

Re-checking orders already with a carrier

Before the dispatcher evaluates any *new* assignments, it first re-examines every order that is **already sitting with a Third-Party carrier from a previous cycle** — meaning a carrier has been given the order but hasn't yet left the store with it.

This happens during step 1 of [The assignment cycle, step by step](#), for every store, every cycle. It never touches brand-new orders or orders already assigned to your own fleet.

The three questions, in order

For each carrier-held order, up to three questions are asked. The first one that resolves the order stops the sequence.

1. Has the carrier been sitting on this order too long?

Governed by **Max time to accept** (set per store, per carrier — see [Where each setting lives](#)).

- Limit exceeded **and** a real alternative exists right now → the order is cancelled with the carrier immediately and freed for reassignment in this same cycle.
- Limit exceeded but **no alternative exists** → the order is left exactly as it is. Cancelling would only leave it worse off, with nowhere better to send it.
- Limit not reached → move to question 2.

2. Is the carrier's own estimate for reaching the store slipping too far?

Governed by **Max time to reach store** (per store, per carrier).

If the carrier's estimated arrival at the store has drifted past this limit, the order is cancelled immediately — regardless of whether an alternative is available. An unreliable estimate from the carrier itself is reason enough to give up on it.

3. Are we still inside the free-cancellation window, and does our own fleet have room?

Governed by **Free-cancellation window** (per carrier, company-wide).

If yes to both, the order is reset to look **unassigned** — only in the dispatcher's working memory. Nothing is saved, and nothing is cancelled with the carrier yet. The order simply **rejoins the normal assignment process for that store**, exactly like any other unassigned order in that cycle.

“ This is not a special "try own fleet" routine. From this point on, the dispatcher has no notion that the order was ever with a carrier — it goes through PG Mode, Google OR-Tools and Third-Party like everything else, under all the same rules.

If the normal process assigns an own driver to it, two things then happen in strict order:

1. A real cancellation is sent to the original Third-Party carrier.
2. **Only if that cancellation succeeds**, the new internal-driver assignment is saved.

If the cancellation fails, step 2 never happens: nothing changes and the order stays with the carrier. This avoids a confusing state where one order looks assigned in two places at once.

If the normal process doesn't place it with an own driver, the in-memory reset is discarded. The order goes back to exactly how it was — still validly assigned to the carrier — as if the reset had never happened. No cancellation is ever sent in this case.

Questions 1 and 2 look similar but compare different things

This is the single most common source of confusion on this page.

Question	What it actually compares	Consequence
Max time to accept	Real elapsed time — now, versus when the order was given to the carrier. Only fails once enough wall-clock time has genuinely passed.	Cancels, but only if an alternative exists
Max time to reach store	The carrier's own promise — their reported arrival estimate, versus when the order was given to them. It never looks at the current time at all. Can fail seconds after assignment, if the carrier's own estimate is already too far out.	Cancels regardless of alternatives
Free-cancellation window	Same style of comparison as <i>Max time to accept</i> (now versus assignment time), but with a different consequence: passing it cancels nothing by itself. It only opens the door to the no-commitment own-fleet attempt described above.	Enables the risk-free retry

What happens if the own-fleet attempt doesn't work out

The order is deliberately **skipped from that same cycle's Third-Party fallback pass** — it is not re-evaluated against carriers again in this run.

This is intentional. The order is already validly assigned to a carrier, so re-running the fallback would be redundant at best, and risks producing a second, conflicting carrier assignment for an order that already has one. It carries on with its existing assignment until the next relevant check — a future cycle, or an actual delivery-time failure.

Why this matters

Without this check, an order could sit with a slow or unresponsive carrier indefinitely, or stay there even after your own fleet became free to take it sooner.

The own-fleet attempt is deliberately risk-free: if it doesn't pan out the order is left exactly as it was, the carrier is none the wiser, and nothing is ever cancelled without a confirmed replacement already lined up.