WHITE PAPER

Support That Meets Its SLAs: Response & Resolution Tracking in ERPNext

How ERPNext turns a service promise into a tracked contract — response and resolution targets per priority, support hours, pause rules, and a Fulfilled-or-Failed status on every ticket.

For support & customer-success leaders · **8 min read**

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

EXECUTIVE SUMMARY

Every support team makes promises — "we'll get back to you today," "we'll have it fixed by tomorrow." The trouble is that a promise nobody measures is a promise nobody keeps. This paper explains how ERPNext turns those promises into a tracked Service Level Agreement: response and resolution targets set per priority, the support hours and holidays that decide when the clock runs, the statuses that pause it while you wait on the customer, and the moment it marks a ticket Fulfilled or Failed. It is grounded in the actual ERPNext doctypes — Service Level Agreement, Service Level Priority, Service Day, and the SLA fields that live on every Issue — so you can see exactly what the software tracks for you, where escalation and reporting fit, and where an experienced partner makes the difference between an SLA that looks good on paper and one your customers actually feel.

Why support without SLAs quietly erodes trust

Most support teams don't lose customers to a single catastrophic failure. They lose them to a slow drip of unanswered tickets, replies that come "eventually," and no shared definition of what "eventually" was supposed to mean. When there's no agreed target, every ticket is judged by feel — the customer feels ignored, the agent feels busy, and both are right. There's nothing to point at.

The damage is invisible until it isn't. A high-value client raises a critical issue on a Friday; it sits behind fifty routine queries because the queue has no notion of priority. A first response goes out inside an hour on a quiet day and three days later on a busy one, and nobody can say whether that's normal because "normal" was never written down. When a renewal conversation turns tense, the account team has no record of how support actually performed — only anecdotes. A Service Level Agreement fixes the root cause: it makes the promise explicit, measurable and visible to everyone, so support stops being a matter of opinion and becomes a matter of record.

- No shared target means every ticket is judged by feel — and feel is where trust erodes.
- Without priority, a critical outage waits behind routine questions in the same undifferentiated queue.
- "We usually respond quickly" is unprovable at renewal time; a tracked SLA is evidence.
- The fix isn't working harder — it's making the promise explicit, measurable and visible.

What a Service Level Agreement actually is

An SLA is a contract about outcomes, not effort. It says: for a request of a given priority, we will make first contact within this long, and resolve it within that long — measured against agreed working hours, and paused when the delay is on your side, not ours. It deliberately says nothing about how the work gets done; it commits only to the timelines.

Two numbers do most of the work. The response target (ERPNext calls it First Response Time) is the promise that a human has acknowledged and engaged — the single biggest driver of perceived support quality, because a fast "we're on it" buys enormous goodwill. The resolution target (Resolution Time) is the promise that the problem is actually fixed. Separating them matters: a team can be excellent at responding and poor at resolving, or the reverse, and you can only manage what you

measure separately. ERPNext models exactly this split, and lets both targets vary by how urgent the request is.

- Response target (First Response Time) — how fast a human acknowledges and engages. Perceived quality lives here.
- Resolution target (Resolution Time) — how fast the problem is actually fixed.
- Both are measured against agreed support hours, not raw wall-clock time.
- An SLA commits to timelines and outcomes — never to the method of delivery.

How ERPNext turns the promise into a working system

In ERPNext, a Service Level Agreement is a real document you configure, not a wiki page. You choose which doctype it applies to (support tickets live in the Issue doctype), give the agreement a name, and set the window it's valid for. The heart of it is the Priorities table: one row per priority — typically Low, Medium, High and Urgent, each a record in the Issue Priority list — and on each row you set that priority's First Response Time and Resolution Time. Urgent gets tight targets; Low gets generous ones. One priority is marked the default, applied when a ticket doesn't specify.

Crucially, an SLA doesn't have to apply to everyone equally. You can make one the organisation-wide default, or scope an agreement to a specific Customer, Customer Group or Territory — so a premium-support client can carry tighter targets than your standard tier, all in the same system. A finer-grained condition (a simple expression, for example matching a particular issue type) can decide when the agreement applies. And there's a master switch in Support Settings — Track Service Level Agreement — that turns the whole mechanism on. The result is that the same promise you'd otherwise keep in a sales contract becomes an enforced, measured rule inside the tool your agents already work in.

- Apply On — the SLA attaches to a doctype; support tickets are the Issue doctype.
- Priorities table — per priority (Low / Medium / High / Urgent), set First Response Time and Resolution Time.
- Default priority — used when a ticket doesn't specify one; tighten or relax each independently.
- Scope by Customer, Customer Group or Territory — premium tiers carry tighter targets in the same system.
- "Track Service Level Agreement" in Support Settings switches the whole mechanism on.

What makes up an ERPNext Support SLA

1. **Apply On + validity**
   the SLA governs the Issue doctype and applies over a start/end window.

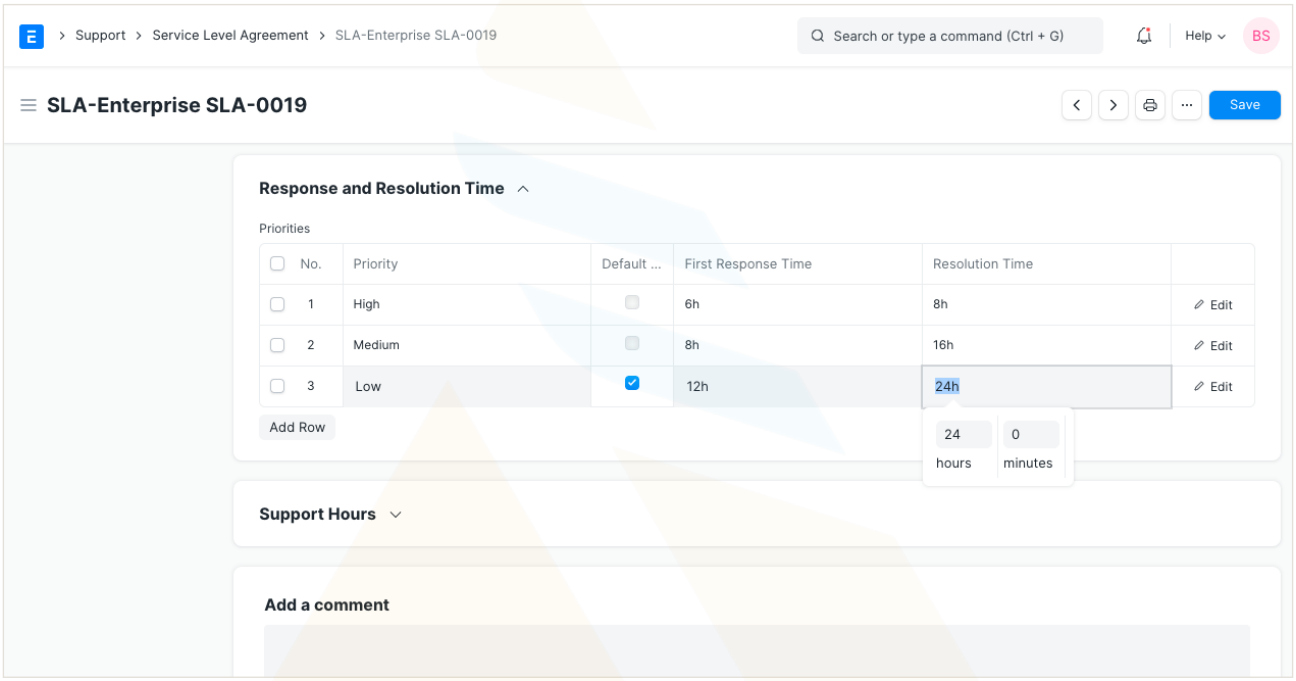
2. **Priorities table**
   one row per priority (Low / Medium / High / Urgent), each with its own First Response Time and Resolution Time; one marked default.

3 **Scope**
organisation-wide default, or targeted to a Customer, Customer Group or Territory (with an optional condition).

4 **Working Hours**
a Monday-to-Sunday table of start/end times, so the SLA clock only runs when support is staffed.

5 **Holiday List**
public/regional holidays that don't count against your targets.

6 **Pause & Fulfilled statuses**
'SLA Paused On' (e.g. On Hold, Replied) freezes the clock; 'SLA Fulfilled On' (e.g. Resolved, Closed) marks the target met.



The Priorities table inside an ERPNext SLA — each priority (High, Medium, Low) carries its own First Response Time and Resolution Time.

When the clock runs: support hours, holidays and pause conditions

A target of "resolve within eight hours" is meaningless until you define eight hours of what. This is where most home-grown SLA spreadsheets fall apart, and where ERPNext is genuinely careful. Each agreement carries a Working Hours table — a row per weekday (Monday to Sunday) with a start and end time — so the SLA clock only ticks during the hours you actually staff support. A ticket raised at 9 p.m. doesn't quietly burn its response target overnight; the clock resumes when the working day does. Each agreement is also linked to a Holiday List, so national and regional holidays don't count against you either.

Just as important is pausing the clock when the ball is in the customer's court. The agreement's "SLA

Paused On" table lets you nominate statuses — commonly On Hold or Replied — during which the timer stops. If an agent has asked the customer for a screenshot or a login and the ticket sits waiting, that waiting time shouldn't count as your delay. ERPNext tracks exactly this: it records when a ticket went on hold and accumulates the Total Hold Time, then measures your resolution performance net of it. There's a matching "SLA Fulfilled On" table — the statuses (such as Resolved or Closed) that mark the promise met. Together, working hours, holidays and pause rules mean the SLA measures your responsiveness, not the calendar's.

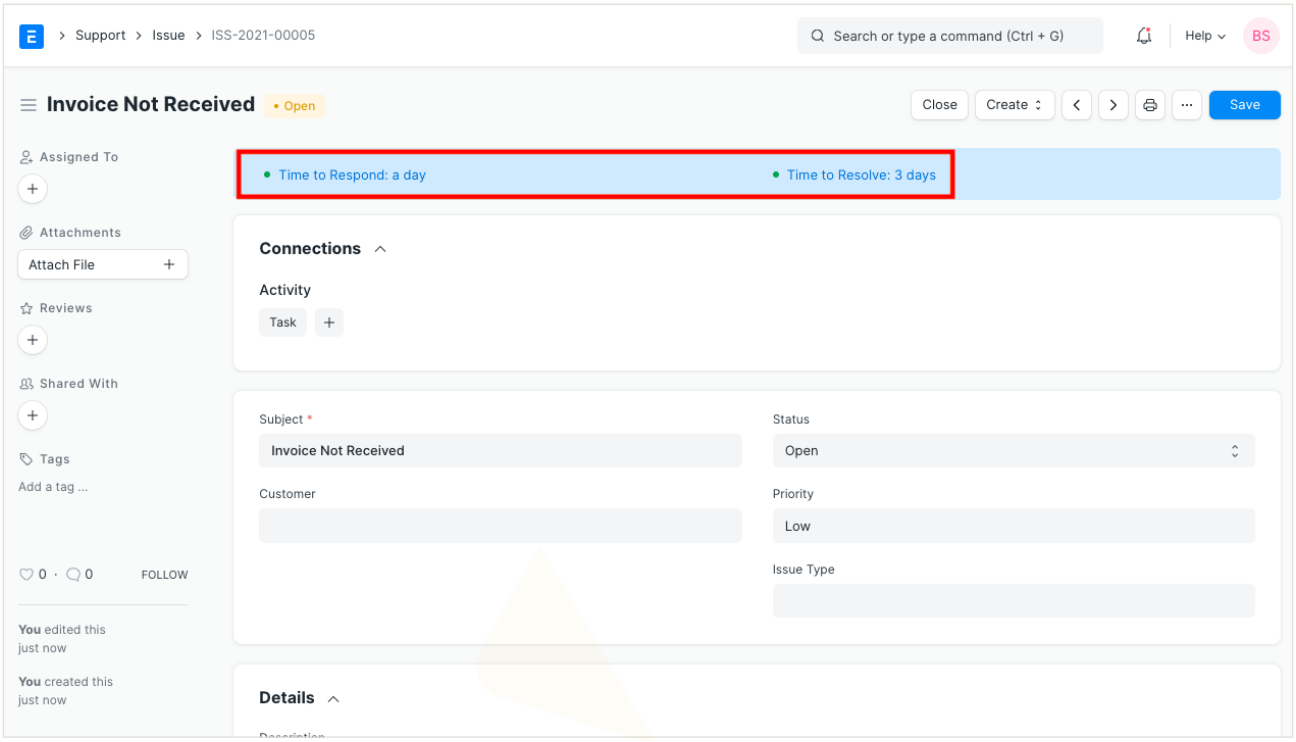
- Working Hours — a Monday-to-Sunday table of start/end times; the clock only runs when you staff support.
- Holiday List — linked to the agreement, so public holidays don't count against your targets.
- SLA Paused On — statuses like On Hold or Replied freeze the timer while you wait on the customer.
- Total Hold Time is tracked and netted out, so waiting-on-customer time isn't counted as your delay.
- SLA Fulfilled On — the statuses (e.g. Resolved, Closed) that mark the target met.

On the ticket: deadlines, status and escalation

Once an SLA is live, every Issue it governs carries the promise on its face. ERPNext stamps two deadlines onto the ticket — Response By and Resolution By — computed from the ticket's priority and the working-hours calendar, so the agent sees the actual clock time they're working against, not an abstract "within 4 hours." A ticket opens on the ERPNext desk showing "Time to Respond" and "Time to Resolve" indicators; the countdown is right there, not buried in a report.

The ticket also carries a live Service Level Agreement Status that moves through a clear lifecycle: First Response Due while you owe the customer a reply, Resolution Due once you've responded but not yet fixed it, then Fulfilled if you met the targets or Failed if a deadline passed. ERPNext records the supporting facts alongside — First Responded On, the measured First Response Time, and Resolution Time — and, because it knows Total Hold Time, it also keeps a User Resolution Time that strips out the waiting. Escalation is built on top of these fields rather than being a separate black box: because Response By and Resolution By are real datetimes and the status is a real field, you use Frappe's notifications and assignment rules to nudge, reassign or alert a manager as a deadline approaches or a status turns to Failed. If circumstances change — the customer reclassifies the severity, say — a Reset Service Level Agreement action re-applies the correct targets to the ticket.

- Response By and Resolution By — concrete deadlines stamped on each ticket from its priority + working hours.
- Service Level Agreement Status — First Response Due -> Resolution Due -> Fulfilled / Failed.
- Captured facts — First Responded On, First Response Time, Resolution Time, and User Resolution Time (net of hold).
- Escalation is built from these fields via Frappe notifications and assignment rules — alert or reassign before a breach.
- Reset Service Level Agreement re-applies the right targets when a ticket's priority or scope changes.



An Issue governed by an SLA — the 'Time to Respond' and 'Time to Resolve' countdowns sit right on the ticket the agent works.

Measuring adherence — from single tickets to a support-quality picture

Because every ticket carries its own SLA status and timing fields, adherence stops being a monthly guessing exercise and becomes something you can read at a glance. The immediate view is operational: filter the Issue list to everything still First Response Due or Resolution Due to see what needs attention right now, and watch the Failed count as your early-warning signal. Nothing has to be reconstructed after the fact — the state is live.

Over time, the same fields roll up into the picture leaders actually care about: what share of tickets met their first-response target, what share were resolved in time, and how that trends by priority, by customer and by team. Because response and resolution are tracked separately, you can diagnose honestly — a team that responds fast but resolves slowly has a very different problem from one that's slow out of the gate. The discipline this rewards is unglamorous but decisive: set targets you can actually meet (an SLA you routinely fail is worse than none), keep priorities honest so "Urgent" still means urgent, and review the Failed tickets as a learning loop, not a blame exercise. Measured this way, the SLA becomes a management instrument, not just a promise.

- Live operational view — filter Issues by SLA status to see what's due now and what's already Failed.
- Response and resolution measured separately, so you can tell a slow-to-answer team from a slow-to-fix one.
- Trends by priority, customer and team turn ticket-level data into a support-quality picture.
- Set targets you can meet, keep priorities honest, and treat Failed tickets as a learning loop.

Getting SLA support right — and where a partner helps

The mechanism is powerful, but a good SLA setup is a design decision, not a checkbox. The judgement calls are where it succeeds or fails: how many priorities you really need and what each one honestly means; targets that are ambitious enough to matter but realistic enough to hit; support hours and a Holiday List that reflect how your team actually works (including India's mix of national and state holidays); and pause rules that are fair to both sides so "waiting on customer" is never gamed. Get these wrong and the SLA either flatters you with soft targets or demoralises the team with impossible ones.

As an official ERPNext partner working with Indian teams, we set the Support SLA system up around how you actually deliver support — the priorities and targets your customers were promised, the working hours and holidays your team really keeps, the pause and fulfilment statuses that make the numbers honest, and the notifications and assignment rules that turn an approaching deadline into an action rather than a surprise. Configured once, correctly, an ERPNext SLA stops being a document you cite after a complaint and becomes the quiet system that keeps the complaint from happening.

KEY TAKEAWAYS

1. An SLA turns a vague support promise into a measured contract: a response target and a resolution target, tracked per ticket.
2. In ERPNext, the Service Level Agreement's Priorities table sets First Response Time and Resolution Time per priority; agreements can be scoped to a Customer, Customer Group or Territory.
3. Working Hours and a linked Holiday List decide when the SLA clock runs; 'SLA Paused On' statuses stop it while you wait on the customer, and Total Hold Time is netted out.
4. Every Issue carries live Response By / Resolution By deadlines and an SLA status — First Response Due, Resolution Due, Fulfilled or Failed — with escalation built from those fields via Frappe notifications and assignment rules.
5. The value comes from honest targets, honest priorities and reviewing Failed tickets as a learning loop — an SLA you routinely fail is worse than none.

FAQ

Does ERPNext support response and resolution SLAs out of the box?
Yes. ERPNext's Support module includes a Service Level Agreement doctype where you set a First Response Time and a Resolution Time for each priority, define support working hours and holidays, and choose statuses that pause or fulfil the SLA. Once you switch on 'Track Service Level Agreement' in Support Settings, every governed Issue carries live Response By and Resolution By deadlines and an SLA status of First Response Due, Resolution Due, Fulfilled or Failed.

Can we give some customers tighter SLAs than others?
Yes. An SLA can be the organisation-wide default, or scoped to a specific Customer, Customer Group or Territory, so a premium-support client can carry tighter response and resolution targets than your standard tier — all in the same ERPNext instance. An optional condition can narrow when an

agreement applies further, for example by issue type.

How does ERPNext handle Indian holidays and support hours in an SLA?
Each SLA has a Working Hours table (a row per weekday with start and end times) and a linked Holiday List. The SLA clock only runs during your staffed hours, and days on the Holiday List don't count against your targets — so you can reflect your team's real working week plus India's national and state holidays, rather than being measured against raw round-the-clock time.

What happens to the SLA timer while we're waiting on the customer?
You nominate pause statuses in the agreement's 'SLA Paused On' table — commonly On Hold or Replied. While a ticket sits in one of those statuses, the timer stops; ERPNext records when it went on hold and accumulates a Total Hold Time, then reports a User Resolution Time that strips the waiting out. That way the SLA measures your responsiveness, not time the ball was in the customer's court.

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