WHITE PAPER

From Downtime to OEE: Measuring True Shop-Floor Performance in ERPNext

How to turn Job Cards, workstations and downtime reasons into a real Overall Equipment Effectiveness number — and use it to find the hours you're losing.

For plant & operations managers · **9 min read**

Acube Innovations · Official ERPNext Partner · acubeinnovations.com · +91 62358 66111

EXECUTIVE SUMMARY

Most plants know they're losing time on the shop floor — they just can't say where, how much, or why in a way anyone can act on. Overall Equipment Effectiveness (OEE) is the discipline that answers those questions, and it only works when it's built on real, captured data rather than end-of-shift memory. This paper is a practical, honest look at how ERPNext records the raw shop-floor signals — operation-level time logs on Job Cards, machine downtime with reasons, and produced-versus-rejected quantities — and how you turn those signals into a defensible OEE number. It is deliberately clear about what ERPNext captures out of the box and what you configure with a report or dashboard, so you go in with the right expectations.

Why gut-feel shop-floor metrics fail

Ask three people on a plant floor how the line ran last week and you'll get three answers, all sincere, none measurable. "It was a good week." "The old press kept tripping." "We hit the numbers, mostly." None of that survives contact with a customer escalation or a capacity decision, because it isn't data — it's memory, and memory quietly rounds off the losses.

The losses that matter most are the ones nobody logs: the twelve minutes waiting for a changeover, the machine that idled while the operator chased a missing component, the batch that ran slow because a tool was worn. Individually they feel trivial. Added across shifts, machines and months, they are usually the single largest — and cheapest — source of capacity a plant has. You can't reclaim what you never measured, and gut feel systematically under-counts exactly these small, frequent stops.

The fix is not a wall of dashboards. It's one honest number, built from data captured as work happens, that tells you how much of your available machine time turned into good product. That number is OEE.

What OEE actually is — in plain terms

Overall Equipment Effectiveness is a single percentage that answers one question: of the time a machine was supposed to be making good parts, how much of it actually did? It is the product of three honest factors, each capping the next:

- Availability — did the machine run when it was scheduled to? It measures lost time: breakdowns, changeovers, waiting for material or an operator. A machine scheduled for eight hours that ran for six has 75% availability.
- Performance — when it ran, did it run at its rated speed? It measures slow cycles and minor stops — the machine is on, but producing below its ideal rate.
- Quality — of what it produced, how much was good the first time? It measures scrap and rework. Parts that need reworking cost the time twice.
- OEE — Availability × Performance × Quality. Because the three multiply, a weak factor drags the whole number down: 90% × 90% × 90% is only about 73%, which is why OEE is harder — and more honest — than any single metric.

The three factors of OEE

1 **Availability**
the share of scheduled time the machine actually ran. Lost to breakdowns, changeovers and waiting. In ERPNext: planned operation time vs actual run time and Downtime Entries.

2 **Performance**
how close the running machine got to its rated speed. Lost to slow cycles and minor stops. In ERPNext: actual run time from Job Card logs vs the operation's expected time for the quantity made.

3 **Quality**
the share of output that was good first time. Lost to scrap and rework. In ERPNext: total completed qty vs process-loss qty and Quality Inspection results.

4 **OEE**
the three multiplied. Because they compound, one weak factor caps the whole number — which is why OEE is a truer read than any single metric.

How ERPNext captures the signals

OEE is only as trustworthy as the data underneath it, and this is where ERPNext's manufacturing module earns its place: it records the raw signals at the point of work, not after the fact. Three doctypes do most of the job.

The **Job Card** is the shop-floor record of a single operation on a Work Order — tied to a specific Operation and Workstation, with a target quantity to manufacture. As the operator works, each run is captured in a **Time Log** (a child table on the Job Card) with a precise From Time, To Time, the minutes elapsed, the employee, and the quantity completed in that stretch. A Job Card can hold many time logs, so a stop-and-restart is recorded as separate intervals rather than smoothed into one — which is exactly the granularity Availability and Performance need.

The **Work Order Operation** stores the *plan* to compare against: planned start and end times and expected minutes per operation, alongside the actuals ERPNext fills in — actual start, actual end and actual operation time. That planned-versus-actual pair is the backbone of the Availability factor. Quantities live here too: completed quantity and process-loss quantity, which feed Quality.

- Job Card -> Time Logs: from/to time, minutes, employee and completed qty per interval — the actual run time.
- Work Order Operation: planned vs actual start/end and time in minutes — the schedule to measure against.
- Workstation: the machine itself, with an hourly rate, working-hours calendar, holiday list and job capacity — and a live status (Production, Idle, Setup, Maintenance, Problem, Off).
- Produced vs lost quantity: total completed qty and process-loss qty per operation, with Quality Inspection available to formalise pass/fail.

Downtime, with a reason attached

Knowing a machine stopped is half the story; knowing *why* is what drives improvement. ERPNext has a dedicated **Downtime Entry** doctype for exactly this. Each entry records the Workstation, a From Time and To Time (and therefore the downtime duration), the operator involved, free-text remarks, and — critically — a **Stop Reason** chosen from a defined list.

Out of the box that list covers the usual suspects: excessive machine set-up time, unplanned machine maintenance, on-machine press checks, machine operator errors, machine malfunction, electricity down, and a catch-all "Other". Because Stop Reason is a structured field rather than a note, every hour of downtime is categorised the moment it's logged — which means you can later add it up *by reason* instead of re-reading a hundred remarks. That single design choice is what makes root-cause analysis possible later on.

The standard reason list reflects general manufacturing, not your plant. Reasons are configuration, not code: using ERPNext's Customize Form, you can tailor the Stop Reason options to your own loss vocabulary — "awaiting QC", "no material", "tool change", "power cut / DG switchover" — so the categories match how your team actually talks about lost time.

Turning signals into OEE — built-in vs configured

Here is the honest part, and the part most vendors gloss over. ERPNext **captures** every raw signal OEE needs — planned time, actual run time, downtime with reasons, produced quantity and losses — natively, as a by-product of running Work Orders and Job Cards. What it does **not** ship is a single, pre-built "OEE = 78%" gauge in core. There is no standard report that multiplies the three factors and hands you the percentage; that roll-up is something you configure.

That's a smaller gap than it sounds, because the inputs are all present and queryable. The Availability numerator and denominator come from Work Order Operation (planned time) and Job Card time logs plus Downtime Entries (actual run time and stoppages). Performance compares the actual run time against the operation's expected/ideal time for the quantity produced. Quality compares total completed qty against process-loss (and any Quality Inspection rejections). Each factor is arithmetic on fields ERPNext already stores.

- Built in: Job Cards, time logs, Downtime Entries with reasons, Workstation status and capacity, planned-vs-actual operation times, and produced/loss quantities.
- Configured: a Query Report or Report Builder view that computes Availability, Performance and Quality per workstation or per shift.
- Configured: a dashboard or Insights view that trends OEE over time and lets you filter by machine, operation or reason.
- Optional: a live Plant Floor view of workstation status, and Manufacturing Settings such as "enforce time logs" to make capture non-optional.

Using downtime reasons to actually improve

Once Stop Reasons are being logged consistently, the most valuable output is almost embarrassingly simple: a Pareto of downtime by reason. Sort your Stop Reasons by total lost hours over a month and the picture is usually stark — a small number of reasons account for most of the loss. It is rarely a

mystery machine; it is far more often changeover time, waiting on material, or one recurring malfunction.

That ranking is what turns a number into a decision. If "excessive set-up time" tops the list, the intervention is a changeover (SMED) project, not a new machine. If "electricity down / DG switchover" is eating hours, that's an infrastructure conversation with a rupee value now attached to it. Because every Downtime Entry carries a workstation and a timestamp, you can slice the same data by machine and by shift to see whether a loss is a plant-wide pattern or one line, one team, one time of day.

The discipline is to review it on a cadence — a weekly or monthly look at the top three reasons — and to close the loop by watching the same Pareto after you act. Improvement you can't see in the next month's chart didn't happen.

Rollout realities — and getting it right

The technology is the easy part; the data habit is the project. OEE lives or dies on capture discipline, so a realistic rollout is as much about the shop floor as the software. Start narrow — one line or one bottleneck machine — and get clean, complete Job Card time logs and Downtime Entries there before scaling. A partial signal produces a misleading OEE, which is worse than none because people stop trusting it.

Expect to invest in a few things beyond configuration: tailoring the Stop Reason list to your real losses; deciding operating calendars and capacity on each Workstation so "planned time" means something; and making capture frictionless for operators — a shop-floor tablet or barcode beats a back-office keyboard. ERPNext's Manufacturing Settings can enforce time logs so a Job Card can't be closed without them, which turns good intentions into a rule.

Done well, the payoff is compounding: a trustworthy OEE number, a ranked list of what's costing you hours, and the before/after evidence to prove an improvement worked. That's the difference between a plant that feels busy and one that can prove where its capacity went — and reclaim it.

KEY TAKEAWAYS

1. OEE = Availability × Performance × Quality — one honest number that shows how much scheduled machine time became good product.
2. ERPNext captures the raw signals natively: operation-level time logs on Job Cards, planned-vs-actual operation times, and produced-vs-loss quantities.
3. Downtime Entries record every stoppage with a structured Stop Reason — so lost time is categorised as it happens, ready for a Pareto later.
4. The OEE roll-up itself is configured, not shipped: a Query Report or dashboard computes the three factors from fields ERPNext already stores.
5. Rollout succeeds on capture discipline — start with one machine, tailor the reasons to your losses, and make logging frictionless for operators.

FAQ

Does ERPNext calculate OEE out of the box?
ERPNext captures every input OEE needs — Job Card time logs, Downtime Entries with reasons, planned-vs-actual operation times and produced-vs-loss quantities — but core does not ship a single pre-built OEE gauge. The Availability, Performance and Quality roll-up is configured with a Query Report, Report Builder view or dashboard on top of data ERPNext already stores.

How does ERPNext track machine downtime?
Through the Downtime Entry doctype. Each entry logs the workstation, a from/to time (so the duration is known), the operator, free-text remarks and a structured Stop Reason. Because the reason is a defined field rather than a note, you can total downtime by reason across a month and see exactly where your hours are going.

Can we change the downtime stop reasons to match our plant?
Yes. The standard Stop Reason list (set-up time, unplanned maintenance, machine malfunction, operator errors, electricity down, and so on) is a starting point. Using Customize Form you can replace it with your own loss vocabulary — for example "awaiting QC", "no material", "tool change" or "power cut / DG switchover" — so the categories match how your team describes lost time. That's configuration, not custom code.

We're a small Indian manufacturer — is OEE tracking in ERPNext worth it for us?
Often yes, because the biggest capacity gains for smaller plants come from small, frequent stops that nobody currently logs — waiting on material, changeovers, an ageing machine that trips. ERPNext lets you start with a single bottleneck machine, capture clean data there, and prove where the hours go before scaling. You reclaim capacity you already own rather than buying new equipment. Acube can help scope a right-sized rollout.

Talk to a real ERPNext expert.
Call or WhatsApp +91 62358 66111 · info@acube.co · acubeinnovations.com