

The Doctrine Development Cycle

The Doctrine Development Cycle is the recurring process through which Malone Doctrine is created, tested, corrected, adopted, applied, and improved. It treats doctrine as a controlled system of institutional engineering rather than a collection of permanent first impressions.

Doctrine must be stable enough to govern, but not so rigid that error becomes sacred.

The cycle begins with a real problem and continues through implementation. Experience generated by implementation then returns to the drafting process, where weaknesses, conflicts, omissions, and unintended consequences are examined. The revised doctrine is tested again before a new version is adopted.

The working cycle is:

**Identify → Develop → Derive → Reconcile → Test → Adopt → Implement → Evaluate →
Revise**

This cycle may repeat several times before a provision is mature.

Identify

Development begins with a consequential problem, conflict, failure, or recurring condition requiring correction.

The problem may arise from:

- lived experience;
- institutional operations;
- community testimony;
- historical research;
- political developments;
- failures of leadership;
- conflicts among existing provisions;
- defects revealed during implementation.

The first task is to define the issue accurately. A poorly identified problem produces irrelevant rules, excessive restrictions, or solutions directed at symptoms rather than causes.

The problem should be placed within the appropriate doctrinal jurisdiction and strategic container.

Develop

The issue is then examined through problem-driven doctrinal engineering.

The drafter identifies:

- the operative mechanism;
- the persons and institutions affected;
- the harm produced;
- the required condition;
- the individual response;
- the institutional response;
- the resources and authority needed;
- the measures that would indicate improvement.

This stage may begin as an article, case analysis, proposed rule, procedure, or implementation plan. The objective is not to force the issue immediately into final doctrinal language. The objective is to understand the problem well enough to design a workable correction.

Derive

Once the response becomes clear, its normative content is derived across the authority hierarchy:

1. axioms;
2. principles;
3. maxims;
4. rules;
5. procedures;
6. authorized commentary and interpretations;
7. articles and applications;
8. implementation documents.

The layers must remain connected.

A rule should implement a principle.

A principle should express a foundational axiom.

A procedure should direct performance of the rule.

Commentary should explain the authorized meaning without replacing the governing text.

Development may move upward or downward through the hierarchy. A concrete rule may reveal the need for a broader principle. A proposed axiom may prove too broad when tested against specific applications. Drafting order is therefore flexible even though authority order remains fixed.

Reconcile

New provisions must be compared with existing Doctrine.

Reconciliation asks:

- Does the new provision contradict a higher authority?
- Does it duplicate an existing rule?

- Does it create conflicting duties?
- Does it exceed its proper jurisdiction?
- Can apparently conflicting provisions be harmonized?
- Which provision controls when harmonization is impossible?
- Does the new language alter the meaning of previously adopted Doctrine?

Where provisions cannot be harmonized, the higher-ranking authority controls.

Reconciliation protects the Doctrine from becoming a series of independent documents that give incompatible instructions. It also reveals when an earlier provision must be revised rather than silently ignored.

Test

A provision must be tested before adoption.

Testing should include:

- ordinary applications;
- difficult cases;
- edge conditions;
- foreseeable misuse;
- hostile interpretation;
- institutional capacity;
- enforcement requirements;
- financial and administrative burden;
- unintended consequences;
- consistency across the four ethical domains;
- effect upon succession and long-term governance.

The drafter should ask not only whether the provision produces a desirable result in the case that inspired it, but whether it remains just and useful when applied elsewhere.

A rule created for one sympathetic case may become dangerous when used by an incompetent or hostile authority. An axiom that sounds morally compelling may fail because it cannot be applied universally. A procedure may be correct in theory but impossible to administer.

Testing exposes these defects before they become institutional commands.

Adopt

A draft becomes Doctrine only through authorized adoption.

Adoption should establish:

- final text;
- authority level;

- document status;
- effective date;
- version number;
- jurisdiction;
- adopting authority;
- relationship to superseded provisions;
- placement within the applicable regulation schedule.

This distinction protects the boundary between research, proposal, interpretation, and binding authority.

A draft may be persuasive without governing.

A Primer essay may be informative without governing.

A proposed rule may be practical without governing.

Only adopted Doctrine governs.

Implement

Adoption is not completion. The provision must be translated into actual conduct.

Implementation may require:

- training;
- forms;
- checklists;
- appointment of responsible officers;
- funding;
- reporting systems;
- disciplinary procedures;
- communications;
- institutional coordination;
- public explanation;
- local adaptation.

This stage determines whether the Doctrine can function outside the page.

A provision that cannot be implemented must be redesigned, narrowed, postponed, or classified as aspirational guidance rather than falsely presented as an operational command.

Evaluate

Implementation creates new evidence.

The responsible authority should determine:

- Was the provision understood?
- Was it followed?
- Did it correct the intended problem?

- Did it create new problems?
- Were responsibilities assigned properly?
- Did people evade or manipulate the rule?
- Were the measurements adequate?
- Did implementation differ across institutions or locations?
- Is the problem declining, changing, or reappearing in another form?

Evaluation should consider documented evidence, circumstantial patterns, testimony, operational reports, and measurable outcomes.

The purpose is not to defend the original draft. It is to determine whether the Doctrine works.

Revise

Where evaluation reveals weakness, the provision returns to development.

Revision may:

- clarify ambiguous language;
- narrow or expand jurisdiction;
- correct an error;
- add an exception;
- strengthen an enforcement mechanism;
- revise a procedure;
- reconcile a conflict;
- replace an ineffective response;
- elevate a recurring rule into a broader principle;
- demote language that cannot support the authority previously assigned to it.

Revision must be documented through version control. Earlier language should be preserved for provenance when historically significant, while the current adopted version remains clearly identifiable.

The cycle then begins again through reconciliation, testing, adoption, and implementation.

Controlled Continuity

The Doctrine Development Cycle protects two competing necessities: continuity and correction.

Continuity prevents every new leader, faction, political trend, or crisis from rewriting the governing code according to temporary preference. Correction prevents inherited language from controlling merely because it was written first.

The proper standard is controlled continuity:

Doctrine remains authoritative until lawfully revised, but no provision is protected from examination when evidence shows that it is defective, contradictory, impractical, or harmful.

The cycle also distinguishes development from instability. Frequent unrecorded changes weaken authority. Formal, reasoned, version-controlled revision strengthens it because the institution can explain what changed, why it changed, and which text now governs.

Institutional Memory

Every cycle should produce a permanent record containing:

- the original problem;
- evidence and testimony considered;
- competing proposals;
- reasons for the adopted response;
- tests performed;
- implementation results;
- discovered failures;
- revision history;
- current controlling language.

This record allows successors to understand both the command and the reasoning behind it. It prevents old errors from being repeated and useful lessons from disappearing.

The governing principle is:

Malone Doctrine shall be developed through recurring cycles of problem identification, derivation, reconciliation, testing, adoption, implementation, evaluation, and controlled revision.

The Doctrine Development Cycle ensures that Malone Doctrine remains authoritative without becoming stagnant, adaptable without becoming arbitrary, and permanent without becoming incapable of learning. It turns doctrine into a living institutional system whose continuity is preserved through disciplined correction rather than resistance to change.