

**HRS workflow may stay for max 3 or 4 days. normally within a day**

Zeebe easily handles workflows that stay active for **hours or a few days**—your HRS target of “normally within a day, occasionally up to four days” is well inside its design envelope.

* Zeebe stores every state change in an **append-only event log** backed by distributable partitions, so a workflow can remain “in flight” as long as its data is kept in the log. The default log retention is size-based, not time-limited; you set the disk quota rather than a maximum duration.[[1]](#fn1)
* Camunda’s own guidance notes that Zeebe is not ideal for workflows that must stay open for **months or years**, but durations of **“days or weeks” are fine** provided adequate log storage and snapshot frequency are configured.[[2]](#fn2)
* Timers and message correlations use 64-bit epoch timestamps, so delays of **years** are technically possible; the main operational consideration is broker disk space and snapshot schedule, not an intrinsic time cap.[[1]](#fn1)

Operational tips for multi-day workflows:

1. **Increase log size or enable S3/NFS off-loading** if you expect many concurrent 3- to 4-day instances.
2. **Tune snapshot/compaction** intervals (e.g., every 5 min) so the broker can release segment files once completed instances are exported.
3. **Set job timeouts** generously (e.g., a few hours) if an activity can block while awaiting an external system; workers can still send heartbeats to keep the job lock active.
4. Use the **Operate** UI or exporter metrics to watch log utilization and snapshot lag.

In short, for HRS workflows that usually finish within 24 h and very rarely extend to four days, Zeebe is an appropriate choice without special work-arounds—just monitor broker storage and cluster health.

⁂

1. <https://docs.camunda.io/docs/components/zeebe/technical-concepts/internal-processing/>

1. <https://forum.camunda.io/t/long-running-months-years-or-maybe-forever-workflows/36659>