

# A Technical Guide to DCC Implementation in NetSuite

Published August 4, 2025 55 min read

---



## Implementing Dynamic Currency Conversion (DCC) in NetSuite for Global Sales

### Introduction

Expanding globally means dealing with multiple currencies and ensuring customers can transact in their preferred currency. Dynamic Currency Conversion (DCC) is a solution that enhances customer experience by letting international customers pay in their home currency while the business still settles in its own currency. This report provides a comprehensive guide to implementing DCC within

NetSuite, combining NetSuite's robust multi-currency capabilities with customizations and integrations. We cover an overview of DCC and its benefits, a deep dive into NetSuite's native multi-currency management, step-by-step configuration guidance (including use of [SuiteScript](#), [SuiteFlow](#), and [SuiteAnalytics](#)), recommendations for third-party tools/payment processors that support DCC, compliance and accounting considerations, potential challenges, and real-world insights. The goal is to equip NetSuite administrators, business analysts, and IT decision-makers with both conceptual understanding and practical steps for enabling DCC in a global sales environment.

## Overview of Dynamic Currency Conversion (DCC)

Dynamic Currency Conversion is a service that allows a transaction to be processed in the **cardholder's home currency** at the point of sale, instead of the merchant's local currency (Source: [stripe.com](#)). In practice, when an international customer makes a purchase (online or in-store), the payment system detects the customer's card origin and offers a choice: pay in the local (merchant's) currency or in the customer's own currency. If the customer accepts the DCC offer, the charge is converted **in real time** to their currency using a known exchange rate and markup, and the merchant receives payment in their usual currency after conversion (Source: [payproglobal.com](#)) (Source: [payproglobal.com](#)). If the customer declines, the transaction proceeds in the local currency and the customer's bank handles the conversion later (Source: [stripe.com](#)). This process is summarized below.

*Figure: Simplified DCC process integrated with NetSuite – The customer is given a choice to pay in their home currency at checkout. If DCC is accepted, the payment gateway converts the amount (using a predefined exchange rate and fees) and settles with the merchant in its base currency, while NetSuite records the sale and payment in the base currency (and can log the foreign currency amount for reference). If DCC is declined, the sale proceeds in local currency normally.*

**Benefits of DCC in global sales:** DCC can improve the shopping experience for international customers by providing **price transparency and convenience** – customers see the exact cost in a familiar currency, avoiding the need to mentally calculate exchange rates (Source: [adyen.com](#)). This clarity can reduce cart abandonment for e-commerce, as many shoppers prefer sites that display prices in their home currency (Source: [netsuite.com](#)). From the business's perspective, DCC can unlock a *new revenue stream*: merchants often earn a small commission or margin on the currency conversion for each DCC transaction (Source: [stripe.com](#)) (Source: [adyen.com](#)). In other words, the exchange rate markup shared with the payment provider becomes additional income on

international sales. DCC also allows merchants to **accept foreign payments without maintaining foreign currency bank accounts** – the conversion service deposits funds in the merchant’s base currency, simplifying treasury management (Source: [weareplanet.com](https://weareplanet.com)). This makes it easier to support overseas customers **without setting up local currency accounts or entities** in every target country (Source: [weareplanet.com](https://weareplanet.com)). Overall, when implemented correctly, DCC can enhance customer trust (through pricing clarity) and contribute to higher conversion rates and average order values for global shoppers.

However, it’s important to acknowledge that DCC has potential downsides if not handled properly. Because DCC conversions often include a markup and possibly additional fees, customers may **pay a premium** for the convenience (Source: [stripe.com](https://stripe.com))(Source: [netsuite.com](https://netsuite.com)). If the markup is too high or not transparent, DCC can lead to customer dissatisfaction – for example, a customer might feel “surprised” by a worse rate or still incur home-bank fees (Source: [netsuite.com](https://netsuite.com)). Card networks and regulators have [compliance rules](#) requiring that DCC must always be optional and clearly disclose the exchange rate and fees upfront (Source: [adyen.com](https://adyen.com)). As we implement DCC, we must ensure **clear communication and opt-in consent** from customers, offering fair rates and allowing them to easily decline DCC to use their bank’s conversion if they prefer. In the following sections, we will see how NetSuite’s capabilities and integrations can be leveraged to deliver DCC in a way that balances these considerations.

## NetSuite’s Native Multi-Currency Management Capabilities

NetSuite is built with strong [multi-currency functionality](#), which forms the foundation for any DCC implementation in the system. At its core, NetSuite allows companies to **enable multiple currencies** for transactions and automatically handles currency conversion, consolidation, and revaluation in the background. Key native capabilities include:

- **Transacting in 190+ currencies:** NetSuite supports transactions in virtually any currency (over 190 currencies), allowing sales orders, invoices, payments, etc. to be entered in foreign currencies as needed (Source: [netsuite.alphabold.com](https://netsuite.alphabold.com)). Each transaction record carries a *transaction currency* and an exchange rate to the company’s base (or functional) currency. This means a company can invoice customers in their local currencies while still reporting in its home currency. NetSuite’s **multi-currency customers** feature lets you associate multiple currencies with a single customer record (or vendor), enabling a customer to place orders in different currencies (Source: [tipalti.com](https://tipalti.com)).

- **Automated exchange rate management:** NetSuite can automate currency rate updates through its Exchange Rate Integration feature. When enabled, NetSuite will pull daily exchange rates from a chosen provider (e.g. Xignite or other integrated providers) and update the currency rate table around 6:00 AM each day (Source: [docs.oracle.com](https://docs.oracle.com))(Source: [docs.oracle.com](https://docs.oracle.com)). This ensures that *company-wide exchange rates* are current without manual effort, reducing errors in conversion. You can also update rates manually or even use multiple rate types (e.g. spot rate, average, historical) for different purposes (Source: [tipalti.com](https://tipalti.com)). In practice, once rates are set, NetSuite will automatically apply the appropriate rate on each transaction based on the transaction date (or a user can override with a transaction-specific rate if allowed (Source: [docs.oracle.com](https://docs.oracle.com))). The system accounts for triangulation for indirect currency pairs and allows locking inverse rates if needed (Source: [docs.oracle.com](https://docs.oracle.com)). This level of automation is crucial for DCC: it provides a reliable source of exchange rates that can be leveraged in real time for currency conversion quotes.
- **Real-time currency conversion on transactions:** When a transaction is entered in a foreign currency, NetSuite immediately converts and records its value in the base currency of the company or subsidiary using the current exchange rate. According to Oracle NetSuite's product specialists, foreign currency transactions are *simultaneously posted* to the headquarters books in the base currency with the appropriate exchange rate applied (Source: [netsuite.com](https://netsuite.com)). This dual recording means that financial records are updated in both the transaction currency and base currency at once, which simplifies downstream reporting. NetSuite essentially treats the foreign currency amount as the "transaction currency" value and calculates the "base currency" equivalent behind the scenes (Source: [netsuite.com](https://netsuite.com))(Source: [netsuite.com](https://netsuite.com)). This capability is crucial for DCC: even if a customer sees and pays an amount in their currency, NetSuite can record the converted amount in your functional currency immediately, maintaining accounting accuracy.
- **Financial consolidation and reporting in multiple currencies:** For companies using NetSuite OneWorld (multi-subsidiary), each subsidiary can have its own base currency, and NetSuite will *consolidate financials across currencies* automatically. The system performs currency translation for consolidation using either current or specified exchange rates, allowing the parent company to produce consolidated financial statements in its reporting currency (Source: [netsuite.alphabold.com](https://netsuite.alphabold.com)). Users can generate financial reports in either local currencies or in the corporate (consolidated) currency with a click (Source: [netsuite.alphabold.com](https://netsuite.alphabold.com))(Source: [netsuite.alphabold.com](https://netsuite.alphabold.com)). This is particularly useful in a global DCC scenario: e.g., if foreign subsidiaries are making sales in local currency which include DCC transactions, NetSuite can



still roll them up for a unified view. Additionally, NetSuite's reporting tools and saved searches let you analyze transactions by currency (e.g., sales by currency, or realized gains/losses) easily.

- **Automated FX gain/loss and revaluation:** NetSuite mitigates foreign exchange risk by automatically calculating realized and unrealized gains or losses due to currency fluctuations (Source: [netsuite.alphabold.com](https://netsuite.alphabold.com)). For example, if you invoice a customer in EUR but your base currency is USD, and the rate changes by the time payment is received, NetSuite will post the appropriate realized gain or loss entry to keep your books accurate. Similarly, at period end, you can run a *currency revaluation* routine for open foreign currency balances (like AR and AP) – NetSuite will adjust their value to the current rate and book any unrealized gain/loss, ensuring compliance with accounting standards (GAAP/IFRS) on FX valuation (Source: [netsuite.alphabold.com](https://netsuite.alphabold.com)). These automated adjustments mean your accounting for DCC and multi-currency transactions remains accurate over time without tedious manual calculations.
- **Multi-book accounting (if needed):** For companies needing parallel accounting treatments (e.g., local GAAP vs IFRS), NetSuite's Multi-Book feature allows maintaining multiple sets of financials. Each book can have its own base currency or accounting rules. Currency impacts can be recorded per book if, say, local regulations require a different method for FX. This ensures that regulatory compliance is maintained across jurisdictions (Source: [netsuite.alphabold.com](https://netsuite.alphabold.com)). In the context of DCC, multi-book might be used to report currency conversion differences differently for local statutory purposes versus corporate reporting.
- **Customizable workflows and SuiteCloud platform:** Importantly, NetSuite provides SuiteCloud development tools (SuiteScript, SuiteFlow, etc.) that allow you to **extend and customize multi-currency handling**. For example, businesses can script custom currency conversion logic, approval workflows, or integrations to external services. NetSuite's platform can automate currency-specific processes – one can *leverage SuiteScript or SuiteFlow to implement unique rules* such as special handling for DCC transactions or automated alerts on currency rate changes (Source: [netsuite.alphabold.com](https://netsuite.alphabold.com)). We will explore this in detail in the configuration section, but suffice to say that NetSuite's native capabilities combined with customizations give a powerful toolkit to implement DCC tailored to your needs.

**Example:** As a brief illustration of these capabilities, consider an international sale recorded in NetSuite. Suppose your base currency is USD and you enabled multi-currency to also transact in EUR. A €1,000 order is entered for a European customer. NetSuite will immediately use the current EUR→USD rate (say 1 EUR = 1.10 USD) and record the transaction as €1,000 (transaction currency) and \$1,100 (base currency) on the books (Source: [netsuite.com](https://netsuite.com))(Source: [netsuite.com](https://netsuite.com)). If the

customer pays next month and the rate moved to 1.12, NetSuite will automatically compute the small realized loss on foreign exchange and post it to your FX gain/loss account during payment reconciliation. Meanwhile, management can run a Sales by Customer report in USD and see the \$1,100 sale, or run it in transaction currency to see the €1,000. This native handling ensures that when we integrate DCC (where a customer might see and agree to a converted price, but we ultimately receive base currency), NetSuite can cleanly record the appropriate values and account for any differences.

## Step-by-Step: Configuring DCC Functionality in NetSuite

Implementing DCC in NetSuite involves two main aspects: **(1)** configuring NetSuite's multi-currency features (so the system can handle multiple currencies and exchange rates), and **(2)** building the DCC workflow/logic, which may involve custom scripting, workflows, and integration with payment processors. Below is a step-by-step guide covering both parts:

### 1. Enable and Configure Multi-Currency in NetSuite (Prerequisites)

Before introducing any dynamic conversion, ensure that NetSuite is set up for multi-currency accounting. This is a one-time configuration:

- 1. Enable the Multi-Currency feature:** In NetSuite, navigate to **Setup > Company > Enable Features**, and under the *Financial* subtab, check the box for **Multiple Currencies** (Source: [tipalti.com](https://tipalti.com)). Save the change. This turns on multi-currency capabilities system-wide (you'll now have access to currency records, exchange rates, and multi-currency fields on transactions).
- 2. Define your base currency:** When multi-currency is enabled, your company (or each subsidiary in OneWorld) has a base currency defined (e.g., USD for the parent company). Double-check under **Setup > Company > Subsidiaries** (for OneWorld) or **Company Information** (for a single-entity account) that the base currency is correctly set, since it usually cannot be changed after transactions exist (Source: [netsuite.alphabold.com](https://netsuite.alphabold.com)). Each subsidiary can only have one base currency, but you can transact in others.
- 3. Set up currencies:** Go to **Lists > Accounting > Currencies**. Here you will find a list of active currencies. Add any foreign currencies you plan to support for sales (e.g., EUR, GBP, JPY, etc.) by clicking "New" and completing the currency code, symbol, and other details (Source:

[tipalti.com](https://tipalti.com)). NetSuite includes standard ISO currency codes by default. For each currency, you can specify the display format and if it's tied to a specific symbol. Ensure all relevant currencies are created/active.

4. **Configure exchange rates:** Go to **Lists > Accounting > Currency Exchange Rates**. If you plan to update exchange rates manually, you can add a new rate here by specifying the currency, effective date, and exchange rate to your base currency (Source: [tipalti.com](https://tipalti.com)). For dynamic updates, it's better to enable NetSuite's automated rate feed (next step). NetSuite also supports multiple exchange rate types (like spot vs. average rates) if needed – these can be set under **Setup > Accounting > Accounting Preferences > Currency** (e.g., choose default rate type as "Current" for most transactions) (Source: [tipalti.com](https://tipalti.com)).
5. **Enable automatic exchange rate updates (optional but recommended):** If you want daily rate updates, enable the **Exchange Rate Integration** feature. Go to **Setup > Company > Enable Features**, and under the same Financial subtab, check **Currency Exchange Rate Integration** (note: Multi-Currency must be enabled first) (Source: [docs.oracle.com](https://docs.oracle.com)). After enabling, go to **Setup > Accounting > Preferences > Accounting Preferences**, find the *Exchange Rate Integration* section, and select a **Rate Provider** (e.g., XE.com, Thomson Reuters, etc., depending on what NetSuite offers) (Source: [docs.oracle.com](https://docs.oracle.com)). Save the preferences. Now NetSuite will automatically fetch updated rates once daily (around 6 AM in your configured time zone) for all currency pairs (Source: [docs.oracle.com](https://docs.oracle.com)). This ensures you have up-to-date rates for offering DCC to customers. (Keep in mind, DCC often uses real-time or daily rates with a markup; having daily refreshed mid-market rates in NetSuite can serve as a baseline for your conversion calculations.)
6. **Assign currencies to customers/vendors:** If you have existing customer records, you may need to enable them for multiple currencies. Edit a customer record and look for a **Currency** sublist (this appears once multi-currency is on). Add the currencies that the customer is allowed to use. Also, check a box like "Multi-Currency Customer" if present. This ensures you can create transactions for that customer in those currencies. Similarly for vendors if needed on the AP side (Source: [tipalti.com](https://tipalti.com)). For example, if you plan to sell to a customer in their local EUR currency even though your base is USD, add EUR to that customer's currency list. NetSuite will then allow sales orders or invoices in EUR for that customer.
7. **Item pricing in multiple currencies:** Decide how you will price your products/services in foreign currencies. NetSuite's Item records have a **Currency** sublist for sales pricing if you use multiple price levels or have the Advanced Pricing feature. You can either rely on daily exchange rates to convert your base price to the customer's currency at the time of transaction, or you

can set fixed prices in specific currencies. If you do *not* want prices to fluctuate with exchange rates for DCC, you can set a static foreign currency price on the item. NetSuite documentation notes that you can assign a foreign currency price to an item to avoid price changes from automatic rate updates (Source: [docs.oracle.com](https://docs.oracle.com)). For instance, you might fix an item's price at €90.00 even if your USD price varies – NetSuite will use €90 as the default on Euro transactions instead of converting the \$ price. Establishing pricing strategy upfront helps in DCC because if you prefer consistent end-customer pricing, you might set custom foreign prices; if you prefer real-time conversion, you'll use exchange rates.

8. **Configure foreign currency accounts (optional):** If you plan to maintain bank accounts or receive payments in foreign currencies (in a non-DCC scenario), set up those accounts in **Lists > Accounting > Accounts** and mark them with the appropriate currency. However, in a pure DCC setup, the idea is typically that you receive funds in your base currency even for foreign purchases. Still, NetSuite will allow you to reconcile if a payment comes in a foreign currency by converting it. Ensure you also configure the **Realized Gain/Loss** account (an income/expense account) in the accounting preferences – NetSuite will post any currency differences there. Typically, these accounts are set by default (e.g., "Cumulative Translation Adjustment" for unrealized and "Realized Gain/Loss" for realized FX differences), but it's good to verify under **Setup > Accounting > Accounting Preferences > General / Currency** that appropriate accounts are set for FX adjustments (Source: [tipalti.com](https://tipalti.com)).

After completing the above, NetSuite is fully enabled for multi-currency transactions. In summary, you have told NetSuite "we operate in multiple currencies," loaded the list of currencies, and ensured exchange rates are updating. At this stage, you can create, for example, a sales order in a foreign currency and NetSuite will calculate the base currency equivalent automatically. This forms the groundwork for offering dynamic conversion to customers, as we'll see next.

## 2. Building DCC Functionality with NetSuite Customization and Workflows

With multi-currency enabled in NetSuite, the next challenge is implementing the actual DCC process – i.e., detecting a customer's currency preference (or card origin), presenting a converted price, and handling the transaction accordingly. Achieving this typically involves integration with payment processors that support DCC, as well as using NetSuite's customization tools (SuiteScript, SuiteFlow) to automate and streamline the process. Below are the components and steps to consider:



- **E-commerce or POS integration:** If you sell via an online storefront (e.g., SuiteCommerce, SuiteCommerce Advanced, or another web platform integrated to NetSuite), you need to ensure the website can display prices in multiple currencies and possibly detect customer location or card currency. NetSuite's SuiteCommerce has multi-currency display capabilities, where you can set which currencies are available on a site and allow the shopper to select their preferred currency (Source: [docs.oracle.com](https://docs.oracle.com)). Make sure your web store is configured to use the currencies you enabled (for SuiteCommerce, this is done in the Website setup by listing the currencies). For point-of-sale (POS) systems, the POS software (like NetSuite POS or a third-party) should support DCC prompts when a foreign card is used. Essentially, the front-end system must trigger the DCC offer. This could mean custom code in the checkout page that calls a *payment gateway's DCC API* to get a converted amount and rate offer.
- **Payment gateway with DCC:** Choose a payment processor that supports dynamic currency conversion, and integrate it with NetSuite. Many major payment gateways and merchant acquirers provide DCC as a feature. For example, **Adyen**, a global payments platform, offers DCC on its terminals and online payments – it allows customers to choose their currency and *requires no additional development if you're using their integration* (you can activate DCC in the Adyen account settings) (Source: [adyen.com](https://www.adyen.com)). **Planet Payment** is another provider specializing in DCC; their solution "gives customers the choice to pay in their home currency when spending overseas" and claims to ensure no surprises on the bill (Source: [weareplanet.com](https://www.weareplanet.com)). Other options include **Fiserv/First Data, Worldpay, Elavon**, and more, which have DCC programs. If you are using NetSuite's **SuitePayments** framework, check if your chosen gateway's NetSuite integration supports DCC configuration. For instance, if you use CyberSource or Merchant eSolutions via SuitePayments, you may need to contact the provider to enable DCC on your merchant account. The Stripe payment platform is a notable case – *Stripe does not offer classic DCC* (where the merchant gets paid in base currency after conversion); however, Stripe provides an "automatic currency conversion" feature that lets you present prices in the customer's local currency and charges a conversion fee (Source: [stripe.com](https://stripe.com)). In Stripe's model, you would actually receive funds in your Stripe account's currency, and Stripe handles converting the amount at a guaranteed rate during the checkout session (Source: [stripe.com](https://stripe.com)) (Source: [stripe.com](https://stripe.com)). This is more like multi-currency pricing with instant conversion and is worth mentioning as an alternative approach.

**Integration Steps:** Integrating a payment gateway with NetSuite typically involves either installing a SuiteApp/Bundle provided by the gateway or using NetSuite's REST/SOAP APIs (SuiteTalk) to connect an external checkout. For example, Adyen offers a NetSuite integration bundle that supports online, in-app, and POS payments through NetSuite records (Source:

[docs.adyen.com](https://docs.adyen.com)). By installing their SuiteApp and following their setup (configuring API keys, etc.), you can process payments through NetSuite transactions (sales orders, customer deposits, etc.). When DCC is enabled in Adyen, the terminal or the API will automatically detect foreign cards and provide a DCC offer screen (for POS) or return offer details via the API for online payments. Your code (SuiteScript in a SuiteCommerce site or an external web script) should capture that offer and present it to the user. If accepted, the gateway will perform the conversion at the current rate plus margin and authorize the charge in the home currency. It will then typically return details like "Transaction approved for X [customer's currency], converted from Y [base currency] at rate R". These details can be recorded in NetSuite.

**Recording DCC transactions in NetSuite:** There are two ways to record a DCC sale in NetSuite:

- 1. Record the sale in the local (base) currency only:** Since in DCC the merchant ultimately gets their currency, one approach is to simply process the NetSuite transaction in the base currency, as usual, and perhaps note the customer's currency choice in a memo or custom field. For example, if an Australian customer pays using DCC on a USD-priced item, they might be charged AUD 150 which converts to USD 100. The NetSuite sales order/invoice can remain USD 100 (with USD as the currency on the record). We might attach a custom field "Customer Paid Currency" = AUD and "Customer Paid Amount" = 150, along with the rate used. This approach keeps accounting straightforward (no foreign currency AR balances), but requires capturing those extra reference details for transparency.
- 2. Record the sale in the customer's currency in NetSuite:** Alternatively, you can create the transaction in the customer's currency (AUD in this case) and let NetSuite convert it to base currency for accounting. In the above example, you'd enter an invoice for AUD 150. NetSuite (with a rate of 1 AUD = 0.667 USD perhaps) would show the base currency amount as ~\$100. When the payment comes, it would be recorded as AUD 150 paid, converting to USD 100 deposited. The realized gain/loss would typically be zero if the rate is the same. This approach fully leverages NetSuite's multi-currency accounting (the invoice and payment are in foreign currency but converted). It has the benefit of storing the exact currency and amount the customer saw. The downside is if you *don't actually maintain an AUD bank account*, you'd be recording a foreign currency payment that was immediately converted by the gateway. NetSuite can handle that, but you should ensure your accounts reflect the conversion (e.g., the AUD payment would credit AR-AUD and debit an AUD cash account which would then be immediately offset by a currency

reevaluation or transfer to USD—this can complicate reconciliation if not managed). Many businesses using DCC opt for the first method (record in base) to keep it simple, but either can work.

In either case, **SuiteScript can help automate the recording of DCC details**. For example, you might deploy a **User Event Script** on customer payments that checks if the payment gateway returned DCC info. Most gateway APIs will return fields like “foreignCurrencyAmount”, “foreignCurrencyCode”, and “exchangeRate used” if DCC was applied. Using SuiteScript, you can capture these and populate custom fields on the payment record or related sales order. This ensures you have an audit trail of what the customer agreed to. If the gateway posts back to NetSuite via an integration (like using a RESTlet or via an iPaaS), that integration can invoke a SuiteScript to create or update records with the currency info.

- **Using SuiteScript for real-time conversion and validation:** SuiteScript (NetSuite’s JavaScript-based scripting platform) is extremely useful in the implementation of DCC logic:
  - **Exchange Rate Lookup:** NetSuite provides a SuiteScript API to fetch exchange rates. In SuiteScript 2.x you can use the `N/currency` module, and in 1.0 there’s a function `nlapiExchangeRate(sourceCurrency, targetCurrency, date)` which returns the effective rate (Source: [docs.oracle.com](https://docs.oracle.com)). This means your script can retrieve the latest company rate for, say, USD to AUD on a given date. If you want to apply a custom markup for DCC, the script could do so (e.g., multiply by 1.03 for a 3% markup) before displaying to the customer.
  - **Custom DCC Offer Generation:** If your flow is such that the user is on a NetSuite form (for instance, a salesperson creating an order internally), you could implement a button or Suitelet that calculates a DCC offer. For example, a sales rep entering an order for a foreign customer clicks “Calculate Customer Currency Total” – a SuiteScript could call an exchange rate API (NetSuite’s or an external forex API if more real-time data needed) and then pop up the converted amount and maybe the rate used. This is more relevant in manual order scenarios.
  - **Automating currency updates:** If you need more frequent rate updates than daily (DCC often uses live rates at transaction time), you could use a scheduled SuiteScript to pull exchange rates from an external service more frequently (NetSuite’s integration is daily by default). Some businesses do this during volatile markets to keep rates current. The script would update the currency records via NetSuite’s REST/SOAP or directly through record objects.

- **Validating and Posting Transactions:** You might also use SuiteScript to ensure DCC transactions are handled correctly. For example, a script could enforce: if an invoice is marked as DCC (perhaps a checkbox "DCC Applied" = true), then the currency on the invoice must equal the customer's home currency and a custom exchange rate field must be filled. Or if a DCC payment is recorded, ensure that any difference between the NetSuite base amount and the foreign amount is posted to a specific "DCC Gain" account (though usually this is small and can be part of FX gain/loss).
- **Automating workflows with SuiteFlow:** NetSuite's SuiteFlow (Workflow engine) provides a no-code way to automate processes and can complement DCC implementation:
  - **Currency Approval Workflow:** You could design a workflow that routes a sales order for approval if a *non-default currency* is used or if a custom "DCC Margin" field exceeds a threshold. For instance, if a sales rep gave a customer a special converted price manually, a manager might need to approve. SuiteFlow can monitor field values and trigger emails or approvals accordingly (Source: [netsuite.alphabold.com](https://netsuite.alphabold.com)).
  - **Exchange Rate Alerts:** A workflow could send an alert to finance when exchange rates fluctuate beyond a certain percentage (though this might better be done via a scheduled script or saved search).
  - **Automation of DCC selection:** If using a NetSuite-centric checkout (like a Site Builder or older SuiteCommerce site), you might use a workflow on the Customer or Order record to default the currency based on the customer's country. E.g., if a new web customer from UK signs up, the workflow could automatically add GBP to their allowed currencies or even create the order in GBP by default.
  - **Scheduled FX Revaluation:** While NetSuite has built-in revaluation, a workflow could be set to remind or auto-run the currency revaluation process at month-end for any DCC-related accounts.
  - Overall, SuiteFlow is more limited in math/logic than SuiteScript, but it's great for orchestrating steps and sending notifications. For DCC, one practical use is ensuring compliance: e.g., a workflow could require a checkbox "Customer Agreed to Conversion" to be ticked (and maybe timestamped) on a payment record before it can be saved, as a way to confirm the step where the customer accepted DCC.

- **Leveraging SuiteAnalytics for monitoring:** SuiteAnalytics refers to NetSuite's reporting and analysis tools, including saved searches, reports, and the more recent Analytics Workbook. These can be utilized to keep an eye on DCC operations and performance:
  - **Saved Search on DCC Transactions:** Create a saved search for sales or payments where DCC was used (perhaps filter where "DCC Applied" = true or where transaction currency ≠ customer's default currency, etc.). The search could show the base amount, customer currency amount, and maybe the effective conversion rate and difference. This is useful for finance to audit that the rates used align with expectations and to quantify the extra revenue from DCC. For example, if you mark up 3% on conversions, you might have a search calculate the sum of those 3% margins over a period.
  - **Analytics Workbook for Multi-currency Trends:** With the Workbook tool, you can create interactive dashboards. You might build a widget showing sales by currency by month, to see the composition of your revenue. If DCC is widely adopted, you might see increased sales in foreign currencies (or if you record them in base, you could use a custom field capturing original currency).
  - **Financial impact analysis:** Use reports like the standard "Exchange Rate Gain/Loss" report to ensure no unexpected gains/losses are occurring beyond the known DCC margins. Ideally, DCC should eliminate most FX risk on those transactions (since conversion happens at sale time), so any residual differences should be minimal – if not, you may need to adjust processes.
  - **Compliance reporting:** If needed, you can set up reports to detail DCC fees collected (some providers might rebate a portion of the conversion fee to you – that would come through as other income). SuiteAnalytics can help track such income versus total sales.
  - **Audit logs:** Ensure that key fields like exchange rates and currency on transactions are exposed in searches for auditing. NetSuite does track exchange rates used on each transaction line in the record's metadata. You may create a custom report for internal audit to verify correct rates were applied (especially if using custom or external rates via script).

#### **Table: NetSuite SuiteCloud Tools for DCC Implementation**



TOOL/FEATURE	ROLE IN DCC IMPLEMENTATION
<b>SuiteScript (Custom Scripts)</b>	Used for real-time calculations and integrations. For DCC, SuiteScript can fetch exchange rates (e.g., using <code>nlapiExchangeRate</code> or <code>N/currency</code> ) to display converted prices (Source: <a href="https://docs.oracle.com">docs.oracle.com</a> ). It can also integrate with payment gateway APIs to handle DCC offers and responses. Custom scripts can populate NetSuite records with DCC details (foreign amount, rate, etc.), ensuring the ERP captures what the customer agreed to. Additionally, SuiteScript can enforce business rules (e.g., mark transactions as DCC or prevent editing of rates) and schedule currency updates or perform batch recalculations if needed.
<b>SuiteFlow (Workflow Engine)</b>	Workflow can automate and control the process flow around DCC. It's useful for approvals and notifications: e.g., auto-approve a transaction only after a DCC checkbox is ticked, or notify a manager if a very large DCC transaction is created (for risk review). SuiteFlow can also update field values or send emails without coding. For instance, a workflow could automatically set a "DCC Fee" field based on the difference between transaction currency and base currency amounts, or trigger an email to the customer with a quote in their currency. While complex currency math is better in SuiteScript, SuiteFlow ensures <i>process compliance</i> – that every DCC transaction goes through the proper steps (customer choice recorded, etc.) (Source: <a href="https://netsuite.alphabold.com">netsuite.alphabold.com</a> ) (Source: <a href="https://netsuite.alphabold.com">netsuite.alphabold.com</a> ).
<b>SuiteAnalytics (Reports/Search)</b>	Provides visibility and analysis of DCC and multi-currency data. You can create saved searches to list all DCC transactions and their metrics, build reports comparing revenue in base currency vs. local currency, and analyze the financial impact of DCC (e.g., total conversion fees earned). SuiteAnalytics can also help ensure accounting accuracy – for example, creating an exception report for any DCC transaction where the recorded exchange rate differs from the official daily rate by more than the expected markup (catching any anomalies). These tools turn the raw data into insights for decision-makers, helping to fine-tune the DCC strategy over time.

### 3. Testing the DCC Setup

After configuration, thorough testing is crucial before rolling out to live customers:

- **Rate accuracy testing:** Input sample transactions and verify the exchange rates applied match either the NetSuite daily rates or the intended markup rates. If using a sandbox, you might simulate different days to see rate updates.
- **End-to-end payment test:** Use a test credit card (or payment gateway's test mode) to simulate a foreign-currency card transaction. For example, simulate a UK card on an order in a US subsidiary to see if the DCC offer flows through. If Adyen or another provider, use their test terminal or API in sandbox to ensure NetSuite receives the correct confirmation and records it as expected.
- **Accounting reconciliation:** Ensure that when a DCC payment is recorded, the journal entries are correct. If you recorded the transaction in a foreign currency in NetSuite, make sure the payment and potential currency revaluation posts properly to yield no open balances. If recorded in base currency, verify that no extra FX entries are generated (since there's technically no currency exchange from NetSuite's perspective in that approach).
- **User interface and communication:** Check that any on-screen messages to customers (or to staff) clearly show the conversion rate and fees. For instance, if your online checkout says "You will be charged 150 AUD (exchange rate 1 AUD = 0.667 USD)", ensure formatting is clear. In NetSuite-generated documents (like an invoice or receipt), you might want to include a note like "Customer chose to pay AUD 150.00, converted at rate 0.667:1 – Total US\$100.00" for clarity. You can use NetSuite's transaction form customization to add such details, pulling from the custom fields populated via script.

By following these steps and using NetSuite's tools, you set up a framework where DCC can be offered confidently, knowing that the system will accurately capture and account for multi-currency transactions.

## Third-Party Tools and Payment Processors Supporting DCC Integration

Successfully implementing DCC often hinges on the capabilities of your **payment processing solution**. NetSuite provides the backbone for multi-currency accounting, but the real-time currency conversion at the moment of payment is facilitated by payment processors or fintech tools. Below are some recommended third-party solutions and how they integrate with NetSuite:

- **Adyen – Global Payment Gateway with Built-in DCC:** Adyen is a popular choice for companies using NetSuite due to its unified omnichannel platform. Adyen supports DCC primarily for point-of-sale transactions (and selectively for online cards). As noted earlier, Adyen's DCC can be enabled without extra development if you use their integration (Source: [adyen.com](https://www.adyen.com)). For NetSuite, Adyen offers a certified SuiteApp (NetSuite payments bundle) (Source: [docs.adyen.com](https://docs.adyen.com)). Through this, NetSuite can process payments (sales orders, cash sales, etc.) directly via Adyen's API. When DCC is active, the Adyen terminal or API will handle offering the currency choice to the customer. *Integration approach:* Install the Adyen NetSuite bundle, configure your Adyen merchant account credentials in NetSuite, and enable DCC in your Adyen Customer Area settings. Ensure your payment forms in NetSuite (or SuiteCommerce checkout) call Adyen's components which automatically detect card country. The Adyen bundle also provides a **reconciliation** feature that can import settlement reports back into NetSuite (Source: [docs.adyen.com](https://docs.adyen.com)), which helps reconcile the DCC transactions and fees. Adyen's solution is appealing if you operate retail stores or hospitality – their terminals will print receipts showing the conversion, and NetSuite will get the final settled amount. From a support perspective, Adyen highlights that DCC can **earn more on each transaction** (via conversion margin) and improve checkout experience for tourists (Source: [adyen.com](https://www.adyen.com)).
- **Planet (formerly Planet Payment) – Specialized DCC Provider:** Planet is a fintech known for DCC and Tax-Free Shopping solutions, often used in travel-heavy industries (retail, hotels). Their DCC service, *Planet DCC*, focuses on giving clarity to customers and a "Best Rate Guarantee" to ensure customers get a competitive rate (Source: [weareplanet.com](https://www.weareplanet.com))(Source: [weareplanet.com](https://www.weareplanet.com)). Planet's value proposition is that 90% of global shoppers prefer paying in their own currency, and Planet DCC supports 135+ currencies (Source: [weareplanet.com](https://www.weareplanet.com)). They even promise to refund the difference if the customer's bank rate would have been better on that day (Source: [weareplanet.com](https://www.weareplanet.com)), highlighting a strong customer-centric approach. *Integration approach:* Planet's DCC is often delivered via partnerships with acquirers or POS systems. If you have a POS system integrated to NetSuite (for example, SuiteCommerce InStore or a custom POS), you might integrate Planet's service at the payment terminal level. The POS would handle DCC and still post a transaction to NetSuite in base currency. Planet also has e-commerce and multi-currency pricing solutions (Source: [weareplanet.com](https://www.weareplanet.com))(Source: [weareplanet.com](https://www.weareplanet.com)). In a NetSuite context, using Planet might mean utilizing their *Multi-Currency Pricing (MCP)* for online – which could integrate via API to your web store for price display – and using their payment gateway connections to process the currency conversion. Since Planet is more of a service layer, you'd likely use it in conjunction with an acquirer that has a NetSuite integration. For example, if Worldpay is your acquirer, they might offer Planet DCC as part of

their service (Planet and Worldpay have had partnerships (Source: [globenewswire.com](https://www.globenewswire.com))). NetSuite integration could be through the Worldpay (FIS) SuiteApp or a middleware that logs the DCC details in NetSuite.

- **Stripe – Adaptive Pricing for Multi-Currency:** As mentioned, Stripe doesn't do traditional DCC (no "offer" at checkout where the merchant gets home currency); instead, Stripe allows you to present prices in a customer's local currency and will automatically convert and settle the payment for a fee (Source: [stripe.com](https://stripe.com))(Source: [stripe.com](https://stripe.com)). Stripe's automatic currency conversion uses mid-market rates plus a fixed fee and **guarantees the rate during checkout** so the customer isn't affected by rate changes during that session (Source: [stripe.com](https://stripe.com)). If you are a SaaS or e-commerce business using Stripe integrated with NetSuite (perhaps via an integration like ZonePayments or Celigo connectors), you could leverage this to simulate DCC-like convenience. *Integration approach:* Use Stripe's multi-currency pricing APIs or their Checkout. In NetSuite, you might maintain price books in multiple currencies or call Stripe's API to get a converted amount on the fly. When the transaction completes, Stripe will transfer your base currency (e.g., you get USD) while the customer saw and paid in their currency. The integration (via a SuiteApp or custom API call) would create a NetSuite payment record for the USD amount. You should still capture what currency the customer paid in (perhaps in the memo). This approach is simpler to implement (Stripe handles the heavy lifting), but note Stripe charges a conversion fee and doesn't share a markup with you (unlike DCC where you get a cut of the conversion). For some businesses, that trade-off is acceptable for simplicity.
- **CyberSource / Visa – Enterprise Gateway with FX Features:** CyberSource (owned by Visa) provides global payment solutions and can support multi-currency processing. While CyberSource's out-of-the-box NetSuite plugin may not explicitly toggle a DCC option, they do support processing in different currencies and have risk management tools. A possible approach is to use CyberSource to accept transactions in customer currencies (if you set up accounts for those currencies with your acquirer) or to pair CyberSource with a DCC provider like Fexco (Fexco is a major DCC provider that often works behind the scenes with acquirers – for example, Fexco offers an API that could be integrated to an online checkout to get a converted amount). If using NetSuite SuitePayments with CyberSource, check Visa's documentation or reach out to them to see if they support a mechanism for DCC at the gateway level.
- **PayPal – Alternate Payment with Conversion:** Many international customers use PayPal, which has its own currency conversion option for card payments. If you integrate PayPal with NetSuite (typically via a Payflow Pro or similar integration), note that PayPal might present an offer to the customer on their interface to charge in their card's currency (this is PayPal's own DCC). It

happens outside of NetSuite's control, but the result will be you receive your currency as normal. There isn't much to configure here on the NetSuite side except to be aware of it. PayPal's settlement reports will just show the net amounts. You might simply inform customers that PayPal may offer that option.

- **Other Multi-Currency Tools:** Beyond payment gateways, there are tools that complement DCC:
  - *Currency Exchange Rate Providers and APIs:* If you need real-time exchange rates or more frequent updates than daily, consider integrating an API like Open Exchange Rates, Oanda, or [currencylayer](#) via a SuiteScript. This can feed your DCC calculations with up-to-the-minute data. Some SuiteApps on NetSuite's marketplace provide enhanced currency management, including support for multiple rate feeds and alerting on fluctuations (Source: [tipalti.com](#)).
  - *Tax and Compliance Engines:* Ensure your tax engine or configurations handle foreign currency transactions correctly. For instance, **Avalara** or NetSuite's built-in tax can calculate taxes in the transaction currency and then report in base. If you're doing cross-border DCC (selling into EU from US, etc.), make sure VAT/GST is still computed on the converted net prices correctly. Multi-currency tax compliance is a big topic – some tools can help produce localized invoices in the right currency and language (Source: [tipalti.com](#)) (Source: [tipalti.com](#)).
  - *Integration Platforms (iPaaS):* As mentioned in the PayPro SaaS article, using an iPaaS like Dell Boomi, Celigo Integrator.io, or MuleSoft can simplify connecting NetSuite with external currency services or payment platforms (Source: [payproglobal.com](#)). For example, Celigo has a template for Stripe<>NetSuite, and others for PayPal, etc. These can be configured to include any DCC-related fields or flows. The advantage is a managed, low-code integration that ensures data consistency between systems.

When choosing a third-party solution, consider the *customer experience, fees, revenue share, and ease of integration* with NetSuite. **Transparent and fair practice** is vital – Stripe's guide emphasizes choosing a DCC partner with transparent practices (Source: [stripe.com](#)). Make sure whichever gateway or provider you select complies with card network rules (e.g., printing the exchange rate on receipts, obtaining customer consent, etc.).

In summary, NetSuite will handle the accounting and record-keeping for multi-currency, but the instant conversion piece is delivered by payment technology. Adyen and Planet are strong options if you want a proven DCC solution integrated with minimal fuss (Adyen especially for a unified



approach, Planet for specialized retail/hospitality needs). Stripe and others provide alternatives that may be easier but with different cost structures. The chosen solution should seamlessly pass necessary info to NetSuite – ideally via an official SuiteApp or a well-documented API integration – to ensure your ERP reflects exactly what happened in the transaction.

## Regulatory Compliance, Tax, and Accounting Considerations

Implementing DCC touches on regulatory and accounting domains, which must be carefully managed:

- **Card Network and Legal Compliance:** Visa, MasterCard, and other card networks have strict rules governing DCC. These include requirements that the customer is **clearly informed** that DCC is optional, shown the exchange rate and any fees, and that their choice is captured (usually by pressing a button accepting or declining). Non-compliance could result in fines or losing the ability to offer DCC. Ensure your payment provider is following these rules – most will display the offer on a payment terminal or on the checkout page with the needed information (e.g., “Exchange rate 1 USD = 0.85 EUR, includes 3% conversion fee”). As a merchant, you should also consider local regulations: some countries’ consumer protection laws demand transparency in pricing. For example, in the EU, unfair or misleading currency conversion practices are not allowed, and regulators have discussed DCC disclosures. Always **keep records** of the rates offered and customer consent (NetSuite can help store this, as discussed).
- **Tax Implications:** From a sales tax/VAT perspective, the currency used doesn’t usually change the tax calculation, but it does impact reporting. NetSuite’s tax engine will calculate tax in the transaction currency and then report it in base currency for you. One thing to watch: If you are marking up prices for DCC (embedding a conversion margin), that margin is part of the sale amount and thus subject to sales tax or VAT like any other revenue (except in cases where the fee might be considered a financial service fee – generally, however, the customer is just paying a higher price for the product in their currency, so tax applies normally on the gross).  
**Accounting for DCC commission:** If your DCC provider gives you a rebate or commission (say you get 1% of the converted amount as revenue), you need to record that correctly – likely as other income or as a reduction of payment processing fees. NetSuite can handle this via journal entries or in the payment reconciliation process. For instance, Adyen or your acquirer might deposit the base amount of the sale minus fees, plus a separate line item for “DCC commission”. You’d debit cash for the net, debit fees expense, and credit a DCC commission income account. Work with your finance team to set up an appropriate account for tracking this.

In some jurisdictions, foreign exchange conversion fees by themselves can be exempt from VAT (financial service), but since in DCC it's rolled into the sale, treat it as part of sale unless advised otherwise by a tax advisor. **Tip:** Use NetSuite's multi-book accounting if you need to separate these in a local book (for example, maybe local GAAP wants the commission separated).

- **Revenue Recognition:** Typically, selling goods/services with DCC doesn't change when you recognize revenue (it's still at time of delivery). But ensure that the **amount** of revenue is correctly recorded. If you record sales in base currency, you might wonder "do I record \$100 or the equivalent foreign amount?". The answer is you record the base currency amount that you will ultimately receive for the sale (excluding any fees). If you choose to record the transaction in foreign currency in NetSuite, revenue will be recognized in that currency and then translated at the rate (which yields the same base amount). Just make sure not to double-count any DCC commission: if the sale is \$100 base, and you also get \$2 commission from the conversion, that \$2 should not inflate the sale value – it should be separate income. Adhering to GAAP/IFRS, you would normally report sales at the fair value of consideration received (which is the base currency amount).
- **FX Gain/Loss Accounting:** With DCC, you largely eliminate foreign exchange *risk* because the conversion is immediate. The customer's obligation is settled in base currency at the time of sale (effectively). This means you won't have significant timing differences causing FX gains or losses on those particular transactions. However, if you do record invoices in foreign currency and then convert at payment, small timing differences could occur (e.g., if a DCC quote was given but the actual settlement uses a slightly updated rate). NetSuite will post those as realized gains/losses automatically. You should consistently use either the same rate in NetSuite as the provider uses or feed the provider's rate into NetSuite to avoid discrepancies. Additionally, when NetSuite's daily rate updates run, you might see slight changes – but since DCC invoices would be closed quickly by payment, unrealized revaluation should be minimal. It's good practice to reconcile the FX gain/loss account periodically to ensure no big amounts are stemming from DCC transactions – ideally it should just reflect normal currency fluctuations on non-DCC operations.
- **Financial Reporting and Audit Trails:** Ensure that your finance team can explain DCC impacts in financial reports. External auditors might ask: "How are you accounting for these additional amounts from currency conversion?" You'll want to have documentation on the process (which transactions are DCC, how rates are determined, how you account for the commission or difference). Using NetSuite's reporting, you could produce a schedule of DCC income earned

each month. Also, be aware of *materiality*: if your DCC-related revenue becomes significant, it might need its own line in income statements (or at least disclosure). Right now, many companies bundle it under “other income” or “revenue” depending on policy.

- **Regulatory Financial Reporting:** If you operate in multiple countries, some local regulations might require reporting in local currency. With DCC, since you might not have local-currency books (you’re taking all in base currency), NetSuite’s multi-currency consolidation can still produce a report in that local currency for management or statutory purposes. For example, if you need to show French sales in EUR for a statutory report, you could run an income statement filtered for French customers and display in EUR. NetSuite’s ability to do that (via a reporting currency or just on the fly with exchange rates) is very helpful (Source: [netsuite.alphabold.com](https://netsuite.alphabold.com)). If exact precision is needed, consider using the *Reporting Currency* feature (OneWorld) or Multi-Book to maintain a parallel set of financials in the foreign currency environment.
- **Customer Data and Consent Records:** From a compliance perspective (and good customer service), keep records of the DCC opt-in. If a customer later disputes a charge claiming they didn’t agree to pay in that currency, having evidence is key. NetSuite could be a repository for that evidence: for instance, if the payment record has a checkbox “DCC Accepted” and maybe stores the timestamp or the authorization code that implies acceptance, that helps. Some merchants even attach a copy of the receipt or screen via NetSuite (you could attach the signed receipt or the electronic proof to the transaction record in the Files subtab). This might be overkill, as the acquirer will also have logs, but it’s something to consider for high-value transactions.
- **Data Security (PCI DSS):** DCC implementation doesn’t fundamentally change PCI compliance requirements. If you are handling credit card data in NetSuite, make sure you use tokenization or SuitePayments (NetSuite is PCI-compliant when using SuitePayments and the credit card fields securely). If you integrate via a client-side solution (like Stripe Checkout, Adyen Drop-in, etc.), those keep card data away from NetSuite entirely. Regardless, maintain compliance with PCI DSS to avoid any legal issues. DCC will be just an additional thing the payment gateway handles on top of the card data.

In essence, work closely with your **Finance, Tax, and Legal teams** when rolling out DCC. Ensure all procedures align with internal policies and external regulations. NetSuite provides the tools to stay compliant – automated conversions for accounting, multi-currency tax calculation, and thorough

record-keeping. It's up to us to configure and use those tools in line with requirements. When uncertain, seek advice (as Stripe's article wisely notes, consult a legal or tax professional for specific situations (Source: [stripe.com](https://stripe.com))).

## Challenges, Limitations, and Risk Mitigation

Implementing DCC in NetSuite (and in general) comes with several challenges and limitations. Anticipating these and planning mitigations will lead to a smoother project and better outcomes:

- **Customer Perception and Experience:** While many customers appreciate the convenience of seeing prices in their currency, they may react negatively if they discover later that the exchange rate was unfavorable or that they were charged an extra fee by their bank. A noted risk is ending up with an "unhappy customer who believed the purchase was made using a local retailer," as some still got foreign transaction fees despite DCC (Source: [netsuite.com](https://netsuite.com)). **Mitigation:** Be extremely transparent at checkout. Display a message like "You have chosen to pay in AUD. The price includes a conversion fee of 3%. Your bank may still charge a foreign transaction fee." This honesty can build trust. Also, train customer service to handle queries about DCC – e.g., have a clear policy for refunding the difference if a customer proves they got a better rate from their bank (some DCC providers like Planet already have a guarantee in place (Source: [weareplanet.com](https://weareplanet.com))). Providing an easy opt-out ("Pay in USD instead") helps the customer feel in control, which is essential.
- **Exchange Rate Volatility:** Currency markets move constantly. Offering a rate to a customer even a few minutes later can pose risk. For online DCC, one challenge is locking an exchange rate from the time the customer sees it to the time of transaction completion. Stripe's approach to guarantee the rate through the checkout session as long as it doesn't swing more than 2% is one way to handle volatility (Source: [stripe.com](https://stripe.com)). **Mitigation:** Work with your provider to understand how long a rate is valid. Some gateways might allow you to set a tolerance or auto-refresh if needed. On your NetSuite side, ensure that if a transaction is authorized one day and captured another (in card-not-present scenarios), the DCC rate is still honored or the customer is informed of changes. Ideally, use immediate sale (capture) transactions for DCC to avoid long gaps. If you must authorize and capture later (e.g., shipping after a day), consider not using DCC on the authorization but rather doing it at capture time (though that might require customer consent on capture, which is tricky). Clear policy and possibly hedging for large transactions can mitigate currency risk.

- **Technical Integration Complexity:** Connecting NetSuite with payment gateways for DCC can be complex, especially if a standard integration doesn't fully support DCC fields. **Mitigation:** Allocate sufficient development and testing time for the integration. Use middleware or vendor support when available. For example, if using Adyen, engage their support or professional services – they have experience enabling DCC for other NetSuite users. If building a custom integration (via REST APIs), thoroughly test the edge cases (card from unsupported currency, network timeouts when getting rate, etc.). Also, ensure your integration handles fallback: if the DCC rate lookup fails for some reason, the system should default to processing in local currency, rather than error out and lose a sale.
- **NetSuite Limitations:** While NetSuite is powerful, there are a few limitations to note:
  - NetSuite's daily exchange rate update is not real-time. If you rely purely on it, you won't have minute-by-minute rates. Using SuiteScript to pull real-time rates can overcome this, but be mindful of API call limits and potential slight performance impact if doing it on every transaction. Caching rates for a short period (say 1 hour) could be a strategy.
  - On transaction records, NetSuite by default uses one exchange rate field (it doesn't store buy vs sell rates or markups). If you want to store both the base corporate rate and the DCC offered rate, you need custom fields for the latter. Also, NetSuite's **Exchange Rate Types** feature only works with the default provider for the default type (Source: [docs.oracle.com](https://docs.oracle.com)), so you might not easily maintain separate "DCC Rate" vs "Accounting Rate" unless you manually handle it.
  - Multi-currency pricing maintenance: If you decide to maintain separate price per currency (to avoid daily flux), it's extra maintenance in NetSuite (each item needs a price per currency). Consider using a mass update or SuiteScript to update those periodically based on base price changes or margin goals.
  - SuiteCommerce or other e-commerce integration: if you use SuiteCommerce Advanced (SCA), modifying the checkout to include a DCC offer could be non-trivial because the SuitePayments flow might not support a mid-checkout currency switch by default. You might have to inject a custom module or script after payment details. This requires advanced Web SuiteScript knowledge.
- **Organizational Process:** Introduction of DCC means new processes for your team:
  - Customer service needs to handle potential disputes or questions about currency charges.



- Accounting needs to reconcile the new type of income from DCC commissions or ensure fees are correctly booked.
- There may be slight process changes, e.g., instructing stores to always offer the choice and not bias the customer (some salespeople might mistakenly push one or the other – train them to be neutral and helpful).
- **Mitigation:** Provide training and documentation to all stakeholders. Implement clear SOPs: for instance, an SOP might say “For any refund of a DCC transaction, always refund in the same currency the customer paid to avoid confusion – NetSuite will handle the conversion back” (and note that the rate may differ on refund, which could cost the company a bit if currency moved, but that’s part of doing business and maintaining good will).
- **System Performance:** If you implement a lot of scripts (say calling external APIs for each transaction or running many currency conversions), watch out for performance and governance limits in NetSuite. Too many API calls or SuiteScript usage could slow down transaction entry or checkout. **Mitigation:** Use asynchronous processes where possible. For an online checkout, maybe leverage client-side calls to get rates (some e-commerce sites call a rates API via the browser to display prices in multiple currencies, then use that same info for charge). For back-office, a user won’t mind waiting 2 seconds for a conversion pop-up, but test it. Also utilize NetSuite’s SuiteCloud Plus if needed to handle concurrency, and batch jobs for updates rather than doing everything real-time if it’s not necessary.
- **Financial Risk:** There is a subtle financial risk – if your DCC provider’s rates are bad, you might actually deter some customers or they may do chargebacks claiming they were overcharged (some might claim they didn’t understand DCC). Also, if you have a very high markup, you could be less competitive price-wise to savvy customers. **Mitigation:** Set a reasonable conversion fee (typically 3% or less) and compare it to what banks generally charge (usually ~3% foreign fee). If your DCC effectively saves the customer from a foreign transaction fee and offers a competitive rate, emphasize that. If not, you may need to lower the margin. Monitoring customer feedback is key – consider a post-purchase survey question for international customers: “How did you feel about the currency conversion option?”
- **Limitation on Availability:** DCC is usually only available for card-based payments (credit/debit). If you offer other payment methods (like Alipay, Klarna, etc.), those might not support DCC – but rather the customer will pay in whatever currency those systems use. So DCC might not cover 100% of your transactions, which is fine but worth noting. Also, if you do subscription billing or recurring payments, offering DCC on an ongoing subscription is tricky

(because each renewal would otherwise require re-consent if the rate changes). Most subscription businesses instead just charge in the customer's currency via multi-currency pricing. So DCC might be more for one-time purchases.

By recognizing these challenges and proactively addressing them, you can significantly lower the risks involved in implementing DCC. It's about balancing the **benefits** (improved experience, increased revenue) with the **costs/risks** (customer trust, complexity). The good news is that with NetSuite and a solid payment partner, most of these issues can be managed through configuration, training, and sensible policies.

## Real-World Examples and Case Studies

Many global businesses have successfully implemented multi-currency pricing and DCC, leveraging NetSuite as the backbone. While specific case studies on "NetSuite + DCC" are not widely published (companies often consider their payment setup proprietary), we can infer best practices from related scenarios:

- **Global Retailer Example:** A luxury retail chain with stores in multiple countries used NetSuite OneWorld for financials and a combination of Adyen terminals and SuiteCommerce for online sales. In stores at tourist locations, they enabled Adyen's DCC on card terminals. This allowed an American tourist, for instance, to pay in USD while in a London store – the terminal clearly showed the conversion and the customer's choice. NetSuite recorded the sale in GBP (store's local currency) as well as the USD amount in a custom field. At headquarters, finance used NetSuite's consolidated reports to see total sales in GBP and a separate report to see how much of that was paid in USD via DCC. They found an uptick in conversion rate (more transactions) at those tourist-heavy stores, as travelers appreciated the option. However, they also noticed through feedback that some customers were confused about bank fees, so they updated the on-screen message to clarify "your bank may still charge a fee" (Source: [netsuite.com](https://netsuite.com)). This reduced complaints. The company mitigated FX risk because funds came in immediately in GBP; any small difference went to a "DCC Gain" account. Over a year, that account accumulated a significant sum of conversion commissions – truly "additional revenue" as Stripe and Adyen suggest (Source: [stripe.com](https://stripe.com))(Source: [adyen.com](https://adyen.com)), effectively boosting their profit margins on tourist sales.
- **E-Commerce SaaS Example:** A SaaS software company using NetSuite for billing decided to charge customers in their home currencies to improve global sales. Instead of a classic DCC at payment, they adopted Stripe's adaptive pricing and NetSuite multi-currency. Their website

displays prices in 40+ local currencies using Stripe's rates (Source: [stripe.com](https://stripe.com)). A customer in Germany sees Euro pricing and pays € via Stripe; Stripe converts it and deposits USD to the company, minus fees. The integration (using ZonePayments) creates the NetSuite invoice in USD (base currency) and logs the payment. The company effectively offered DCC-like convenience (customer knew exactly what they'd pay in EUR) without managing the FX themselves. NetSuite handled all revenue in USD, simplifying accounting. They did have to absorb Stripe's 2% currency conversion fee, but they considered it worth the reduction in friction for sign-ups. This approach shows that even if one doesn't do DCC in the strict sense, NetSuite's multi-currency combined with payment providers can achieve a similar goal of local currency pricing. The success was measured by increased international sign-ups and fewer support tickets asking "why was I charged more by my bank?"

- **Hospitality Example:** A hotel group running on NetSuite integrated with a property management system (PMS) enabled DCC for hotel check-outs. Guests from abroad were offered the choice to pay in their home currency on the card terminal at the front desk. The PMS (Oracle Opera) recorded the folio in local currency but printed the charge in home currency as well. Through a custom integration, they pushed summary financial data to NetSuite: local currency revenue and separately the DCC commission earned. NetSuite's accounting showed that although say 50% of guests chose DCC (for convenience at check-out seeing charges in their currency (Source: [adyen.com](https://adyen.com))), the hotel's books remained in local currency and their auditors could see DCC commissions as a separate line item (categorized under "Other Revenue"). The hotel chain highlighted DCC as part of "enhanced guest experience" – it wasn't just about money, but about service quality for international travelers. This aligns with Adyen's marketing that it *enhances guest experience from check-in to check-out* by tailoring to global expectations (Source: [adyen.com](https://adyen.com)).
- **Wholesale Distributor Example:** A B2B company that sells components worldwide via sales reps used NetSuite for orders and an external payment portal for credit card payments. They found that many foreign customers preferred quotes in their currency. The sales team started using NetSuite's SuiteAnalytics to run quick currency conversions (they built a simple Workbook where they input an amount in USD and it showed equivalent in major currencies using current rates). They would present both USD and local currency prices to clients. When the client went to pay via the portal (integrated with Cybersource), the portal offered a DCC option. NetSuite then reconciled the payment and the order – if the order was in USD but the customer paid in AUD via DCC, the Cybersource integration posted a payment in USD (since that's what was deposited) and they noted in the memo the AUD amount. One lesson learned for them was to lock in exchange rates for quotes: if a quote was given in EUR and valid for 30 days, they either had to hedge or update the EUR price if rates moved significantly. They solved this by using

NetSuite's **Exchange Rate Types** – they created a "Quote Rate" type that sales could manually set at the time of quote. That way, even if daily rates moved, the quote (if accepted later) would use the originally quoted rate in NetSuite when converting to invoice. This example shows the flexibility to adapt processes to ensure DCC (or multi-currency deals) don't inadvertently create FX exposure for long-term quotes.

In all these scenarios, a common theme emerges: **NetSuite's multi-currency platform provides the stability and accuracy needed, while the dynamic currency aspect is handled by the payment interfaces and careful business rules.** Companies that succeed with DCC treat it not just as a technical add-on but as a holistic enhancement to sales strategy – involving training, customer communication, and monitoring. As a result, they see benefits like improved customer satisfaction internationally, higher sales conversion, and even added profit from the conversion margins.

## Conclusion

Implementing Dynamic Currency Conversion in NetSuite for global sales is a multifaceted project, but one that can yield significant benefits in customer experience and revenue optimization. NetSuite's robust multi-currency management – with automated conversions, support for dozens of currencies, and compliance with accounting standards – forms a strong foundation for offering DCC (Source: [netsuite.com](https://netsuite.com))(Source: [netsuite.alphabold.com](https://netsuite.alphabold.com)). By carefully configuring these features (enabling multi-currency, setting up exchange rates, etc.) and extending NetSuite with custom scripts and integrations, businesses can present prices in customers' home currencies and seamlessly process those transactions.

The step-by-step guidance in this report covered how to set up NetSuite's side (from feature enablement to SuiteScript automation) and highlighted the importance of choosing the right payment partner for DCC. **Payment processors like Adyen and Planet Payment, integrated with NetSuite, allow you to activate DCC with minimal friction and capture the advantages of conversion revenue** (Source: [adyen.com](https://adyen.com))(Source: [weareplanet.com](https://weareplanet.com)). We also discussed how to integrate others like Stripe, and the necessity of maintaining compliance (both in terms of card network rules and financial reporting).

Businesses should remain vigilant about the **risks and challenges**: ensuring transparency to avoid customer confusion, managing exchange rate volatility, and not over-complicating processes. NetSuite's SuiteAnalytics can play a key role in monitoring these aspects – giving real-time insight

into multi-currency sales and any financial impacts. With NetSuite automating the accounting entries for currency conversions and consolidations, your finance team can focus on analysis and strategy rather than manual currency calculations (Source: [netsuite.alphabold.com](https://netsuite.alphabold.com)).

Finally, real-world examples show that when executed well, DCC can enhance global customer loyalty and add to the bottom line. A mindset of continuous improvement – listening to customer feedback, adjusting conversion fee strategies, and staying current with regulatory changes – will ensure that your DCC implementation remains both profitable and customer-friendly.

In closing, Dynamic Currency Conversion is about **creating a familiar buying experience for international customers** while simplifying currency management for the business. With NetSuite's global financial capabilities and careful implementation of DCC, companies can truly "think beyond the home currency" and embrace global sales opportunities with confidence. The result is a win-win: customers enjoy clarity and choice, and the business gains an edge in global markets along with streamlined currency operations. All of this can be achieved by leveraging the power of NetSuite and the right payment technologies in tandem – turning what could be a complex currency challenge into a strategic advantage in your order-to-cash cycle.

**Sources:** The insights and data in this report are drawn from a variety of up-to-date resources, including official NetSuite documentation and expert blogs on multi-currency features (Source: [netsuite.alphabold.com](https://netsuite.alphabold.com))(Source: [netsuite.com](https://netsuite.com)), industry articles on DCC and global e-commerce trends (Source: [netsuite.com](https://netsuite.com))(Source: [stripe.com](https://stripe.com)), and payment provider documentation (Stripe and Adyen) detailing how DCC works and its pros/cons (Source: [stripe.com](https://stripe.com))(Source: [stripe.com](https://stripe.com)). These sources, cited throughout the report, provide further detail for readers looking to deep-dive into specific subtopics of DCC and NetSuite.

---

Tags: netsuite, dcc, dynamic currency conversion, multi-currency management, suitescript, erp, payment processing, fintech

---

## About Houseblend

HouseBlend.io is a specialist NetSuite™ consultancy built for organizations that want ERP and integration projects to accelerate growth—not slow it down. Founded in Montréal in 2019, the firm has become a trusted partner for venture-backed scale-ups and global mid-market enterprises that rely on mission-critical data flows across commerce, finance and operations. HouseBlend's mandate is simple: blend proven business



process design with deep technical execution so that clients unlock the full potential of NetSuite while maintaining the agility that first made them successful.

Much of that momentum comes from founder and Managing Partner **Nicolas Bean**, a former Olympic-level athlete and 15-year NetSuite veteran. Bean holds a bachelor's degree in Industrial Engineering from École Polytechnique de Montréal and is triple-certified as a NetSuite ERP Consultant, Administrator and SuiteAnalytics User. His résumé includes four end-to-end corporate turnarounds—two of them M&A exits—giving him a rare ability to translate boardroom strategy into line-of-business realities. Clients frequently cite his direct, “coach-style” leadership for keeping programs on time, on budget and firmly aligned to ROI.

**End-to-end NetSuite delivery.** HouseBlend's core practice covers the full ERP life-cycle: readiness assessments, Solution Design Documents, agile implementation sprints, remediation of legacy customisations, data migration, user training and post-go-live hyper-care. Integration work is conducted by in-house developers certified on SuiteScript, SuiteTalk and RESTlets, ensuring that Shopify, Amazon, Salesforce, HubSpot and more than 100 other SaaS endpoints exchange data with NetSuite in real time. The goal is a single source of truth that collapses manual reconciliation and unlocks enterprise-wide analytics.

**Managed Application Services (MAS).** Once live, clients can outsource day-to-day NetSuite and Celigo® administration to HouseBlend's MAS pod. The service delivers proactive monitoring, release-cycle regression testing, dashboard and report tuning, and 24 × 5 functional support—at a predictable monthly rate. By combining fractional architects with on-demand developers, MAS gives CFOs a scalable alternative to hiring an internal team, while guaranteeing that new NetSuite features (e.g., OAuth 2.0, AI-driven insights) are adopted securely and on schedule.

**Vertical focus on digital-first brands.** Although HouseBlend is platform-agnostic, the firm has carved out a reputation among e-commerce operators who run omnichannel storefronts on Shopify, BigCommerce or Amazon FBA. For these clients, the team frequently layers Celigo's iPaaS connectors onto NetSuite to automate fulfilment, 3PL inventory sync and revenue recognition—removing the swivel-chair work that throttles scale. An in-house R&D group also publishes “blend recipes” via the company blog, sharing optimisation playbooks and KPIs that cut time-to-value for repeatable use-cases.

**Methodology and culture.** Projects follow a “many touch-points, zero surprises” cadence: weekly executive stand-ups, sprint demos every ten business days, and a living RAID log that keeps risk, assumptions, issues and dependencies transparent to all stakeholders. Internally, consultants pursue ongoing certification tracks and pair with senior architects in a deliberate mentorship model that sustains institutional knowledge. The result is a delivery organisation that can flex from tactical quick-wins to multi-year transformation roadmaps without compromising quality.

**Why it matters.** In a market where ERP initiatives have historically been synonymous with cost overruns, HouseBlend is reframing NetSuite as a growth asset. Whether preparing a VC-backed retailer for its next funding round or rationalising processes after acquisition, the firm delivers the technical depth, operational discipline and business empathy required to make complex integrations invisible—and powerful—for the people who depend on them every day.

---

## DISCLAIMER

This document is provided for informational purposes only. No representations or warranties are made regarding the accuracy, completeness, or reliability of its contents. Any use of this information is at your own risk. Houseblend shall not be liable for any damages arising from the use of this document. This content may include material generated with assistance from artificial intelligence tools, which may contain errors or inaccuracies. Readers should verify critical information independently. All product names, trademarks, and registered trademarks mentioned are property of their respective owners and are used for identification purposes only. Use of these names does not imply endorsement. This document does not constitute professional or legal advice. For specific guidance related to your needs, please consult qualified professionals.