

DP16 – Roadmap and Milestones

Honest, resourced, and revisable commitments about what gets built, when, and with what evidence

1. Purpose of This Draft

This draft articulates Desirable Property 16 (DP16) as the condition under which the meta-layer's evolution is honestly communicated and milestone accountability is structural, so participants, builders, and communities can plan, trust, and contest direction without being surprised by silent pivots, vapor schedules, or roadmaps that function as marketing fiction.

DP16 connects DP14 (trust and transparency), DP15 (security posture evolution), DP17 (financial sustainability), DP3 (adaptive governance), DP9 (builder expectations), and DP11 (ethical AI capability claims).

If DP16 is weak, predictable failures follow: hype cycles, burned contributors, communities built on assumptions later voided, and regulators correctly skeptical of the entire effort.

DP16 does not promise perfect prediction. It defines minimum honesty conditions for roadmapping in a complex sociotechnical system.

2. Problem Statement

In today's web, roadmaps are often performative rather than operational.

Dates shift silently, features ship without promised safeguards, and "soon" becomes a placeholder for uncertainty rather than a bounded commitment. Participants and builders are expected to plan around signals that are not designed to be reliable.

This produces recurring failures:

- roadmap theater without resourcing or dependency disclosure
- silent removal or reshaping of public commitments
- optimistic AI timelines that misallocate trust (DP11)
- security and privacy work deferred or hidden (DP15)
- milestones dependent on funding that does not exist (DP17)

These failures are structural. Roadmaps coordinate collective action. When they mislead, they coordinate collective harm.

DP16 reframes roadmaps as accountability artifacts: scoped, resourced, dependency-aware, and revisable with memory.

3. Threats and Failure Modes

3.1 Hype-driven timelines

Dates are set for visibility rather than feasibility.

Example: A feature demo launches publicly while required safety systems are not ready.

Why this matters: Capability claims must match operational reality.

3.2 Dependency denial

External constraints are omitted from planning.

Example: Interoperability is promised while critical partner agreements or standards are unresolved.

Why this matters: Honest planning includes what you do not control.

3.3 Moving goalposts without memory

Roadmap changes occur without explanation.

Example: A feature disappears from public documentation without a changelog.

Why this matters: Governance requires traceability of decisions and learning.

3.4 Security and privacy debt hiding

Assurance work is deprioritized in visible planning.

Example: Roadmaps highlight features but omit audit backlogs or known vulnerabilities.

Why this matters: Security posture must be part of the same narrative as feature progress.

3.5 Financial unreality

Milestones assume resources that are not secured.

Example: Public goods infrastructure is promised without funding for maintainers.

Why this matters: Roadmaps must reflect actual capacity, not aspirational intent.

4. Core Principle

Roadmaps and milestones in the meta-layer are honest, resourced, and revisable commitments.

Dependencies, risks, and uncertainties are visible. Security, governance, and safety work are treated as first-class alongside features. Changes are explained with memory of what was learned.

Example: A roadmap includes dependency graphs, funded roles, security milestones, and explicit unknowns, with periodic retrospectives explaining changes.

What this feels like: Respect for participant time, attention, and planning.

Without this: Roadmaps become mechanisms for eroding trust rather than coordinating progress.

5. Primary Mechanisms and Structural Conditions

5.1 Dependency-aware planning

Roadmaps explicitly identify legal, technical, governance, and ecosystem dependencies and classify them by **control and certainty** (controlled, shared, external; verified, assumed, unknown).

This includes identifying **critical-path dependencies** that can block delivery, and attaching owners, verification status, and expected update cadence.

Failure mode: **hidden blockers**, where untracked dependencies invalidate timelines after commitments are made.

5.2 Dual-track visibility

Feature delivery and assurance work (security, privacy, accessibility, safety, governance) are presented together with **linked milestones and shared completion criteria**.

Assurance work is not deferred to “later phases” without explicit gating; features that depend on assurance must reference those gates.

Failure mode: **assurance deferral**, where visible progress masks growing risk debt.

5.3 Milestone receipts

Completed milestones link to verifiable evidence such as documentation, code, tests, audits, or third-party attestations (DP15 alignment).

Receipts should be **durable, accessible, and specific to the claimed capability**, enabling independent inspection.

Failure mode: **receipt theater**, where artifacts exist but do not substantiate the claim.

5.4 Change logs with governance salience

Material changes (scope, timelines, removals, reclassification) are recorded with **who decided, why, and who is affected**.

Changes that impact communities or partners should trigger **targeted notifications** and, where appropriate, governance review (DP3 linkage).

Failure mode: **silent pivots**, where affected parties cannot detect or respond to changes.

5.5 Risk registers

Top risks to roadmap integrity are identified, ranked, and updated with **likelihood, impact, mitigation, and owner**.

Risk registers should distinguish between **delivery risk** (can we build it) and **integration risk** (can it work with others) and be visible alongside milestones.

Failure mode: **risk invisibility**, where known issues are not surfaced to participants.

5.6 Scenario planning

Pre-mortems and alternative scenarios are documented to anticipate failure modes, including **best case, expected case, and worst case** timelines.

Scenarios should be tied to triggers (e.g., dependency slips, funding changes) that cause movement between states.

Failure mode: **single-path planning**, where plans assume ideal conditions and lack adaptation paths.

5.7 AI capability boundaries

Clear distinction between research, prototype, pilot, production, and scaled systems, with **explicit gating criteria and evidence thresholds** (DP11 alignment).

Claims about AI capability must reference **evaluation context, limitations, and deployment constraints**.

Failure mode: **capability inflation**, where demos are presented as operational systems.

5.8 Funding linkage

Milestones are tied to known funding sources, staffing, and maintenance commitments, or clearly labeled as contingent.

Roadmaps should indicate **runway, dependency on grants or partners, and maintenance ownership** for shipped work.

Failure mode: **funding ambiguity**, where delivery is implied without resources.

5.9 Cross-organizational coordination

Shared dependencies across organizations are made visible with **common identifiers, ownership boundaries, and divergence signaling**.

Where coordination is loose, roadmaps should indicate **assumption ranges** rather than fixed dates.

Failure mode: **misaligned expectations**, where partners operate on incompatible timelines.

5.10 Retrospective accountability

Regular reviews compare planned versus actual outcomes, capturing **variance, root causes, dependency shifts, and lessons learned**.

Retrospectives should distinguish between **uncertainty, error, and misrepresentation**, and feed updates back into planning and governance.

Failure mode: **non-learning systems**, where mistakes repeat without structural correction.

5.11 Roadmap System Layer: Commitment Integrity, Dependency Enforcement, and Claim Verifiability

Beyond roadmap artifacts and communication practices, DP16 requires a coherent roadmap system layer that ensures commitments remain traceable, constrained, and resistant to misrepresentation under pressure.

Roadmaps are not only planning documents. They are coordination systems that shape participant expectations, builder effort, funding decisions, governance priorities, and public trust. If roadmap claims cannot be bound to scope, evidence, dependencies, and change history, they become narrative instruments rather than infrastructure.

5.11.1 Commitment integrity and binding

Roadmap entries must represent bounded commitments, not open-ended aspirations.

This requires explicit scope, success criteria, completion conditions, and commitment classification, such as research, exploratory, committed, funded, blocked, or deprecated. Claims must remain proportional to the commitment class.

A failure mode is aspirational drift, where language implies commitment while avoiding operational accountability.

5.11.2 Dependency verification and propagation

Dependencies must not only be listed. They must be validated, monitored, and propagated when they change.

This includes technical dependencies, partner dependencies, legal or regulatory dependencies, funding dependencies, security dependencies, and governance dependencies. Critical-path dependencies should be distinguishable from ordinary dependencies so participants understand what can actually block delivery.

A failure mode is dependency illusion, where plans assume conditions that are unresolved, unavailable, or outside the initiative's control.

5.11.3 Claim-to-outcome traceability

Every roadmap claim must resolve to a delivered artifact, a documented change in status, or an explicit retirement.

This requires persistent identifiers for roadmap items and linkage to code, documentation, audits, governance decisions, release notes, or public explanations. Partial completion should be represented as partial completion, not quietly reframed as success.

A failure mode is untraceable claims, where promises cannot be compared against outcomes.

5.11.4 Change integrity and versioned commitments

Roadmaps must preserve versioned history of commitments and changes.

This includes changelogs, previous states, rationale for material shifts, and explanations for additions, removals, timeline changes, or scope changes. Revision is healthy when it is visible; silent revision is trust decay.

A failure mode is historical erasure, where past commitments disappear without record.

5.11.5 Anti-hype and signaling constraints

Roadmaps must resist distortion from marketing, fundraising, competition, and AI capability theater.

Systems should distinguish clearly between concept, prototype, pilot, production, verified capability, and scaled deployment. Public claims should not exceed operational readiness, and uncertainty should be surfaced rather than hidden.

A failure mode is signal inflation, where external incentives distort internal reality.

5.11.6 Cross-organizational roadmap coherence

When milestones depend on multiple organizations, roadmap meaning must remain coherent across boundaries.

This requires shared milestone references, explicit ownership, divergence signaling, and visibility into where one organization's delay, pivot, or withdrawal affects others.

A failure mode is coordination divergence, where different actors present incompatible versions of shared progress.

5.11.7 Roadmap memory and auditability

Participants must be able to reconstruct what was promised, what changed, what was delivered, and what was learned.

This requires accessible archives, comparison views, retrospective links, and evidence artifacts that allow roadmap integrity to be evaluated over time.

A failure mode is roadmap opacity, where evolution cannot be understood, audited, or contested.

This roadmap system layer ensures that coordination signals remain trustworthy under uncertainty, incentives, and scale rather than degrading into narrative management.

6. Milestone Framework

DP16 requires that milestones be structured objects rather than sentences on a slide. A milestone that cannot be classified, scoped, resourced, and closed is a narrative claim, not a commitment.

6.1 Milestone anatomy

Every milestone carries:

- a persistent identifier that survives reorganization and renaming
- scope (in-scope, out-of-scope) and explicit completion criteria
- commitment class and confidence level
- dependency references with control and certainty classification (5.1)
- assurance gates that must clear before completion (5.2)
- funding state and maintenance ownership (5.8, DP17)
- an accountable owner and a next review date
- evidence links once claimed complete (5.3)

Failure mode: **unbounded milestone**, where completion is a matter of interpretation.

6.2 Commitment classes

Public roadmap language must remain proportional to class:

- **exploratory**: under investigation; no delivery implied
- **committed**: scoped, owned, and resourced; delivery expected within a stated window
- **funded**: committed with secured resources and named maintenance ownership
- **blocked**: committed but held by an identified dependency
- **deprecated / withdrawn**: intentionally stopped, with rationale retained

Failure mode: **class inflation**, where exploratory work is communicated in committed language.

6.3 Status transitions and triggers

Transitions between classes are events, not edits. Each transition records who decided, why, what changed, and who is affected, and where material, triggers notification and governance review (5.4, DP3).

Failure mode: **silent transition**, where status changes without a record.

6.4 Completion and partial completion

Completion requires evidence proportional to the claim. Partial delivery is represented as partial, with the remaining scope carried forward under the same identifier rather than absorbed into a success statement.

Failure mode: **rounding up**, where partial work is reported as delivered.

6.5 Milestone receipts and comparison views

Each closed milestone produces a receipt linking claim to artifacts (docs, code, tests, audits, attestations), and roadmap views allow comparison of committed versus current versus delivered state over time (DP15 linkage).

Failure mode: **unverifiable closure**, where completion cannot be independently checked.

7. Phased Rollout

Roadmap honesty depends on distinguishing stages of maturity. DP16 requires explicit phases with entry and exit criteria so that demonstrations are not mistaken for deployed capability.

7.1 Phase ladder

- **research**: feasibility and risk exploration; no external reliance expected
- **prototype**: demonstrable behavior in controlled conditions; not for dependence
- **pilot**: limited real participants, bounded scope, monitored, reversible
- **limited production**: available with declared constraints and support commitments
- **general availability**: broadly usable with assurance, support, and maintenance in place
- **sustained / maintained**: funded upkeep, security response, and deprecation policy

Failure mode: **phase skipping**, where pilots are marketed as general availability.

7.2 Entry and exit criteria

Movement between phases requires satisfying declared gates, typically including security and privacy review (DP15), safety and containment posture for AI capability (DP11, DP13), governance readiness (DP3), accessibility, operational support, and funding for the next phase (DP17).

Failure mode: **gate erosion**, where criteria are waived under schedule pressure without record.

7.3 Reversibility and rollback

Each phase declares how it can be paused, narrowed, or withdrawn, including participant notification, data handling on withdrawal, and preservation of records.

Failure mode: **irreversible rollout**, where problems cannot be contained without breaking dependents.

7.4 Communication rules per phase

Public claims are bounded by phase. Demonstrations, benchmarks, and previews are labeled with phase, evaluation context, and known limitations, and cannot imply availability that does not exist.

Failure mode: **demo-to-production drift**, where audiences reasonably infer capability that is not deployed.

7.5 Cohort and zone sequencing

Where rollout proceeds by community, zone, or region, sequencing and selection criteria are published, along with what earlier cohorts are expected to absorb in risk and what later cohorts gain.

Failure mode: **invisible experimentation**, where participants do not know they are the pilot.

8. Governance, Accountability, and Agency Surfaces

DP16 requires that roadmap signals are not only visible but **actionable and contestable** within governance processes.

Participants must be able to:

- compare promises to delivered outcomes using stable identifiers and evidence links
- understand which dependencies are blocking or delaying work
- distinguish between committed, contingent, and exploratory milestones
- contest misleading, incomplete, or outdated roadmap representations through defined channels

Communities and governance bodies must be able to:

- align decisions (funding, prioritization, standards) with roadmap reality (DP3 linkage)
- trigger reviews when roadmap integrity degrades (e.g., repeated silent changes, missing receipts)
- require correction, reclassification, or withdrawal of misleading claims
- attach **confidence signals** or integrity ratings to roadmap segments

Systems SHOULD support:

- issue/appeal pathways tied to specific roadmap items
- public commentary or annotation layers for milestones
- escalation mechanisms for high-impact discrepancies

Failure modes include:

- **non-actionable transparency**, where information is visible but cannot be used to influence outcomes
- **accountability gaps**, where no actor is responsible for correcting misleading signals
- **participation fatigue**, where communities lack effective recourse and disengage

9. Incentives and Power Analysis

Roadmaps sit inside powerful incentive fields. Teams, funders, partners, contributors, and communities often benefit in the short term from optimistic claims, compressed timelines, and selective disclosure of risk. DP16 exists because those incentives do not naturally produce truth.

Communication incentives often reward optimism over accuracy. A roadmap that looks confident can attract funding, attention, talent, and legitimacy even when the underlying capacity is uncertain. This creates pressure to convert aspiration into implied commitment and ambiguity into promotional momentum.

DP16 requires aligning incentives with truthful forecasting, postmortem transparency, and long-term credibility rather than short-term attention.

Key incentive risks include:

- overstating readiness to attract funding, partners, or adoption
- hiding security, privacy, or governance debt to maintain confidence
- framing dependent milestones as controlled milestones
- using AI capability claims to create urgency without operational evidence
- shifting definitions of success after work has begun

Roadmap integrity therefore requires consequences for repeated misrepresentation. These may include governance review, downgraded confidence ratings, public correction receipts, funding conditions, or reduced eligibility for ecosystem support.

A critical failure mode is credibility arbitrage, where actors gain near-term benefits from inflated claims while distributing long-term costs to contributors and communities.

10. Community Signals Informing DP16

Community signals consistently show that people do not expect perfect prediction. They expect honesty, memory, and respect for the planning burdens created by public commitments.

Recurring signals include:

- frustration with shifting timelines without explanation
- demand for security and safety work to be visible
- skepticism of AI claims in public communication
- desire for funded maintenance commitments
- concern that public promises are used to recruit labor or attention without durable obligation
- need for clear distinctions between pilots, prototypes, production systems, and aspirational futures

These signals point to a deeper pattern: participants can tolerate uncertainty when it is named, but lose trust when uncertainty is converted into certainty for strategic advantage.

DP16 treats these signals as design inputs. Roadmap systems should make uncertainty legible, show dependency reality, and allow communities to distinguish delay caused by honest learning from delay caused by misrepresentation.

11. Evaluation Criteria

Roadmap integrity is measurable. DP16 evaluates whether signals were reliable, not whether predictions were perfect.

11.1 Forecast and variance measures

- **variance distribution:** planned versus actual delivery windows, reported without post-hoc rebasing
- **explained variance ratio:** share of slips accompanied by published rationale
- **re-forecast timeliness:** interval between a dependency change and public timeline update

11.2 Dependency and resourcing accuracy

- **dependency disclosure rate:** share of blocking dependencies identified before commitment
- **control classification accuracy:** how often "controlled" dependencies proved external
- **funding-label accuracy:** how often committed work was actually resourced (DP17)

11.3 Evidence and memory measures

- **receipt coverage:** share of completed milestones with verifiable evidence links
- **changelog completeness:** share of material changes with recorded author, rationale, and impact
- **history integrity:** absence of silent removals; prior states remain retrievable

11.4 Assurance visibility

- **assurance ratio:** proportion of roadmap surface allocated to security, privacy, safety, and accessibility work
- **gate adherence:** share of phase transitions that satisfied declared criteria
- **capability claim accuracy:** alignment between stated AI readiness and evaluated capability (DP11)

11.5 Thresholds, triggers, and ratings

Evaluation must be consequential:

- integrity ratings attach to roadmap segments and are visible alongside milestones (Section 6, Section 8)
- breaching thresholds (repeated silent changes, missing receipts, sustained unexplained variance) triggers governance review
- correction, reclassification, or withdrawal is required where claims were misleading

Failure modes:

- **accuracy scoring that punishes honesty**, discouraging early disclosure of uncertainty
- **metric gaming** through milestone splitting or scope shrinkage
- **unowned measurement**, where results are published but change nothing

12. Implementation Patterns

These patterns make roadmap integrity operational rather than aspirational. They are illustrative, not mandated.

12.1 Roadmap item registry

A structured store where each milestone is a record with stable identifier, class, scope, dependencies, assurance gates, funding state, owner, and evidence links. Public views render from the registry, so presentation cannot drift from the underlying record.

12.2 Dependency graph with propagation

Dependencies are modeled as edges with control and certainty attributes. When a dependency changes state, downstream milestones are flagged automatically for re-forecast rather than silently retaining old dates.

12.3 Dual-track board

A single view presents feature work and assurance work with shared status vocabulary and linked gates, making deferral of security, privacy, or accessibility visible instead of invisible.

12.4 Milestone receipt store

Completed milestones publish receipts referencing documentation, releases, test results, audits, or attestations (DP15), with permanent addresses so future readers can verify past claims.

12.5 Change feed and targeted notification

Material roadmap changes emit an append-only public feed entry and notify affected communities, partners, and dependents, converting silent pivots into observable events.

12.6 Confidence and integrity badges

Roadmap segments display confidence level, funding state, and integrity rating derived from measured history, giving readers a calibration signal at the point of reading.

12.7 Retrospective and pre-mortem templates

Standard formats capture planned versus actual, dependency shifts, distinctions between uncertainty, error, and misrepresentation, and the specific planning changes adopted in response.

12.8 Public status surface

A durable status page maps current phase, known blockers, open risks, and next review dates, so participants do not need to reconstruct state from announcements.

Anti-patterns to avoid:

- roadmaps maintained only in slides or marketing pages
- status conveyed exclusively through release announcements
- private trackers whose public rendering is manually curated
- date changes applied by editing history rather than recording transitions

13. Non-Goals and Explicit Boundaries

DP16 does not aim to eliminate uncertainty or impose rigid planning. It defines what roadmaps are **not allowed to become**.

DP16 does not:

- guarantee precise timelines in uncertain environments; variance is expected when it is explained and bounded
- eliminate the need for confidentiality; sensitive details may be abstracted, but their existence and impact should be signaled
- replace investor, partner, or internal planning norms; it constrains how those norms translate into public commitments
- mandate a single methodology; agile, waterfall, and hybrid approaches are compatible if integrity conditions are met

DP16 also does not permit:

- using ambiguity to imply commitment where none exists
- presenting dependent or unfunded work as controlled delivery
- omitting assurance work to improve perceived velocity
- rewriting history to preserve narrative coherence

Boundary principle:

Roadmaps may be uncertain, but they must not be misleading about what is known, what is controlled, and what is promised.

Failure modes include:

- **methodology masking**, where process language obscures lack of commitment
- **selective disclosure**, where only favorable information is surfaced
- **narrative protection**, where truth is suppressed to maintain external perception

14. Minimum DP16 Alignment (Non-Normative)

Minimum alignment is the point at which roadmap signals are **reliable enough to coordinate real work**. The aim is not perfection, but to avoid systematically misleading participants.

A DP16-aligned initiative should, at minimum:

- **Bind milestones to scope and outcomes:** Each milestone clearly states what is in/out of scope and what counts as completion (e.g., shipped artifact, audit, or documented capability).
- **Make dependencies legible:** Key dependencies (technical, legal, ecosystem) are identified, with notes on what is controlled vs. external and what is uncertain.
- **Show assurance alongside features:** Security, privacy, and safety work appears in the same roadmap view as features (not deferred or hidden).
- **Provide evidence for completion:** Completed milestones link to verifiable artifacts (docs, code, tests, audits) where appropriate.
- **Maintain change memory:** Material changes (scope, dates, removals) are recorded with brief explanations; prior states are not silently erased.
- **Signal uncertainty honestly:** Distinguish research, prototype, and production readiness (DP11), and avoid presenting demonstrations as deployed capability.
- **Link to resourcing reality:** Indicate whether milestones are funded, partially funded, or contingent (DP17), and avoid presenting unfunded work as committed delivery.
- **Enable comparison over time:** Participants can compare earlier commitments to current status without reconstructing history.

Failure modes to avoid:

- **Aspirational drift:** commitments implied without binding scope or resourcing
- **Dependency illusion:** plans rely on unresolved external conditions without disclosure
- **Assurance invisibility:** security/safety work omitted from visible planning
- **Historical erasure:** past commitments disappear without record
- **Signal inflation:** capability claims exceed operational reality

15. Open Questions and Future Work

DP16 leaves several questions open because roadmap integrity depends on context, maturity, and governance capacity. The goal is not to impose one planning methodology, but to define the integrity conditions that any credible method must satisfy.

Key open questions include:

- **Transparency versus security:** How should initiatives disclose roadmap risk without revealing vulnerabilities, attack windows, or sensitive partner dependencies?
- **Multi-organization alignment:** What shared formats or receipts allow independent organizations to coordinate milestones without creating centralized control?
- **Forecast accuracy measurement:** How can communities measure forecasting reliability without punishing honest learning, adaptation, or uncertainty disclosure?
- **Regulatory and legal timelines:** How should legal uncertainty, policy change, and compliance review be represented in public roadmap systems?
- **AI capability claims:** What evidence thresholds should distinguish research demos, controlled pilots, production deployments, and frontier capability claims?
- **Funding-contingent work:** How should public goods work be represented when it is necessary but not yet funded?
- **Community recourse:** What should participants be able to do when roadmaps repeatedly mislead or coordination harms accumulate?

These questions should mature through future ML-Drafts, reference implementations, and governance experiments. DP16 should not freeze roadmap practice too early; it should make roadmap integrity inspectable while methods evolve.

16. Relationship to Other Desirable Properties

DP16 is a cross-cutting property because roadmap integrity affects whether other desirable properties can be trusted over time.

- **DP3: Adaptive Governance.** Governance needs accurate roadmap signals to sequence decisions, allocate attention, and adjust priorities. If roadmap changes are silent, governance becomes reactive and legitimacy suffers.
- **DP7: Interoperability.** Interoperability depends on shared timelines, version expectations, and dependency clarity. Roadmap drift across systems can break integration even when protocols are sound.
- **DP9–DP10: Builder and learner expectations.** Contributors and learners invest time based on public signals. Roadmap integrity protects that investment from being misdirected by hype or vague commitments.

- **DP11–DP15: AI, security, and provenance alignment.** Capability claims, safety gates, audits, provenance systems, and security posture all require visible sequencing. DP16 ensures assurance work is not hidden behind feature marketing.
- **DP17: Financial sustainability.** Milestones must reflect funding and maintenance reality. Unfunded commitments should be labeled as contingent rather than presented as guaranteed delivery.

If DP16 fails, other DPs can appear stronger than they are. A system may claim future privacy, future AI safety, future security, or future interoperability while coordinating present action around promises that are not structurally accountable.

17. Foresight and Failure Design

DP16 assumes delays, failures, and shifting conditions. It does not treat change as failure. It treats untracked, unexplained, or strategically obscured change as failure.

Common roadmap failure paths include:

- teams overcommitting to preserve confidence
- security and governance work falling behind visible features
- funding assumptions becoming invalid without public relabeling
- AI capability claims spreading faster than verification
- external dependencies blocking milestones while public timelines remain unchanged

These failures often compound. A single optimistic milestone can attract contributors, partners, and community expectations. If it later shifts without explanation, trust damage spreads beyond the missed milestone and affects the perceived legitimacy of the broader initiative.

DP16 therefore requires recovery practices:

- public status changes with explanation
- retrospectives that distinguish error, uncertainty, and misrepresentation
- updated dependency maps after delays
- repair pathways for communities or builders harmed by misleading signals
- governance review for repeated roadmap integrity failures

A mature roadmap system does not avoid failure. It makes failure learnable, bounded, and repairable.

18. Path Toward ML-RFC

Advancement from ML-Draft to ML-RFC requires demonstrating that roadmap integrity can be operationalized across real initiatives, not merely described.

Key progression steps include:

- **Standardize roadmap artifact formats:** Define common fields for scope, status, dependency class, funding state, assurance gates, confidence level, and evidence links.
- **Develop milestone receipt patterns:** Establish how completed, delayed, blocked, or retired milestones produce verifiable records.
- **Create retrospective practices:** Define minimum retrospective contents, including planned versus actual outcomes, dependency shifts, and lessons learned.
- **Align communication policies with trust frameworks:** Ensure public claims, demos, and announcements remain consistent with roadmap status and operational readiness.
- **Test multi-organization coordination:** Demonstrate shared roadmap references across independent teams with explicit divergence signaling.
- **Define conformance checks:** Create tests for aspirational drift, dependency illusion, historical erasure, signal inflation, and assurance invisibility.

Promotion to ML-RFC should require evidence that participants can compare commitments over time, verify milestone outcomes, and understand why material changes occurred.

19. Closing Orientation

DP16 is where the meta-layer demonstrates respect for time.

Participants, builders, funders, and communities make real decisions based on roadmap signals. They allocate labor, money, trust, attention, and hope. When those signals are inflated, incomplete, or silently rewritten, the harm is not merely reputational. It is coordination harm.

When roadmaps are honest, participants can coordinate effort and trust direction. They can understand what is real, what is uncertain, what is blocked, and what has changed.

When roadmaps are not honest, trust erodes faster than any feature can rebuild it. Even successful delivery becomes suspect when the path to delivery cannot be reconstructed.

DP16 is the claim that the meta-layer will not build the future on vapor, hidden debt, or erased commitments.

Roadmaps should not be marketing fiction. They should be public instruments of shared orientation, accountable learning, and coordinated care.
