

# Baseline assessment

Check each statement that's true today, not planned. Each stage's score is the number checked, out of three. Start at the lowest-numbered stage scoring below three.

## 1. Split the bill Score: \_\_\_\_ / 3

- ☐ We can separate build spend from run spend on this quarter's bill.
- ☐ AI calls are tagged to a feature or system at the source, not reconciled afterward.
- ☐ Someone owns the tagging convention, and new agentic workloads inherit it automatically.

## 2. Attribute spend to outcomes Score: \_\_\_\_ / 3

- ☐ For each AI surface, we have named the business outcome it supports.
- ☐ We can state cost per outcome for a production feature, in the unit our business already reports in.
- ☐ For at least one surface, we can point to a metric (CSAT, retention, MTTR) that moved.

## 3. Set a baseline you can prove Score: \_\_\_\_ / 3

- ☐ An eval baseline was recorded at deployment for each production surface.
- ☐ Model, context, and tool changes automatically trigger a comparison against that baseline.
- ☐ Automated quality gates run before AI-generated changes reach production.

## 4. Decide: fund, fix, or stop Score: \_\_\_\_ / 3

- ☐ Every production surface carries a current decision: fund, fix, or stop.
- ☐ We have increased investment in at least one surface because the numbers justified it.
- ☐ We have retired at least one surface because it cost more than it returned.

## 5. Forecast before you build Score: \_\_\_\_ / 3

- ☐ New AI features carry a cost estimate at expected volume before build starts.
- ☐ Unit economics appear in the review gate alongside the design.
- ☐ We have changed a feature's design because the forecast did not work.

## 6. Make it stick Score: \_\_\_\_ / 3

- ☐ Each surface has a named cost owner who sees its spend.
- ☐ Cost and autonomy boundaries are machine-enforced, and alerting is tied to business metrics, not raw spend.
- ☐ A defined process exists for evaluating new model tiers as providers release them.

### Interpreting your score: if your lowest incomplete stage is:

- **Stage 1 or 2:** You cannot yet say what an outcome costs or split build spend from run spend.
- **Stage 3:** You cannot yet prove safe autonomous operation.
- **Stage 4:** You have the data but are not yet using it in every direction, the most common place to stall.
- **Stage 5:** You are still finding out what a feature costs after the invoice arrives.
- **Stage 6:** You have the measurements but nothing keeps them current as models, volume, and teams change.