AIPIVT

Leave PHP without freezing the product.

12 checks before you rebuild a live PHP product in Go, Rust, Python, .NET, or Node.js.

Users stay on the PHP app until their path is replaced. This page is not a quote and not an estimate.

Pin

1. List the routes, cron jobs, and queue workers production actually runs, from access logs and the crontab, not from the router file. A route nobody called in 90 days is a delete candidate, not a port.
2. Each path that stays has a parity test: the same recorded request goes to the PHP app and to the replacement, and the responses are compared. The PHP app is the spec.
3. The behavior nobody wrote down is listed: loose == comparisons, float rounding in money math, the timezone and locale in php.ini, what a null does inside a string function. Until someone decides otherwise, it is part of the contract.
4. The replacement reads and writes the same database the PHP app does. A schema change waits until no PHP code reads the old shape.

Replace

5. Users stay on the PHP app until their path is replaced. A proxy in front routes one path at a time. Nobody waits for a cut-over weekend.
6. Every replaced path has a way back: one routing change sends its traffic to the PHP app again.
7. Sessions, login cookies, and stored password hashes work on both sides of the proxy. A user does not log in again because a route moved.
8. Coding agents may write the port. The parity test, not the agent's summary, decides whether a path ships.

Own

9. The target stack is chosen by who will run it in 18 months, not by a benchmark. If nobody on the team writes Rust, Rust needs a hire before it needs a port.
10. Before anyone claims the new stack is faster or cheaper, measure the PHP app: p95 latency and cost per request on today's traffic. A rebuild justified by speed needs that number first.
11. One person on your team is named for each replaced path and can change it without the people who ported it.
12. The PHP app is switched off one path at a time, and the last route has a date. A rebuild that keeps both forever is two products.

AIPIVT. A rebuild is a sequence of fixed milestones. The first is shaped for four weeks: the boundary decision and one replaced path.