

# ALIGN Framework

TPM Compass - A five-step program planning framework

---

## How to use this framework

Use ALIGN at the start of program planning, during steering reviews, and whenever the program drifts from its original intent. It is a checklist for thinking, not a methodology to follow blindly. Each step produces a concrete artifact you can share with sponsors, engineers, and stakeholders.

## A - Align Business Outcomes

Define the business change the program exists to create. Every successful program begins with a business outcome - not with a feature backlog.

### Questions to answer

- What business outcome will change because of this program?
- How will we measure that change?
- Who is ultimately accountable for delivering it?

### Output

A one-paragraph outcome statement and 2-4 measurable objectives.

### Example

"By the end of Q2, engineering teams will deploy production changes through a standardized platform with predictable governance, reducing production incidents caused by deployment variance by 30% and cutting release coordination time by 50%."

## L - List Capabilities

Identify what the organization must be able to do once the program is complete. Capabilities describe behavior and competence, not systems and features. A feature delivers functionality. A capability changes organizational behavior.

### Questions to answer

- What should the organization be able to do after this program that it cannot do today?
- Which capabilities must exist before others can be built?
- What adoption, training, or operating-model changes are required?

### Output

A prioritized capability map grouped by phase.

### Example capabilities

- Standardize application deployments across engineering organizations.
- Enable repeatable production deployments with consistent governance.
- Provide deployment visibility for Engineering, SRE, and Leadership.
- Support controlled rollout of platform capabilities across environments.

## ALIGN Framework

TPM Compass - A five-step program planning framework

### I - Identify Dependencies

Surface cross-team, cross-system, and cross-organization dependencies before execution begins. The most important dependencies in enterprise programs are rarely technical. They are organizational.

#### Dependency categories

| Category    | Examples                                              |
|-------------|-------------------------------------------------------|
| Technical   | Shared services, APIs, deployment frameworks          |
| Platform    | Kubernetes upgrades, infrastructure readiness         |
| Security    | Approvals, secret rotation, vulnerability remediation |
| Operational | SRE readiness, monitoring, runbooks                   |
| Governance  | Architecture reviews, release sign-offs               |
| Product     | Feature readiness, adoption sequencing                |

#### Output

A dependency register with owner, type, blocker risk, target date, status, and escalation path.

### G - Govern Execution

Design lightweight decision forums that accelerate decisions without creating bureaucracy. Good governance creates clarity, accountability, and faster decisions.

#### Governance layers

- Steering Forum - Sponsors and executives decide trade-offs, funding, and major scope changes.
- Architecture Forum - Engineering leads decide technical direction and long-term platform consistency.
- Program Review Forum - Delivery leads resolve blockers, dependencies, and release readiness.

#### Output

A governance calendar with named forums, decision rights, attendees, and escalation paths.

### N - Navigate Using Metrics

Select a small set of outcome and health metrics that drive decisions. Start with four metrics, not forty. Every metric should answer: Will this metric change a decision?

#### Metric categories

| Type     | Purpose                   | Examples                                   |
|----------|---------------------------|--------------------------------------------|
| Outcome  | Did the business change?  | Adoption rate, incident reduction          |
| Decision | Should we change course?  | Dependency readiness, milestone confidence |
| Health   | Is execution sustainable? | Team capacity, action closure rate         |

#### Output

A metrics scorecard with owner, target, frequency, and decision trigger for each metric.

## ALIGN Framework

TPM Compass - A five-step program planning framework

---

### ALIGN Checklist

Use this checklist before moving from planning into execution.

- Business outcomes are written and agreed by the accountable sponsor.
- Capabilities are listed and sequenced by phase.
- Dependencies are captured in a register with named owners and escalation paths.
- Governance forums are defined with clear decision rights.
- Metrics are selected and linked to decisions, not just reporting.
- The Program Charter documents outcomes, capabilities, dependencies, governance, and metrics.
- Stakeholders have reviewed the plan before execution begins.

### Summary

| Step | Focus                   | Key Question                                 |
|------|-------------------------|----------------------------------------------|
| A    | Align Business Outcomes | Why does this program exist?                 |
| L    | List Capabilities       | What must the organization be able to do?    |
| I    | Identify Dependencies   | What must be true before execution succeeds? |
| G    | Govern Execution        | How will decisions be made and escalated?    |
| N    | Navigate Using Metrics  | How will we know we are on track?            |