Automatic Stage Progression Using Milestone Dates

1. Overview

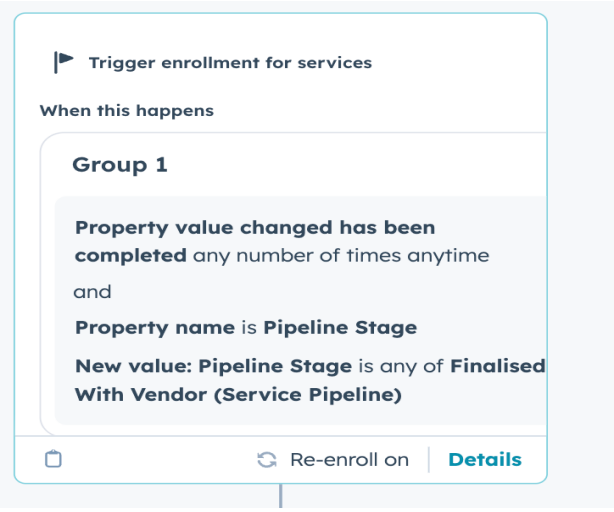
This automation advances service records through pipeline stages based on key date properties (e.g., moving a record to **Campaign Live** on its scheduled live date, and moving to **Campaign End** once the end date is reached). It ensures the pipeline reflects real operational status automatically, without relying on manual stage updates.

Why this matters: In date-driven operations, stage accuracy isn't just "nice to have", it affects delivery, reporting, and accountability. Automating stage updates based on milestone dates improves operational visibility, reduces missed transitions, and keeps reporting consistent on what is upcoming, live, or completed. It becomes especially valuable when there are **many records moving in parallel**, because it:

- **Prevents manual errors at scale** (missed updates, wrong stages, inconsistent timing)
- **Maintains accurate workload visibility** (teams can clearly see what is launching today vs. later)
- **Reduces operational risk** by ensuring critical date-based transitions happen even when staff are busy or unavailable
- **Improves SLA performance** by triggering the next workflow steps on time (notifications, handoffs, checks)
- **Strengthens forecasting and planning** (upcoming vs. live vs. completed is always reliable)
- **Creates a consistent "source of truth"** across teams everyone sees the same status without chasing updates
- **Enables downstream automations** (billing, client comms, approvals, reporting) to run at the right time
- **Supports auditability** because workflow history shows exactly when and why the stage changed

2. Trigger design

The workflow enrolls records when they reach a specific stage and only proceeds if the relevant date property is known, keeping the automation targeted and prevents accidental stage changes.

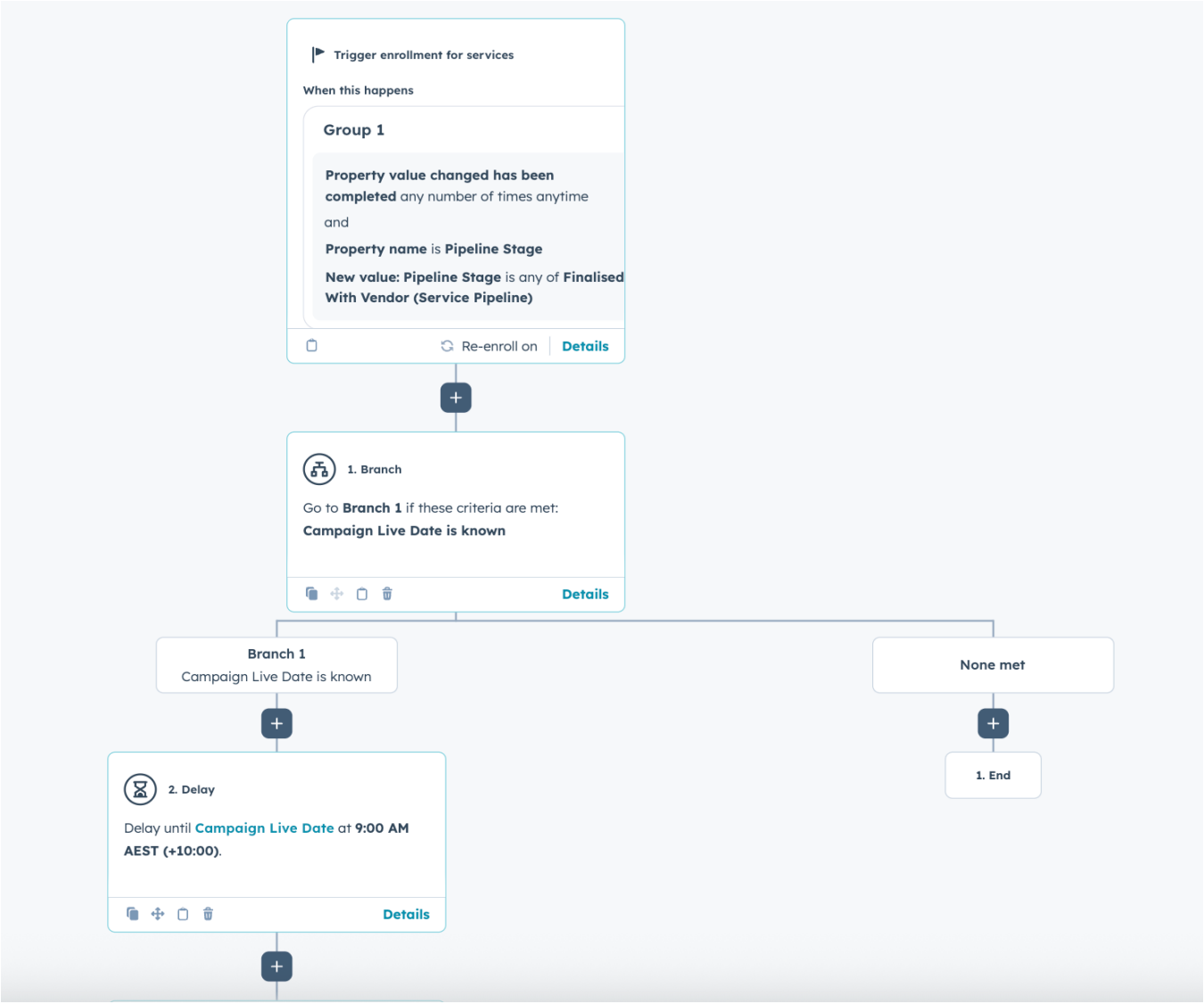


- **Workflow type:** Service (or equivalent object) workflow in HubSpot.
- **Enroll when:** Pipeline Stage changes to a defined “ready” stage
- **Re-enrollment:** Enabled (so records can re-trigger if they re-enter the ready stage).
- **Gate condition:** Only proceed if *Campaign Live Date* is known (not empty).

3. How it works

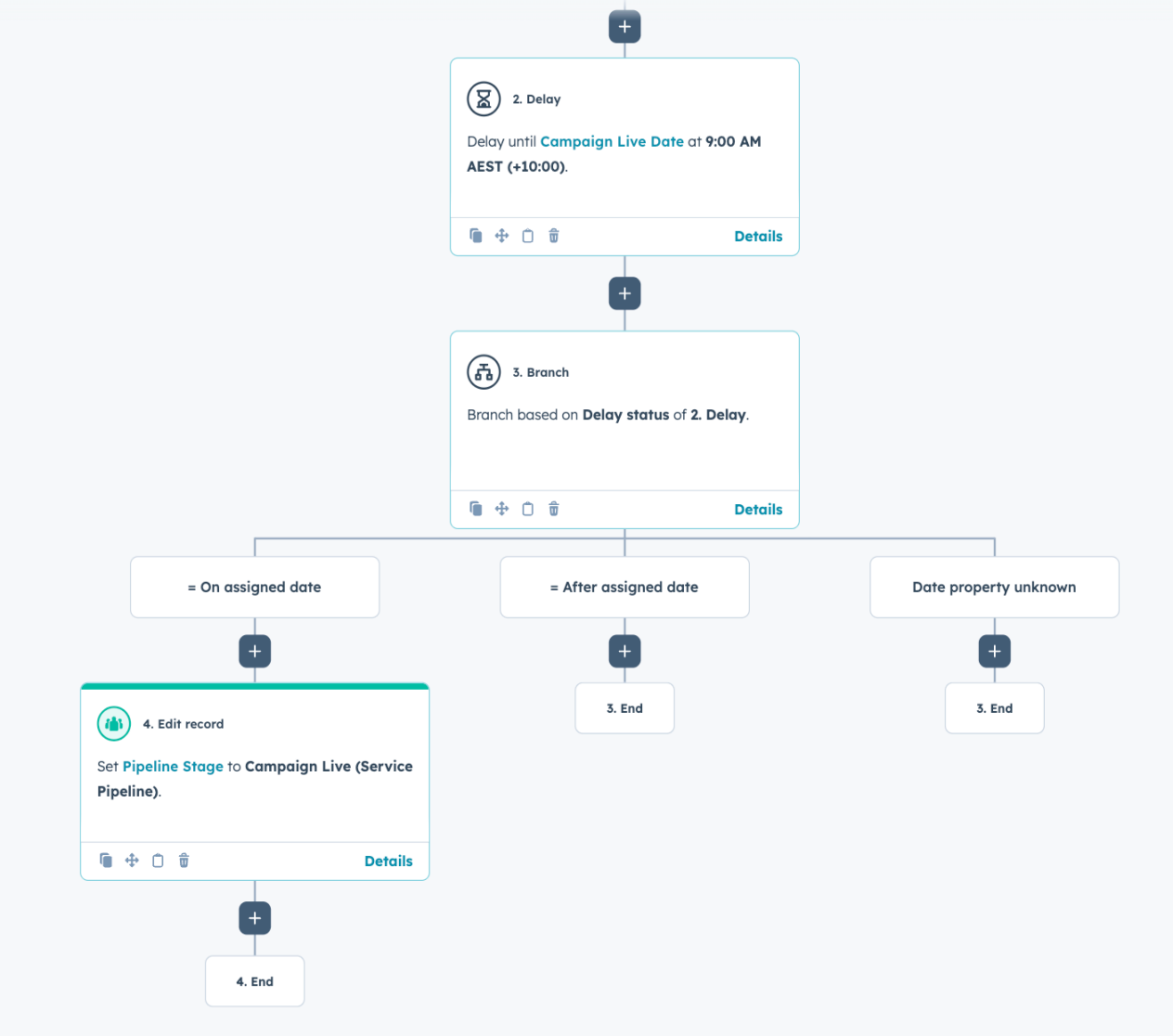
Once enrolled, the workflow waits until the scheduled date/time and then updates the pipeline stage. A follow-up branch ensures the update only happens at the correct time and avoids changes if the date becomes unknown or is moved.

4. **Branch 1:** Check that Campaign Live Date is known. If unknown, end safely.



- **Delay:** Wait until Campaign Live Date at a set time (e.g., 9:00 AM in the portal time zone).

- **Branch 2:** Evaluate the delay outcome (on assigned date / after assigned date / date unknown).



- **Edit record:** If on the assigned date, set Pipeline Stage to "Campaign Live".
- **End:** For other paths, end (prevents unexpected updates).

5. Stage transitions (template)

Trigger stage (enrollment)	Date property used	Delay until	Stage set by automation
Record enters **Ready / Approved stage** (e.g., "Finalized with Vendor")	Campaign Live Date	Wait until **Campaign Live Date** (e.g., 9:00 AM)	Set stage to **Campaign Live**

6. Configuration steps (HubSpot)

- Set enrollment: Pipeline Stage changes to the chosen "ready" stage.
- Add a branch: Campaign Live Date is known. If not, end.
- Add a delay: Delay until Campaign Live Date at your chosen time (e.g., 9:00 AM).
- Add a branch based on the delay status. In the "on assigned date" path, set Pipeline Stage to Campaign Live.
- End all other paths safely (after assigned date / date unknown).

Key Information:

- **Date property required:** Always gate the workflow with "date is known" checks.
- **Time zone consistency:** Document the portal time zone used by delays (e.g., AEST).
- **Stage naming:** If stages are renamed, update the trigger and edit-record steps.
- **Avoid double automation:** If you also move stages manually, define a clear process to prevent conflicts.

7. Conclusion

This date-driven stage automation keeps the service pipeline accurate and current by aligning stage movement with real campaign timelines. It reduces manual administration, improves team visibility into what is going live and when, and strengthens reporting by ensuring stages update consistently. As a result, stakeholders can rely on the pipeline as an operational source of truth for upcoming launches and completed campaigns.

Optional extension: second workflow to move records to "Campaign End"

In many date-driven pipelines, reaching **Campaign Live** is only the first milestone. A second workflow can be added to ensure records also progress automatically when the campaign finishes. This prevents records from remaining "live" indefinitely and keeps the pipeline aligned with real operational status.

How the second workflow works (high level):

1. **Enrollment trigger:** The record enters the **Campaign Live** stage (or another defined "live/in-progress" stage).
2. **Gate condition:** The workflow checks that **Campaign End Date** is known (not blank). If it is unknown, the workflow ends safely without changing anything.
3. **Delay:** The workflow waits until the **Campaign End Date** at a fixed time (e.g., 4:00 PM in the portal time zone).
4. **Stage update:** Once the delay completes on the assigned date, the workflow updates the record to **Campaign End** (or equivalent completed stage).

5. **Safeguard branch (recommended):** If the end date changes mid-way or becomes unknown, the workflow should end without updating the stage to avoid incorrect transitions.

Why this second workflow is useful:

- Ensures “live” records don’t remain open after completion
- Improves reporting accuracy (live vs. completed counts are reliable)
- Reduces manual admin when managing many campaigns/records in parallel
- Supports downstream actions tied to completion (handoff, reporting, invoicing, archival, follow-up)

Implementation note: This is typically set up as a **separate workflow** (rather than combining everything into one), because it keeps each automation simpler to test, maintain, and audit.