ENDER TURING

THE ASYNC SLA ONE-PAGER

Two clocks: split your service level into live and async targets

80/20 was built for a caller waiting on a line. Email, messaging and web forms need their own clock, targets and rules. This page is the standard; pages 2–7 are the toolkit to roll it out.

The rule: measure live contacts by how long the customer waits for you. Measure asynchronous contacts by how soon you reply with substance, how fast you resolve, and whether the customer has to come back.

SYNCHRONOUS

The customer is waiting

Phone · live chat · video

CLOCK RUNS IN

Seconds

PRIMARY TARGET

X% answered within Y seconds

GUARDRAIL

Abandonment rate

CALIBRATE WITH

Your abandonment curve

ASYNCHRONOUS

The customer has left

Email · messaging · web forms · social DMs

CLOCK RUNS IN

Business hours

PRIMARY TARGET

X% substantive replies within N hours

GUARDRAIL

Repeat contact within 72 hours

CALIBRATE WITH

The 30-day test + your promise

The wait-state test: is the customer stuck in the conversation until you reply?

Yes → synchronous clock. No → asynchronous clock. An auto-acknowledgment stops neither.

1 Start at arrival
When the message lands, not when someone picks it up.

2 Stop on substance
A reply that answers, resolves or asks for what you need.

3 Auto-replies don't count
They set expectations. They never stop the clock.

4 Pause only on them
Only while you wait for the customer's answer.

5 Solved, not closed
A repeat contact within 72 hours reopens it.

1 · Measure each contact on its own clock

Classify by the customer's wait state, not by the channel's name. The same chat is live while the customer watches the window and asynchronous once they leave.

The wait-state test

CONTACT	WHILE YOU HAVEN'T REPLIED, THE CUSTOMER…	CLOCK
Phone or video call in a queue	is on the line and can't do anything else	Live
Live chat, window open	watches the window	Live
Chat after the customer leaves	will read your reply later	Async
Messaging: WhatsApp, SMS, in-app	gets a notification and carries on	Async unless both sides are typing
Email, web form, social DM	expects a reply later	Async
Scheduled callback	was promised a time	Async against the promised time

What to measure on each clock

	LIVE (SYNCHRONOUS)	ASYNC
Primary target	X% answered within Y seconds, per 15- or 30-minute interval	X% substantive replies within N business hours
Outcomes	Abandonment rate; first-contact resolution	Time to resolution; repeat contact within 72 hours
Clock stops on	An agent answering	A reply that answers, resolves or asks for what you need
Clock pauses	Never	While waiting on the customer; outside the hours you publish
Report as	Interval service level and abandonment	P90 reply and resolution time; oldest waiting conversation
Staff with	Erlang C or simulation, per interval	Workload within the window; route by due time
Calibrate with	Your own abandonment curve	The 30-day test, your promise and platform limits

Edge cases

- **Messaging that turns live.** When both sides are typing, keep the queue on the async clock and track the reply gap inside the conversation separately.
- **Public social posts.** Asynchronous, but everyone sees the wait. Give them a shorter N only if you can staff it.
- **Bots.** A bot message counts as a reply only when it answers or resolves.
- **Channel labels differ.** ICMI files web chat and SMS under service level and email under response time. The wait-state test decides for your queue.

Platforms set a ceiling

On WhatsApp, each customer message opens or resets a 24-hour customer service window. Once it closes, a business can send only pre-approved template messages.

An async target longer than 24 hours fails on WhatsApp by design. Check the rules of each messaging platform before you publish a promise.

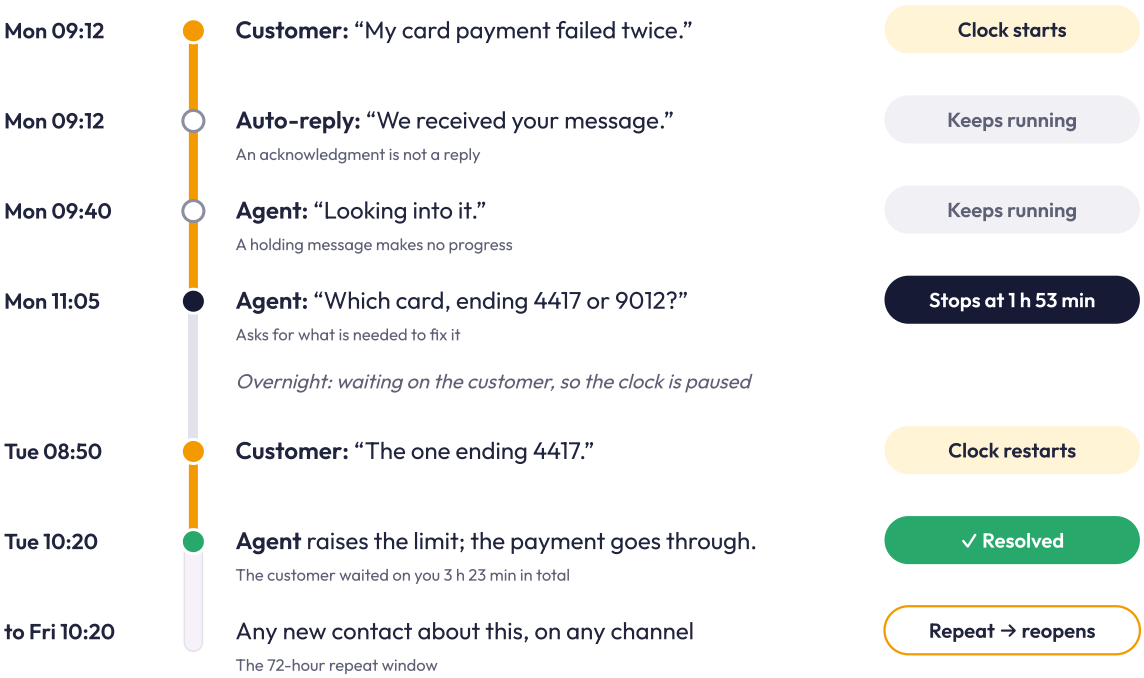
2 · Fix the clock before you set a target

Most async SLAs fail on definitions, not numbers. Write these rules into your helpdesk configuration and into every outsourcing contract.

RULE	IN PRACTICE
1. Start at arrival	Use the timestamp of the customer's message. Routing, triage and assignment time count.
2. Stop on substance	Only a reply that answers, resolves or asks for information you need to proceed stops the clock.
3. Auto-replies never stop it	Acknowledgments, "looking into it", notes, tags, merges and status changes don't count. Keep the auto-reply to set expectations; just don't count it.
4. Pause only on the customer	From the moment you ask the customer for something until they answer. Never for internal hand-offs.
5. Chasers don't reset it	"Any update?" means the clock has run too long. Count the chaser; never restart from it.
6. Resolved means solved	Not "closed". A new contact from the same customer about the same issue within 72 hours, on any channel, reopens it.
7. Hours match the promise	Pause outside business hours only if business hours are what you tell customers.
8. Percentiles, not averages	Report P90 or P95. An average hides the conversations that waited two days.

When does the async clock stop?

One messaging conversation, scored with the clock rules.



Count the auto-reply and this conversation's first response time is 0 minutes.
With the clock rules: first substantive reply 1 h 53 min · resolved in 25 h 8 min of calendar time.

3 · Run the 30-day test

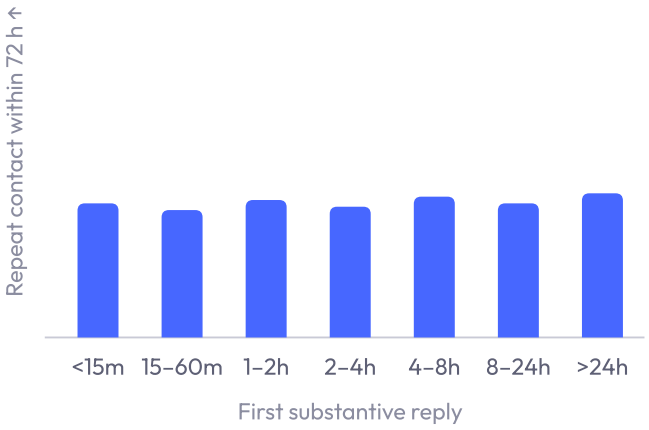
Before you choose N, find out whether reply speed changes what customers do next. The question: does first substantive reply time predict a repeat contact within 72 hours?

1. **Export resolved async conversations** from the last 33 days, leaving out the last 3 so each has a full 72-hour window.
2. **Export every new contact** from the same customers over the same period plus 3 days, on every channel, calls included.
3. **Flag a repeat** when the customer made a new contact within 72 hours of resolution. A "thanks!" in the same thread doesn't count.
4. **Bucket the reply time:** under 15 min, 15–60 min, 1–2 h, 2–4 h, 4–8 h, 8–24 h, over 24 h.
5. **Chart the repeat rate per bucket** with the count beside it. Ignore buckets with fewer than a hundred or so conversations.
6. **Repeat it per topic.** Hard problems get slower replies and more repeats, so an all-topics curve can show the mix, not speed.

The SQL for steps 1–5 and a spreadsheet method are in the appendix on page 7.

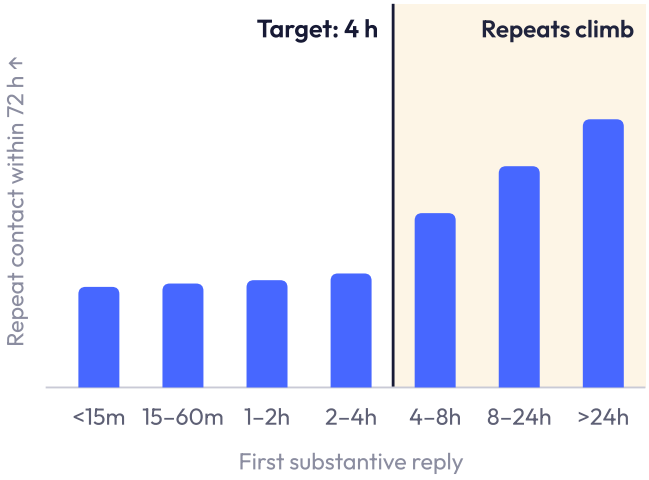
Flat: reply speed isn't the lever

Hold the target and spend the effort on resolution.



Knee: slow replies cost repeats

Set the target just before repeats start to climb.



ILLUSTRATIVE SHAPES Not benchmark data. Run the test on your own 30 days, one topic at a time.

How to read it

- **Flat:** speed isn't driving repeats in that range. Keep a target customers find reasonable and put the effort into resolution.
- **Knee:** set N just before it.
- **Highest in the fastest bucket:** read those conversations. Fast replies may be templates that don't solve the problem.

Then run it again

- With **time to resolution** on the x-axis: the target that matters most.
- With **CSAT or DSAT** on the y-axis, if you survey.
- With a **7-day window** if issues surface slowly; HBR's effort research tracked repeats over 7–14 days.
- **Every quarter**, to check the targets still sit where the curve says.

4 · Set the targets, then staff to them

A target is a promise with a cost. Derive each number from your own data, write down how it is computed, and price it before you commit to it.

Live: X% within Y seconds

- **Y:** plot the share of callers who abandoned by how long they had waited, over 90 days. Put Y before the point where the curve bends upward.
- **X:** each extra point of service level costs more agents than the last. Price the step from 80% to 90% before you promise it.
- **Interval:** measure per 15 or 30 minutes. A good afternoon hides a bad morning.
- **Formula:** write down how abandoned calls are counted and use that formula everywhere.

Async: X% within N hours

- **N:** the earliest of the knee from the 30-day test, the promise you publish, and platform limits such as WhatsApp's 24 hours.
- **X:** set at P90 or P95, not 100%, with a backstop: no conversation waits longer than 2 × N.
- **Guardrail:** repeat contact within 72 hours must not rise when you change N.
- **Resolution:** a P90 time to resolution per topic family, not one number for everything.

Target card — one row per queue; the first row is an example, not a benchmark

QUEUE	CLOCK	PRIMARY TARGET	GUARDRAIL	CLOCK STOPS ON	PAUSES	OWNER · REVIEW
Card support · WhatsApp	*Async*	*90% substantive replies ≤ 2 business h; none > 4 h*	*72-h repeat ≤ today's baseline*	*Answer, fix or needed question*	*Waiting on customer; 20:00–08:00*	*Head of digital care · quarterly*

Staffing the live clock

- Forecast arrivals per interval, size agents with Erlang C or a simulation against X and Y, then add shrinkage.
- Erlang C models callers who wait in a queue until an agent is free. That describes a phone queue, not an inbox.
- **Callbacks** turn a long live wait into an async promise. Measure each one against the time you promised, not the queue's Y.
- **Blended agents:** reserve the live capacity first; async work takes the rest of the hour.

Staffing the async clock

- **Agent hours per day** = conversations × handle time ÷ target occupancy. Handle time is all agent time on the conversation, across every reply.
- **Schedule to the window:** the work due in the next N hours must never exceed the capacity in those hours.
- **Under about an hour**, async work behaves like a queue again: ICMI points back to Erlang C or simulation.
- **Route by due time**, earliest deadline first, not by arrival order.
- **Blend:** async work can fill the quiet hours of live demand, as long as N leaves that slack.
- **Messaging concurrency:** cap it by complexity. More parallel threads mean longer gaps inside each one.

5 · Roll it out in six weeks

One step a week. Weeks 1–3 change how you measure; weeks 4–6 change what you commit to.

WEEK 1
Classify
List every queue and entry point. Tag each live or async with the wait-state test. Record where each current target came from.

WEEK 2
Fix the clock
Configure the clock rules: business-hours calendars, auto-reply exclusion, pending-customer pause, 72-hour reopen.

WEEK 3
Run the 30-day test
Per channel and topic. Add the resolution-time and CSAT views.

WEEK 4
Set targets
Live Y and X from the abandonment curve; async N and X from the knee, the promise and platform limits; guardrails.

WEEK 5
Staff and route
WFM models for both clocks, deadline routing, concurrency caps, a backlog plan.

WEEK 6
Report and contract
Separate scorecards, promises on channel pages and auto-replies, outsourcing SLAs rewritten, quarterly review set.

Checklist

CLASSIFICATION

- [ ] Every queue tagged live or async by the wait-state test
- [ ] Live chat has a timeout after which it becomes async

CLOCK

- [ ] Auto-replies and holding messages excluded from response time
- [ ] Clock starts at arrival, not assignment
- [ ] Pauses only while waiting on the customer
- [ ] Business-hours calendar matches the published promise
- [ ] A repeat contact within 72 hours reopens resolution

TARGETS

- [ ] Live Y set from your own abandonment curve
- [ ] One written service-level formula, used everywhere
- [ ] Async N set from the 30-day test, the promise and platform limits
- [ ] Targets at P90/P95 with a 2 × N backstop

OPERATIONS

- [ ] Repeat contact tracked on every async queue
- [ ] Async staffed by workload and routed by due time
- [ ] Separate live and async scorecards; no blended service level
- [ ] Oldest waiting conversation on the wallboard
- [ ] 30-day test re-run every quarter

Sample wording for outsourcing contracts

1. Service level applies to live contacts only and is measured per 30-minute interval with the formula in Annex A.
2. For asynchronous contacts, a response is a reply that answers the request, resolves it or asks the customer for information needed to proceed. Automated acknowledgments and holding messages are not responses.
3. A conversation is resolved when the customer's request is fulfilled. A new contact from the same customer about the same request within 72 hours reopens it.
4. Repeat contact within 72 hours is reported monthly per queue and may not exceed the baseline in Annex B.

Sample wording only; have your counsel adapt it to your contract.

Read the full guide

How to calculate service level, what a good target is, and the evidence behind this toolkit.

enderturing.com/blog/call-center-service-level

A · Appendix: the 30-day test in SQL

Written for PostgreSQL; rename the tables and fields to match your helpdesk or data-warehouse export. Two inputs: resolved conversations, and every new contact on every channel.

```sql
-- Repeat contact within 72 h by first substantive
-- reply. first_substantive_reply_at must skip
-- auto-replies and holding messages (field notes).
WITH convs AS (
  SELECT conversation_id, customer_id, topic,
         resolved_at,
    EXTRACT(EPOCH FROM first_substantive_reply_at
      - created_at) / 3600.0 AS reply_h
  FROM conversations
  WHERE channel IN ('email', 'messaging', 'web_form')
    AND first_substantive_reply_at IS NOT NULL
    AND resolved_at >= now() - interval '33 days'
    AND resolved_at <  now() - interval '3 days'
), flagged AS (
  SELECT v.*, EXISTS (
    SELECT 1 FROM contacts k  -- every channel
    WHERE k.customer_id = v.customer_id
      AND k.created_at > v.resolved_at
      AND k.created_at <= v.resolved_at
                        + interval '72 hours'
  ) AS repeated
  FROM convs v
)
SELECT CASE
    WHEN reply_h < 0.25 THEN '1 <15m'
    WHEN reply_h < 1    THEN '2 15–60m'
    WHEN reply_h < 2    THEN '3 1–2h'
    WHEN reply_h < 4    THEN '4 2–4h'
    WHEN reply_h < 8    THEN '5 4–8h'
    WHEN reply_h < 24   THEN '6 8–24h'
    ELSE '7 >24h' END AS reply_bucket,
  COUNT(*) AS conversations,
  ROUND(100.0 * AVG(repeated::int), 1)
    AS repeat_pct
FROM flagged
GROUP BY 1 ORDER BY 1;
-- Next: add topic to SELECT and GROUP BY,
-- then swap reply_h for resolution hours.
```

Field notes

FIELD	DEFINITION
created_at	When the customer's first message arrived.
first_substantive_reply_at	First reply by a person or bot that answers, resolves or asks for what you need. Skip automation users, templates and holding macros.
resolved_at	When the request was fulfilled; the last "solved" if the ticket was reopened.
contacts	New conversations and calls, one row each. Replies inside an existing thread are not contacts.
topic	Contact reason: your helpdesk's field or tags, or topics from conversation analytics.

No SQL?

Use a spreadsheet: one row per conversation, one sheet of contacts. The repeat flag is a COUNTIFS over the contacts sheet: same customer ID, start time after resolution and no later than resolution + 3 days.

Business hours: wall-clock hours are fine for the shape of the curve. Use your helpdesk's business-hours timer when you set N.

Sources. ICMI, *Guide to Contact Center Metrics* (2015): service level vs. response time, the three types of response, example objectives · ICMI, *Call Center Metrics: Key Performance Indicators* (2003): staffing work that can wait · ICMI, *Balancing Call Center Efficiency and the Customer Experience* (2011): no industry-standard service level; 22.6% use 80/20 as "the standard" · Toister Performance Solutions (2020): a reply within an hour meets 88% of email expectations · Dixon, Freeman & Toman, "Stop Trying to Delight Your Customers," *Harvard Business Review* (2010): repeat contact and customer effort · Meta, WhatsApp Business Platform documentation: the 24-hour customer service window. Links are in the full guide at enderturing.com/blog/call-center-service-level.