

zapier.georgelambert.org · Markdown indexes

VERAE TIME × ZAPIER

Uptime / replica floors

<site/packages/overview/84-uptime.md>

4. How the system maintains uptime

Uptime is **NATS durability + fleet replica floors + more than one machine**, not a single always-on Zapier connection.

Replica floors (`verae-fleet`)

`packages/verae-fleet/fleet.json` sets `min` / `max` / `keepFloor` per service. **Available** means running, healthy, and not paused.

Service	Default min	keepFloor
tree-node	3	yes
archive-worm	3	yes
archive-aggregator, job-poller, webhook-deliver	1	yes
zappier-edge, middleware-http	1	yes

If a tree node is paused, crashes, or fails `/health`, fleet **starts another copy** until three are available. Operator console: <http://127.0.0.1:3850/> (Fleet tab — green / yellow / red; Trace tab for hop tests; Docs tab for reading order).

Restart and pause

- Unhealthy `/health` → same instance id restarted.
- Pause does not count toward `min`.
- `stop` on a service disables `keepFloor` for that service.

Spread across machines

`machines.json` lists hosts (`local` , `ns1 = marchon@70.88.205.138` with `~/.ssh/id_ed25519` , optional `lan-134`). New replicas go to the **least-loaded** eligible host. Remote spawn/health/kill is SSH; workers bind loopback on the remote box.

JetStream

Work queues (`jobs.watch` , `webhooks.deliver`) replay if a consumer dies. Event stream (`jobs.events`) lets waiters and webhook routers catch up. A 3-node cluster keeps the stream if one NATS server is down.

What Zapier sees

HTTPS 202 `jobId` , wait JSON, or REST Hook. Timeouts return `pending` + `jobId` so the **Timestamp Completed** trigger can finish the job. Zapier retries are safe: the same SHA-256 returns the original seal (`existing: true`).