Coupa and NetSuite, connected end to end through Celigo

The Source-to-Pay integration I designed, built and took live for The Jockey Club: 24 Celigo flows covering suppliers, chart of accounts, invoices, payments, expenses and budgets across nine group entities.

Coupa

Procurement, invoicing, expenses

- Supplier information
- Suppliers and remit-to
- Billing-string lookups
- Invoices and credit notes
- Coupa Pay
- Expense reports
- Budget lines

Celigo integrator.io JavaScript hooks, lookups and filters on every flow; idempotent by design

◀ NetSuite to Coupa Coupa to NetSuite ▶

Suppliers

`01` `01B` `01C` `01D` `02` `03` `03C`

Onboarding from approved Supplier Information, bank and remit-to, activation, entity stamping

Finance approves in NetSuite before the supplier goes live

Chart of accounts

`04` `04F` `04A` `04C` `04D`

GL combinations as a six-level billing-string hierarchy, with label sync and deactivation

~159,000 combinations across nine entities

Invoices & credits

`07`

Vendor bills and credits with item-master resolution, credit application and write-back

777-invoice backlog re-imported

Payments

`08a` `08b` `09`

Paid status at applied-line grain, void reversals, Coupa Pay virtual-card settlement

Sync ledger with composite idempotency keys

Expenses

`05` `06`

Expense reports at net-of-VAT with attendee capture, expense payments written back

Subsidiary-aware expense categories

Budgets

`11` `11B`

Budget lines from SuiteAnalytics Connect; consumption and drift returned to NetSuite

NetSuite changes posted as audited adjustments

NetSuite OneWorld

Ledger, vendors, payments

- Vendors and subsidiaries
- Bank-detail records
- GL validation combinations
- Vendor bills and credits
- Vendor payments
- Expense reports
- Budgets

Integration Control Centre

NetSuite Suitelet (SuiteScript 2.1) giving Finance and support one view of every Celigo flow

Coupa ↔ NetSuite — integration control centre

Celigo · REST · the Jockey Club · last 7 days

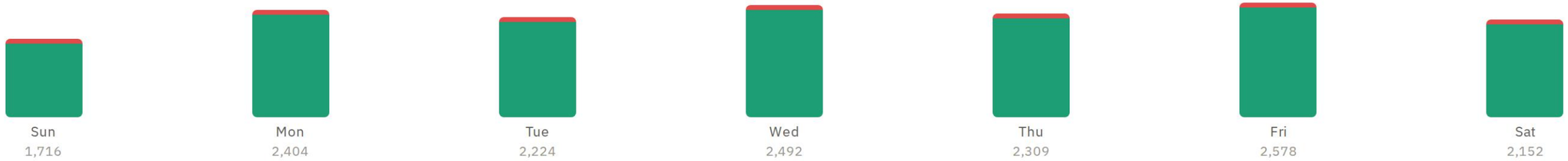
Open in Celigo | Refresh

4 record-level errors across runs in the last 24h (0.2%) — open the Errors tab to see which step failed

Runs today	Records synced 24h	Success rate 24h	Errors 24h	Stale flows
186	2,148	99.8%	4	0

Overview | Flows | Run log | Errors (4)

Records processed, last 7 days



F01 Supplier onboarding

Coupa → NetSuite — Success

Last run	3 min ago	Runs (7d)	2,016
Synced today	64	Success	100%

F02 Supplier status & activation

NetSuite → Coupa — Success

Last run	3 min ago	Runs (7d)	2,016
Synced today	36	Success	100%

F03C Bank, tax & remit-to

Coupa → NetSuite — Success

Last run	2 min ago	Runs (7d)	2,016
Synced today	52	Success	98%

F04 Chart of accounts delta

NetSuite → Coupa — Success

Last run	7 h ago	Runs (7d)	7
Synced today	1,284	Success	100%

F05 Expense reports

Coupa → NetSuite — Success

Last run	24 min ago	Runs (7d)	168
Synced today	96	Success	99%

F07 Invoices & credits

Coupa → NetSuite — Error

Last run	6 min ago	Runs (7d)	672
Synced today	1,162	Success	99%

F08a Invoice payments

NetSuite → Coupa — Success

Last run	41 min ago	Runs (7d)	168
Synced today	438	Success	100%

F09 Coupa Pay settlement

Coupa → NetSuite — Success

Last run	41 min ago	Runs (7d)	168
Synced today	14	Success	100%

F11 Budget lines

NetSuite → Coupa — Success

Last run	5 h ago	Runs (7d)	7
Synced today	12	Success	100%

F11B Budget consumption & drift

Coupa → NetSuite — Success

Last run	5 h ago	Runs (7d)	7
Synced today	189	Success	100%

Integration Control Centre: flow register

Every production flow with direction, health and 7-day throughput

Coupa ↔ NetSuite — integration control centre

Celigo · REST · the Jockey Club · last 7 days

Open in Celigo | Refresh

Overview | Flows | Run log | Errors (4)

Flow		Direction	Status	Last run	Runs (7d)	Records OK	Errors	Ignored	Success
F01	Supplier onboarding	Coupa → NetSuite	Success	3 min ago	2,016	64	0	1,932	100%
F01B	Supplier reconciliation	Coupa → NetSuite	Success	3 min ago	2,016	41	0	1,975	100%
F01C	Supplier updates	Coupa → NetSuite	Success	9 min ago	672	23	0	649	100%
F01D	Supplier entity stamp	NetSuite → Coupa	Success	4 min ago	672	18	0	654	100%
F02	Supplier status & activation	NetSuite → Coupa	Success	3 min ago	2,016	36	0	1,980	100%
F03	Remit-to & tax push	NetSuite → Coupa	Success	3 min ago	2,016	29	0	1,987	100%
F03C	Bank, tax & remit-to	Coupa → NetSuite	Success	2 min ago	2,016	52	1	1,963	98.1%
F04	Chart of accounts delta	NetSuite → Coupa	Success	7 h ago	7	1,284	0	0	100%
F04F	Chart of accounts full walk	NetSuite → Coupa	Success	6 d ago	1	159,017	0	0	100%
F04A	Orphan combination sweep	NetSuite → Coupa	Success	6 d ago	1	212	0	0	100%
F04C	Golden-record canary	NetSuite → Coupa	Success	7 h ago	7	7	0	0	100%
F04D	Fast deactivation	NetSuite → Coupa	Success	7 h ago	7	3	0	4	100%
F05	Expense reports	Coupa → NetSuite	Success	24 min ago	168	96	1	71	99%
F06	Expense payments	NetSuite → Coupa	Success	24 min ago	168	71	0	97	100%
F07	Invoices & credits	Coupa → NetSuite	Error	6 min ago	672	1,162	11	0	99.1%
F08a	Invoice payments	NetSuite → Coupa	Success	41 min ago	168	438	0	15	100%
F08b	Payment reversals	NetSuite → Coupa	Success	41 min ago	168	2	0	166	100%
F09	Coupa Pay settlement	Coupa → NetSuite	Success	41 min ago	168	14	0	154	100%
F11	Budget lines	NetSuite → Coupa	Success	5 h ago	7	12	0	177	100%
F11B	Budget consumption & drift	Coupa → NetSuite	Success	5 h ago	7	189	0	0	100%

Ignored records are deliberate skips, such as already-converged suppliers or payments outside the ERP channel.

Integration Control Centre: error triage

Each failure shown with the system, step, record, message and a suggested fix

Coupa ↔ NetSuite — integration control centre

Celigo · REST · the Jockey Club · last 7 days

Open in Celigo | Refresh

4 record-level errors across runs in the last 24h (0.2%) — open the Errors tab to see which step failed

Runs today	Records synced 24h	Success rate 24h	Errors 24h	Stale flows
186	2,148	99.8%	4	0

Overview | Flows | Run log | Errors (4)

Each failed run is broken down by the step that errored, the affected record, the message and a suggested fix. Links open the record in NetSuite or Coupa; “View errors in Celigo” opens the run to retry.

F07 Invoices & credits — Sat 09:15 — 2 errors

NetSuite · Vendor bill · INVALID_FLD_VALUE · **Invoice 10482 · Sample Supplier Ltd**

Invalid Field Value 11 for the following field: subsidiary

Supplier is not shared with this subsidiary — add the vendor–subsidiary relationship in NetSuite, then retry.

Open in NetSuite | Open in Coupa

NetSuite · Vendor bill · INVALID_KEY_OR_REF · **Invoice 10517 · Example Catering Ltd**

Invalid item reference key 824 for subsidiary 3

Item is not available to this subsidiary — assign it in NetSuite, then retry. Coupa-side fixes need a fresh run.

Open in NetSuite | Open in Coupa

View errors in Celigo | Open flow

F03C Bank, tax & remit-to — Sat 08:36 — 1 error

NetSuite · Bank details · DUP_CSTM_RCRD_ENTRY · **SIM 214 · Demo Logistics Ltd**

A record with this name already exists

Bank-detail names must be unique — check for another supplier using the same beneficiary name.

Open in NetSuite | Open in Coupa

View errors in Celigo | Open flow

F05 Expense reports — Sat 07:48 — 1 error

NetSuite · Expense report · INVALID_FLD_VALUE · **Report 3317 · Sample Employee**

Invalid Field Value 2 for the following field: category

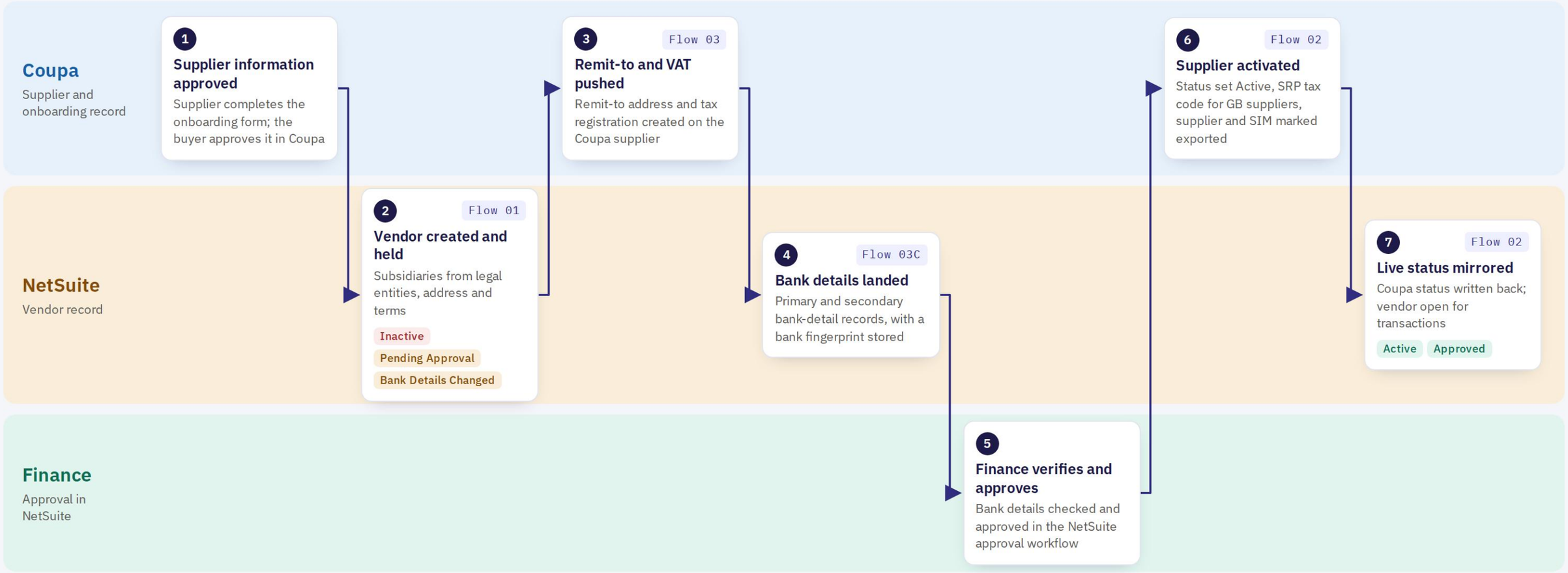
Category belongs to another subsidiary — map this subsidiary’s own expense category, then retry.

Open in NetSuite | Open in Coupa

View errors in Celigo | Open flow

Supplier onboarding, gated by Finance

A new supplier never becomes payable until Finance has verified the bank details in NetSuite. Celigo moves the record between the two systems at each step.



After go-live, changes keep both sides aligned

Flow 03C
Bank details change
A new bank fingerprint re-flags the vendor, sending it back to Finance (step 5)

Flow 01C
Supplier edited in Coupa
Name, email, terms and VAT updated in NetSuite, with an echo-loop guard

Flow 01D
Subsidiary added in NetSuite
Coupa Chart of Accounts and content groups re-stamped to match

Chart of accounts in Coupa, generated from NetSuite

About 159,000 valid GL combinations across nine entities, provisioned as a six-level billing-string hierarchy so users can only code spend to combinations NetSuite accepts.

ENT . 6200 . L03 . ET05 . FIN . 888

Entity	Account	Location	Event type	Department	Event space
NetSuite subsidiary	NetSuite GL account	NetSuite location	NetSuite custom segment	NetSuite department	NetSuite custom segment

How Coupa stores it: one parent-child chain per combination, account first

- `ENT` Sample Entity Ltd — Entity
  - `6200` Event staffing — Account
    - `L03` Main grandstand — Location
      - `ET05` Raceday — Event type
        - `FIN` Finance — Department
          - `888` `ENT.6200.L03.ET05.FIN.888` — Event space, leaf

Why account first. Coupa invoice lines carry the billing string in this order, so the invoice flow parses the same structure on the way back into NetSuite.

Kept in step by the 04 flow family

04F — Full walk
Walks every valid combination, creating missing nodes and re-stamping labels. A full run is 1.28 million API operations, with every error classified.

04 — Daily delta
New and changed GL validation records reach Coupa the same day.

04D — Fast deactivation
Combinations made invalid in NetSuite are switched off in Coupa.

04A — Orphan sweep
Monthly backstop that retires anything no longer valid.

04C — Canary
A golden combination is checked first, so a change to the segment order stops the run.

Source of truth. NetSuite GL validation records, read through a saved search; labels follow NetSuite and Coupa-side renames are overwritten.

Raja Saqib, NetSuite & Celigo integration for The Jockey Club

Illustrative recreation with sample data

Coupa invoices to NetSuite vendor bills

Invoices and credit notes from every entity post to the right subsidiary, vendor and item without a single hard-coded list. The Coupa item master is the crosswalk, maintained by the business as data.

1 Coupa

Approved invoice or credit note

Picked up once approved and not yet exported

→

2 Celigo export hook

Lines normalised

Billing string split, approver, PO, event ID and date, document link

→

3 Celigo lookups

References resolved

Item master, subsidiary by External ID, vendor by Coupa supplier ID

→

4 NetSuite

Vendor bill or vendor credit

Item or expense lines; virtual-card bills posted on payment hold

→

5 NetSuite

Credit applied

Credit notes matched to the open bill and applied automatically

→

6 Coupa

Written back, then locked

NetSuite IDs first, exported flag second, because Coupa locks exported invoices

How each line finds its NetSuite item: first match wins

1	**Coupa catalogue item** The item carries its NetSuite internal ID	Item line
2	**Item number on the line** A numeric NetSuite item ID typed on the invoice line	Item line
3	**Item code in the item master** The code is looked up in Coupa and mapped to its NetSuite item	Item line
4	**No match** Posted as an expense line, coded from the billing string; never an error	Expense line

Built to be re-run safely

Idempotent posting. Bills and credits are matched on the Coupa transaction ID, so a retry updates rather than duplicates.

Subsidiary from the data. The entity segment of the billing string resolves the subsidiary through its External ID.

Virtual-card bills held. Coupa Pay invoices post on payment hold; the settlement flow releases the hold and records the vendor payment against the subsidiary's clearing account.

Backlog cleared. A 777-invoice re-import ran through the same rules, with every residual error traced to a named data fix.

Raja Saqib, NetSuite & Celigo integration for The Jockey Club

Illustrative recreation with sample data

What was built inside NetSuite

The NetSuite side of the integration: SuiteScript 2.1, custom records and fields, all promoted from sandbox to Production through staged SuiteBundler bundles.

SuiteScript 2.1

10 scripts

Integration Control Centre — Suitelet
Run health, flow register and error triage

Integration log writer — RESTlet
Single endpoint for run and error rows

Integration alerter — Scheduled
Flags failing or stale flows on a schedule

Amortisation from event date — User Event
Sets amortisation schedules on bill and credit lines from the event date, whatever the entry path

Remit and tax change detection — User Event, Client Script
Raises the remit-changed flag on genuine value changes only

Expense attendees — User Event, Client Script, Suitelet
Button on the expense report opening an attendee popup

Coupa supplier ID backfill — Map/Reduce
Dry-run by default, CSV audit of every change

Custom records

Coupa payment sync ledger
One row per payment, reversal, settlement and budget line, keyed for idempotency

Supplier event log
Every bank, remit-to and status change, per supplier

Expense attendees
Child records of the expense report, shown on their own subtab

GL deactivation log
Audit of combinations switched off in Coupa

Budget configuration
Entity, period and defaults, so budget scripts hold no hard-coded values

Integration run log and errors
The data behind the Control Centre

Custom fields and searches

Vendor
Coupa supplier and SIM IDs, record links, approval and payment status mirrors, bank fingerprint

Transaction body
Coupa transaction and expense IDs, object type, source system, record link, payment channel, prepaid and synced flags

Transaction line
Coupa line ID, item description, event ID, attendee count

Saved searches
Every Celigo lookup, plus a finance and admin reconciliation suite

House rule. Every integration value lives in a new, purpose-built field. Legacy procurement fields were never reused, so old and new data cannot collide.