SJH Solutions

5-Point Programme Governance Framework: How Mid-Market Organisations Can Rescue Failing Programmes

Introduction

Mid-market organisations face a critical challenge: programmes fail silently. Not in one dramatic moment, but through a slow accumulation of misaligned decisions, unclear accountability, and absent governance.

Over 20 years leading transformation programmes at FTSE100 companies, I've observed a consistent pattern. Programmes that deliver on time and within budget share one thing in common: they have clear governance structures. Programmes that spiral in cost and timeline share another: they don't.

This article outlines the 5-Point Programme Governance Framework I've used to rescue failing programmes and deliver consistent results across digital transformations, infrastructure modernisation, and organisational change initiatives.

The framework addresses the root causes of programme failure: unclear decision rights, absent accountability, undefined dependencies, uncontrolled scope, and misaligned stakeholders.

Why Governance Fails in Mid-Market Organisations

Most mid-market organisations operate without formal programme governance. Here's why this becomes a problem:

1. No Clear Decision Rights

When decision authority isn't defined, programmes stall. A single approval can take weeks because no one is accountable for making it. Teams escalate decisions upward repeatedly, creating bottlenecks. Leadership becomes reactive rather than proactive.

2. Accountability Spreads Across Multiple People

When everyone is responsible, no one is responsible. Multiple team members believe others are tracking scope, budget, or risk. Budget overruns aren't caught until it's too late. Schedule delays compound because no single person owns the schedule.

3. Dependencies Are Invisible

Teams work in parallel without understanding how their work connects. One team's delay cascades across the programme because dependencies weren't mapped. Bottlenecks appear unexpectedly. Recovery becomes chaotic.

4. Scope Creep Is Uncontrolled

Change requests arrive continuously. Some are approved informally. Others slip in without approval. By month six, the programme is 40% larger than planned, but no one explicitly decided to expand it. Budget and timeline assumptions no longer apply.

5. Stakeholders Are Misaligned

Executive sponsors expect one outcome. Project teams are delivering something different. Business users have different priorities again. The programme succeeds by one measure and fails by another. Trust erodes.

These five failures feed each other. Unclear decision rights lead to scope creep. Scope creep creates untracked dependencies. Untracked dependencies cause delays. Delays trigger stakeholder misalignment. Misalignment undermines accountability.

The solution is not more process. It's clarity.

The 5-Point Programme Governance Framework

The framework consists of five interconnected elements. Each one addresses a specific failure point. Together, they create a system where programmes stay on track.

Point 1: Establish Clear Decision Rights

Definition: Every decision required to deliver the programme has a clear owner, timeline, and escalation path.

Implementation:

Create a Decision Rights Matrix that defines who decides what. This matrix answers three questions:

1. Who makes this decision?
2. Who must be consulted before the decision is made?
3. Who must be informed after the decision is made?

Examples of decisions that need clear ownership:

- Scope changes (who approves? what's the process?)
- Budget reallocation (who can approve? up to what amount?)
- Timeline adjustments (who decides? based on what criteria?)
- Risk mitigation (who owns each risk? who escalates?)
- Stakeholder conflicts (who resolves when stakeholders disagree?)

Assign decision owners at two levels:

Level 1: Tactical decisions (within the programme team). Example: Project Manager decides daily task priorities. Reported weekly.

Level 2: Strategic decisions (steering committee). Example: Executive Sponsor decides major scope changes. Triggered when Level 1 escalates.

Define decision timelines. A decision that takes three weeks to make is a bottleneck. Set expectations: "Scope change decisions are made within 5 business days."

When decisions stall, escalation is automatic. If the assigned owner hasn't decided within the timeline, the decision moves to the next level.

Result: No ambiguity. No endless waiting. No decisions made by default because no one stepped forward.

Point 2: Define Clear Accountability Structures

Definition: Every deliverable, risk, budget line, and timeline has a single owner who is accountable for delivering or escalating.

Implementation:

Create a Responsibility Assignment Matrix (RACI) for the programme. This maps:

- Every work package / deliverable
- Every risk
- Every budget responsibility
- Against the person accountable

RACI defines four roles:

- **Responsible**: Does the work
- **Accountable**: Ultimate owner; escalates if work is at risk
- **Consulted**: Provides input
- **Informed**: Gets updates

Critical rule: Every deliverable has exactly one person marked "Accountable". Not two. Not a team. One person.

This person is responsible for knowing status at all times. They escalate before problems become crises. They coordinate across teams when dependencies emerge.

Accountability extends beyond delivery:

- Budget accountability: Who tracks spending on this workstream? Who escalates if spending exceeds forecast?
- Schedule accountability: Who owns this milestone? Who escalates if it's at risk?
- Risk accountability: Who owns this risk? Who implements mitigation? Who escalates if the risk materialises?

Hold accountability reviews monthly. Each accountable person reports:

- Status (on track / at risk / off track)
- Upcoming risks or blockers
- Escalations needed

Result: Clear ownership. Early escalation. No surprises in month six.

Point 3: Map Dependencies and Identify Bottlenecks

Definition: Every deliverable that depends on another deliverable is explicitly mapped. Bottlenecks are identified before they cause delays.

Implementation:

Create a Dependency Map for the programme. This shows:

- Which deliverables must complete before others can start
- Which teams are dependent on each other
- Which external dependencies exist (vendors, third parties, other programmes)
- Which dependencies are critical path (delays here delay the entire programme)

Use a simple format:

Deliverable A → (depends on) → Deliverable B Timeline: Deliverable B must complete by [date] for Deliverable A to start on schedule

Identify bottlenecks explicitly:

A bottleneck occurs when:

- One team depends on multiple other teams, but those teams have conflicting priorities
- A single external vendor is critical path for multiple workstreams
- A key person is required for multiple parallel deliverables

Once bottlenecks are identified, solve them proactively:

Example bottleneck: Infrastructure team is critical path for three parallel workstreams. Solution: Assign additional capacity to infrastructure team, or re-sequence work to reduce parallel dependencies.

Update the Dependency Map monthly. As work completes, dependencies change. New bottlenecks may emerge.

Result: No unexpected delays from hidden dependencies. Bottlenecks are visible and managed.

Point 4: Control Scope Through Formal Change Management

Definition: Scope is baselined at the start. Changes are documented, approved, and tracked. Budget and timeline are adjusted when scope changes.

Implementation:

Establish a Scope Baseline:

At programme kick-off, document:

- What is included in scope
- What is explicitly excluded from scope
- The assumptions that underpin scope

This baseline is the reference point for all future decisions.

Create a Change Management Process:

Every scope change follows a formal process:

Step 1: Change Request submitted (what's changing? why?) Step 2: Impact Assessment (how does this affect budget, timeline, resources, risks?) Step 3: Approval (using Decision Rights from Point 1) Step 4: Documentation (change is logged, baseline updated) Step 5: Communication (stakeholders are informed of scope and timeline implications)

Categorise changes by impact:

Minor changes (low cost, low risk): Approved by Project Manager. Reported to Steering Committee. Major changes (significant cost, timeline impact): Approved by Steering Committee. Communicated to all stakeholders.

Track all changes:

Maintain a Change Register that shows:

- What changed
- Why it changed
- Who approved it
- Budget impact
- Timeline impact

At monthly reviews, discuss trends: Are changes increasing? Are they driven by poor initial planning or genuine business needs?

Result: Scope is controlled. Surprises are eliminated. Budget and timeline forecasts remain accurate.

Point 5: Align and Engage Stakeholders Systematically

Definition: Every stakeholder knows the programme status, risks, and decisions affecting them. Communication is proactive, not reactive.

Implementation:

Identify all stakeholders:

Not just executives. Include:

- Executive Sponsors (strategic decisions)
- Steering Committee (oversight)
- Project teams (delivery)
- Business users (change adoption)
- Operations teams (ongoing support)
- Finance (budget tracking)
- IT / Infrastructure (technical dependencies)

Create a Stakeholder Engagement Plan:

For each stakeholder group, define:

1. **What they need to know**: What information is relevant to them?
2. **When they need to know**: What's the communication frequency?
3. **How they'll receive information**: Status report? Meeting? Newsletter?
4. **What decisions they influence**: Where do they have a voice?

Example:

Steering Committee

- Need to know: Schedule status, budget status, major risks, escalations
- Frequency: Monthly steering meeting
- Format: Status report + dashboard + discussion
- Decisions: Approve scope changes, reallocate budget, resolve stakeholder conflicts

Project Teams

- Need to know: Next week's priorities, dependencies from other teams, new risks
- Frequency: Weekly team sync
- Format: 30-minute standup + async updates
- Decisions: Daily task planning, tactical escalations

Business Users

- Need to know: How the programme affects their work, when they'll need to change, how to provide feedback
- Frequency: Monthly updates during critical phases, quarterly during delivery
- Format: Town hall or email update
- Decisions: Feature prioritisation, change adoption feedback

Create a communication cadence:

- Weekly: Team syncs, dependency reviews
- Bi-weekly: Escalation review
- Monthly: Steering Committee meeting, status report to all stakeholders
- Quarterly: Executive update, programme health review

Use dashboards:

One-page dashboard showing:

- Schedule status (% complete, on track / at risk)
- Budget status (spent vs. forecast vs. approved)
- Risk register (top 5 risks, mitigation status)
- Upcoming milestones

Everyone sees the same data. Discussions are fact-based, not opinion-based.

Manage conflicts proactively:

When stakeholders disagree (e.g., executive wants faster timeline, but teams say it's not possible), escalate using Decision Rights (Point 1). Don't let conflicts simmer.

Result: Stakeholders are informed. Conflicts are resolved early. Trust is maintained.

How These Five Points Work Together

The framework is not five separate activities. It's a system:

Decision Rights (Point 1) enable **Accountability** (Point 2). When decision owners are clear, accountable people know when to escalate.

Accountability (Point 2) feeds **Dependency Management** (Point 3). Accountable people track their dependencies and surface bottlenecks.

Dependency Management (Point 3) informs **Scope Control** (Point 4). When dependencies are clear, scope changes are understood in context.

Scope Control (Point 4) requires **Stakeholder Alignment** (Point 5). When scope changes, stakeholders must be informed and aligned on implications.

Stakeholder Alignment (Point 5) reinforces **Decision Rights** (Point 1). Aligned stakeholders support decisions because they understand the reasoning.

In a poorly governed programme, these break down: decisions stall, accountability is unclear, dependencies hide bottlenecks, scope creeps, stakeholders misalign, and the programme fails.

In a well-governed programme, these five points create a feedback loop that keeps the programme on track.

Real-World Application: A Case Study

I rescued a failed network security programme at a major retailer using this framework. The programme had already failed twice. Three years of delays. Cost overruns. Failed go-lives.

When I arrived, here's what I found:

- No clear decision rights. Security team, IT team, and business were making conflicting decisions. Approvals took months.
- Accountability was distributed. Multiple people "owned" the programme. No one person was accountable for delivering.
- Dependencies were invisible. The programme was attempting parallel deployment across 200+ locations without understanding where bottlenecks would occur.
- Scope was uncontrolled. Initial scope was for 500 endpoints. By year three, it had grown to 200,000+ endpoints. No one had formally approved this change.

- Stakeholders were misaligned. IT wanted technical perfection. Business wanted fast deployment. Security wanted comprehensive coverage. No one was aligned.

I implemented the 5-Point Framework:

Point 1 - Decision Rights: Created a Decision Matrix. Security approvals took 5 days instead of months. The programme moved again.

Point 2 - Accountability: Assigned clear owners for each region's deployment. Each owner reported weekly. No more diffused responsibility.

Point 3 - Dependencies: Mapped deployment sequencing. Identified that proof of concept in test stores was a bottleneck. Solved it by parallel testing.

Point 4 - Scope Control: Baselined scope at 200+ endpoints. Changes to larger deployments went through formal approval. Scope was now predictable.

Point 5 - Stakeholder Alignment: Monthly steering meetings with IT, Business, and Security. Same dashboard. Shared visibility. Conflicts were resolved in meetings, not through escalation.

Result: 95% schedule adherence. Zero service disruption. Global deployment completed on time.

The framework works because it removes ambiguity. When programmes fail, it's rarely because of technical complexity. It's because of unclear governance.

Conclusion

Mid-market organisations don't need more process. They need clear governance.

The 5-Point Programme Governance Framework provides that clarity:

1. **Clear Decision Rights** eliminate bottlenecks
2. **Clear Accountability** enables early escalation
3. **Visible Dependencies** expose bottlenecks before they cause delays
4. **Controlled Scope** keeps budget and timeline forecasts accurate
5. **Aligned Stakeholders** maintain trust and support

These five points work together as a system. Implement all five, and programmes stay on track. Skip one, and the others weaken.

If your programme is showing signs of failure—budget spiralling, timeline slipping, stakeholders misaligned—governance is where to start.

The good news: implementing governance doesn't require restarting the programme. It can be introduced mid-stream. Often, it's the fastest way to recover.

Next Steps

If your programme is at risk, or you want to assess your current governance, book a free 15-minute Discovery Call.

We'll discuss your programme structure, identify governance gaps, and recommend specific fixes.

Email: info@sjhsolutions.co.uk Phone: 07857 315259 Website: sjhsolutions.co.uk