



WHITE PAPER

Reconcile Thousands of Transactions in Minutes: Rule-Based Bank Reconciliation in ERPNext

How ERPNext imports a bank statement, proposes matches to your payments and invoices, and uses Bank Transaction Rules to auto-classify the rest.

For finance & accounts leaders · 8 min read

EXECUTIVE SUMMARY

Bank reconciliation is the month-end job everyone underestimates. A busy Indian business runs hundreds or thousands of NEFT, RTGS, UPI and cheque lines through its current accounts, and someone has to tie every one of them back to a payment, an invoice, a bank charge or a transfer. Done by hand in a spreadsheet, it is slow, error-prone and the reason books close late. This paper walks through how ERPNext turns that grind into a mostly-automated flow: import the bank statement once, let the Bank Reconciliation Tool propose ranked matches against your existing vouchers, and set up Bank Transaction Rules so recurring lines — bank charges, interest, a regular vendor — classify themselves. It is grounded in the actual ERPNext doctypes that do the work, and it is honest about where judgement and a clean setup still matter.

The reconciliation grind — and why it eats your month-end

Every payment your business makes or receives shows up twice: once in ERPNext when you book it, and once on the bank statement when the money actually moves. Those two never line up on the same day. A cheque you issued clears a week later. A customer's NEFT lands the morning after you raised the invoice. UPI collections arrive in a steady drip. Bank charges, GST on those charges, interest credits and inter-account transfers appear on the statement with no matching entry in your books at all.

Reconciliation is the work of matching those two views line by line, so the closing balance in ERPNext equals the closing balance on the statement. Done manually — export the statement to Excel, eyeball each row, hunt for the matching voucher, tick it off — it scales badly. A hundred lines is tedious; a thousand across several accounts is a multi-day job that blocks the monthly close and quietly hides errors: a payment booked twice, a receipt never recorded, a charge nobody noticed.

ERPNext's answer is to stop treating this as manual matching and start treating it as a pipeline: get the statement in as structured data, let the system propose the obvious matches, and codify the repetitive decisions as rules so they never reach a human again.

- The books and the bank never agree on the same date — clearing lag is normal, not an error.
- Cheques, NEFT, RTGS, UPI, standing instructions and transfers all clear on their own timelines.
- Bank charges, interest and their GST appear on the statement with no entry in your books yet.
- Manual, spreadsheet-based matching doesn't scale — it's why the month-end close slips.

From bank statement to reconciled books

1

Import

attach the bank's CSV/Excel (or MT940) statement and map its columns; each row becomes a Bank Transaction.

2**Auto-classify**

Bank Transaction Rules match recurring lines by description, amount and type, and set the account and party.

3**Propose matches**

the Bank Reconciliation Tool ranks existing vouchers (payments, invoices, journals) by fields matched.

4**Match or create**

confirm the proposed voucher, or create a new one for charges, interest and one-offs.

5**Clear**

confirming a match sets the voucher's clearance date automatically, driving the bank-vs-book difference toward zero.

Step one: get the statement into ERPNext

Nothing can be matched until the bank's transactions exist inside ERPNext as data. The Bank Statement Import tool handles this. You download the statement from your bank's portal — as CSV or Excel — attach it, and tell ERPNext which column in the file maps to which field on the Bank Transaction: date, deposit, withdrawal, description, reference number, currency. That mapping step is the whole trick, because no two Indian banks — ICICI, HDFC, SBI, Axis, Kotak — export their statements in the same layout.

Crucially, you only map each bank once. ERPNext stores the column mapping against the Bank record, so the next month's upload from the same bank is picked up automatically; you can still adjust it if the format changes. For banks that support it, ERPNext also reads the MT940 format, and it can pull from a Google Sheet. The import runs as a background job and creates one Bank Transaction per row, each carrying its date, amount as a deposit or withdrawal, description, reference number and the bank's own party-name fields where the statement provides them.

The outcome of this step is simple but important: your bank statement is now a set of structured, queryable records inside the system — the raw material everything downstream works on.

- Bank Statement Import — attach a CSV/Excel export and map its columns to Bank Transaction fields.
- The column mapping is saved per Bank, so future imports from the same bank need no re-mapping.
- Supports MT940 bank files and import from a Google Sheet, with custom delimiters if needed.
- Each statement row becomes a Bank Transaction: date, deposit/withdrawal, description, reference number.
- Optional bank-feed synchronisation (via Plaid) can import transactions automatically instead of a file.

Bank Reconciliation Tool

[Upload Bank Statement](#) [Save](#)

Company <input type="text" value="moha"/>	Bank Statement From Date <input type="text" value="01-09-2020"/>	Account Opening Balance <input type="text" value="₹ 0.00"/>
Bank Account <input type="text" value="ICICI Moha - ICICI"/>	Bank Statement To Date <input type="text" value="31-03-2021"/>	Bank Statement Closing Balance <input type="text" value="23,543.00"/>

RECONCILE

Closing Balance as per Bank Statement ₹ 23,543.00	Closing Balance as per ERP ₹ 543.00	Difference ₹ 23,000.00
---	---	----------------------------------

	Date	Party ...	Party	Description	Deposit	Withdrawal	Unallocate...	Reference Number	Actions
1	2020-10-04			Travelling Expenses	₹ 0.00	₹ 146.00	₹ 146.00	738456287104774	Actions
2	2020-10-03			T-Mobiles Payment	₹ 0.00	₹ 15,000.00	₹ 15,000.00	756837465783456	Actions
3	2020-10-02			M-Services	₹ 20,000.00	₹ 0.00	₹ 20,000.00	283756348568493	Actions
4	2020-10-01			Interest	₹ 15,000.00	₹ 0.00	₹ 15,000.00	753643865899226	Actions

The Bank Reconciliation Tool: imported statement lines, the live bank-vs-book difference, and per-transaction Actions.

Step two: let ERPNext propose the matches

This is where the time is won. Open the Bank Reconciliation Tool, pick your company and bank account, set the statement's from/to dates, and enter its closing balance. ERPNext lays out every imported Bank Transaction that hasn't been cleared yet and shows the running difference between the closing balance as per the bank and as per your books. The whole goal of the exercise is visible in one number: drive that difference to zero.

For each transaction you click Actions and choose Match Against Voucher. ERPNext searches your existing documents — Payment Entries, Journal Entries, Sales Invoice and Purchase Invoice payments, Expense Claims — and presents the candidates ranked by how many fields line up: amount, reference number and party. The best match, where reference number and amount and party all agree, floats to the top. A 'Show Only Exact Amount' filter narrows the list to vouchers matching the transaction's value exactly, and you can tick one voucher or several (a single bank credit that settles multiple invoices) against the same line. Confirming the match sets the voucher's clearance date automatically — the very thing manual reconciliation exists to do.

For the cleanest cases, ERPNext can also auto-reconcile: where a transaction's reference number and amount match a single voucher unambiguously, it clears them for you in bulk, leaving only the genuine exceptions for a human to look at.

- The tool shows unreconciled transactions and the live difference between bank and book balances.
- Match Against Voucher ranks candidates by fields matched — amount, reference number and party.
- Match works against Payment Entries, Journal Entries, invoice payments and Expense Claims.
- Match one or several vouchers to a single transaction; confirming sets the clearance date automatically.
- Unambiguous reference-plus-amount matches can be auto-reconciled in bulk, leaving only exceptions.

Reconcile the Bank Transaction
Close Submit

Action

Match against Voucher
▼

FILTERS

Payment Entry

Sales Invoice

Expense Claim

Journal Entry

Purchase Invoice

Show Only Exact Amount

SELECT VOUCHERS TO MATCH

<input type="checkbox"/>		Document Type	Document Name	Reference Date	Amount	Party	Reference Number
<input type="checkbox"/>							
<input type="checkbox"/>	1	Payment Entry	ACC-PAY-2021-01591	2021-01-11	₹ 146.00	assa	12763
<input type="checkbox"/>	2	Payment Entry	ACC-PAY-2021-01592	2021-01-11	₹ 146.00	usd supp	12763
<input type="checkbox"/>	3	Payment Entry	ACC-PAY-2021-01593	2021-01-11	₹ 146.00	assa	738456287104774

TRANSACTION DETAILS ▼

Match Against Voucher proposes candidate vouchers, filterable to exact amounts and ticked one or many at a time.

Step three: Bank Transaction Rules — teach ERPNext to classify the repetitive lines

Matching to an existing voucher works when a voucher exists. But a large share of statement lines are recurring items that never had a voucher: the monthly bank charge, the quarterly interest credit, a regular payout to the same vendor, a sweep between your own accounts. Handling those by hand every month is exactly the kind of repetitive decision software should absorb — and that is what a Bank Transaction Rule does.

A rule is a saved instruction that matches transactions on their description, amount band and type, then classifies them automatically. You give it a priority (so rules are applied in a defined order), a transaction type (Any, Withdrawal or Deposit), an optional min and max amount, and one or more description conditions — the description Contains, Starts With, Ends With, or matches a Regex on a string like the bank's narration. When a transaction fits, the rule's Action fires: classify it as a Bank Entry, a Payment Entry or an internal Transfer, and pre-set the account, the party (type and name) and, where relevant, a split across multiple accounts.

The effect compounds. Once a rule exists for 'BANK CHARGES' or a recurring vendor's narration, every future statement carrying that pattern is classified without anyone touching it — ERPNext even records on each Bank Transaction that a rule was evaluated and which rule matched. Set up thoughtfully, rules are what let a thousand-line statement reconcile in minutes instead of days, because the long tail of predictable lines simply handles itself.

- A Bank Transaction Rule matches on description conditions (Contains / Starts With / Ends With / Regex), amount band and type.
- Priority decides which rule wins when several could apply, so behaviour is predictable.
- The rule's action classifies the line as a Bank Entry, Payment Entry or Transfer — and can pre-set account and party.
- One rule per recurring pattern — bank charges, interest, a regular vendor, own-account sweeps — handles it every month thereafter.
- Each Bank Transaction records that a rule was evaluated and which rule matched, so the automation is auditable.

Anatomy of a Bank Transaction Rule

1

Match conditions

description (Contains / Starts With / Ends With / Regex), a min–max amount band, and transaction type (Any / Withdrawal / Deposit).

2

Priority

decides which rule wins when more than one could apply.

3

Action

classify the line as a Bank Entry, Payment Entry or Transfer.

4

Pre-fill

set the account and the party (type and name), or split across multiple accounts.

5

Audit

each matched Bank Transaction records that a rule was evaluated and which rule matched.

Handling the exceptions the machine leaves behind

Automation clears the easy 80–90%; the value of a good tool is how gracefully it hands you the rest. ERPNext gives you three moves for a transaction that neither matched a voucher nor fell under a rule.

First, Create Voucher: for a line with no entry in your books — a bank charge, an interest credit, a one-off receipt — you create the Payment Entry, Journal Entry or Bank Entry right there, fill in the details the statement didn't carry, and submitting it books the entry and sets the clearance date in one step.

Second, Update Bank Transaction: sometimes a match fails only because the transaction is missing a reference number or a party; editing the transaction to add that detail lets ERPNext find the match it couldn't before. Third, party matching: ERPNext can optionally auto-assign the customer, supplier or employee on a transaction by reading the bank's party-name fields — with an opt-in fuzzy match for near-matches — provided you've stored bank details against those parties. Together these turn the exception queue from a dead end into a short, deliberate list.

- Create Voucher — book a Payment Entry, Journal Entry or Bank Entry for lines with no existing record (charges, interest, one-offs).
- Update Bank Transaction — add a missing reference number or party so ERPNext can find a match it missed.
- Automatic party matching — optionally set the customer/supplier/employee from the bank's party-name fields (with optional fuzzy matching).
- The goal throughout is the same: drive the bank-vs-book difference to zero, one cleared line at a time.

What good reconciliation discipline looks like

The tooling is strong, but a clean, fast reconciliation is as much about setup and habit as features. A few disciplines separate teams that close in an afternoon from teams that dread month-end.

Get the column mapping right for each bank once, and reconcile little and often — importing weekly rather than saving a thousand lines for the last day makes the difference approach zero in manageable steps. Build your rule library deliberately: every time an exception turns out to be recurring, ask whether a Bank Transaction Rule should catch it next time, and use amount bands and priorities so rules stay precise rather than greedy. Store bank account details against your regular customers and suppliers so party matching has something to work with. And treat a non-zero difference as a signal to investigate, not a number to force — a stubborn gap is usually a double-booking, a missed entry or a transfer recorded on one side only, and reconciliation is exactly the control that surfaces it.

- Map each bank's statement format once; reconcile weekly, not in one month-end marathon.
- Grow a rule library — promote every recurring exception into a Bank Transaction Rule, scoped by amount and priority.
- Register bank accounts against key parties so automatic party matching actually fires.
- Never force the difference to zero — a residual gap is a control catching a real error worth finding.

When to get help

The Bank Reconciliation Tool is usable out of the box, and many teams run it themselves. Where an experienced partner earns their fee is in the setup that makes automation pay: getting each bank's import mapping right, designing a rule library that matches how your narrations actually read, wiring party matching to clean masters, and deciding how bank charges, GST on those charges, and inter-account transfers should be classified so the accounting is correct as well as fast. Those choices, made once at implementation, are the difference between a tool that clears 90% of lines untouched and one that still needs babysitting every month.

As an official ERPNext partner working with Indian businesses, we set up bank statement imports, reconciliation and Bank Transaction Rules mapped to your banks and your transaction patterns — so month-end reconciliation stops being the task that holds up your close and becomes something that mostly runs itself.

KEY TAKEAWAYS

- 1 Reconciliation is a pipeline, not manual matching: import the statement, auto-classify with rules, let the tool propose matches, clear the rest.
- 2 Bank Statement Import maps each bank's CSV/Excel columns to Bank Transaction fields once — future imports from that bank are automatic.
- 3 The Bank Reconciliation Tool ranks candidate vouchers (payments, invoices, journals, expense claims) by fields matched and can auto-reconcile unambiguous ones.
- 4 Bank Transaction Rules classify recurring lines — bank charges, interest, a regular vendor, transfers — by description, amount and type, so they never reach a human again.
- 5 Exceptions are handled in place — create a voucher, update the transaction, or auto-assign the party — always driving the bank-vs-book difference to zero.

FAQ

Can ERPNext import statements from Indian banks like ICICI, HDFC or SBI?

Yes. The Bank Statement Import tool takes a CSV or Excel statement exported from any bank's portal and lets you map its columns — date, deposit, withdrawal, description, reference number — to ERPNext's Bank Transaction fields. Because no two Indian banks export the same layout, this mapping matters, but you only do it once per bank: ERPNext saves the mapping and reuses it for every future upload from that bank. It can also read MT940 files and import from a Google Sheet.

How does ERPNext decide which payment or invoice a bank transaction matches?

In the Bank Reconciliation Tool you choose Match Against Voucher, and ERPNext searches your existing documents — Payment Entries, Journal Entries, Sales and Purchase Invoice payments, and Expense Claims — then ranks the candidates by how many fields agree: amount, reference number and party. The strongest match, where all three line up, appears at the top. You can filter to exact-amount matches and tick one voucher or several against a single transaction; confirming the match sets the clearance date automatically.

What are Bank Transaction Rules and when should we use them?

A Bank Transaction Rule auto-classifies recurring statement lines that don't have a matching voucher — bank charges, interest credits, a regular vendor payout, or a sweep between your own accounts. The rule matches on the description (Contains, Starts With, Ends With or Regex), an amount band and the transaction type, then classifies the line as a Bank Entry, Payment Entry or Transfer and pre-sets the account and party. Once a rule exists for a pattern, every future statement carrying it is classified without manual work — this is what lets very large statements reconcile in minutes.

How do we handle bank charges, interest or GST that have no invoice in ERPNext?

For any statement line with no existing entry in your books, use Create Voucher in the reconciliation tool to book a Payment Entry, Journal Entry or Bank Entry directly against it — filling in details the statement didn't carry, such as the charge and its GST — and submitting both records the entry and sets the clearance date. Better still, once you've seen a recurring charge or interest line, capture it as a Bank Transaction Rule so future occurrences classify themselves automatically.

Talk to a real ERPNext expert.

Call or WhatsApp +91 62358 66111 · info@acube.co · acubeinnovations.com

