retained Earnings, use “Accounts Payable – Historical” or other to illustrate that this information came from your Trial Balance import work.
5. **Use Account Internal IDs**:  In general, it is easier to import related data using internal IDs versus names for cross reference information.  While it is more work to setup, it saves time during the import process– the CSV File import tools needs pointers to the related lists and table data and the internal ID is precise; otherwise, you may find that NetSuite throws errors when it can’t quite lookup on name / text values.
6. **Import Multiple Journal Entries:**  In general, it is easier to perform one import versus 12 imports to constitute a year of balances.  The key to make this happen is to organize your data to have a different entry number for each period.   Name each entry period distinctively.
7. **Confirm Import Data Foots to Zero:**  All trial balances structured into a single column of data representing both debits and credits, including data import files containing multiple journal entries, must foot to zero.
8. **Close Periods once Satisfied**: Once your data is imported without error, run financial statements (Balance Sheet and Income Statements) to confirm expectations.  If good, close the periods to prevent changes.