



CREF 3.1

Civic Record Evaluation Framework

A Deterministic Methodology for Structured Civic Record Evaluation

Version 3.1 — Revised Edition

Incorporates the Forty-Stage Summary, a worked evaluation example, and a current implementation status note.

Colophon

This publication describes CREF Version 3.1, a methodological specification.

It was published alongside the Civic Decision Engine (CDE), build v12 Stage 1, its reference implementation.

The methodology version and the implementation build version are independent identifiers and do not correspond to one another.

See Chapter 19 for current implementation status and Chapter 22 for the versioning principle.

The institutional seal on the title page identifies CDE as steward and publisher; it carries no version number by design.

Revision Note — 3.0 to 3.1

Version 3.1 revises Version 3.0 in four respects, each raised during review of the completed specification:

- Appendix A now enumerates all forty declared stages by name and function, rather than asserting the count without a visible list.
- A new chapter, "Worked Example," walks a single record through evaluation, inspection, governance, and preservation, grounding the framework's abstract behaviours in one concrete pathway.
- Boundary statements throughout Part III have been condensed to the limitations that are load-bearing for each chapter, rather than a repeated full enumeration of what the framework does not do.
- Chapter 19 (Reference Implementation) now includes an Implementation Status note distinguishing the completed methodology from the current build state of its reference implementation, the Civic Decision Engine.

No methodological behaviour declared in Version 3.0 has been altered. Version 3.1 clarifies and completes the specification; it does not extend the framework's scope.

Preface

CREF (Civic Record Evaluation Framework) is a deterministic methodology for evaluating civic records through declared evidence structures, transparent inspection, and governed methodology.

Version 3.1 documents the complete forty-stage framework as a self-contained methodological specification, with the stage list, a worked example, and implementation status made explicit.

The framework distinguishes evidence from interpretation, evaluation from authority, and methodology from implementation. Throughout the framework, every evaluation is derived from declared rules, visible metadata, and inspectable outputs.

CREF does not determine truth, legal liability, or institutional authority. It provides a structured methodology for examining civic records within explicitly declared boundaries.

Version 3.1 represents the completed methodology implemented through Stages 1–40, revised for clarity and completeness.

Part I — Foundations

Chapter 1. Introduction

1.1 Why CREF Exists

Modern public administration depends upon records — decisions, communications, evidence, procedures, responses. These records form the observable basis upon which civic accountability can be examined.

While institutions frequently maintain extensive records, methodologies for evaluating those records often remain implicit, inconsistent, or dependent upon subjective interpretation. CREF was developed to provide a deterministic methodology for evaluating civic records through declared structures rather than undocumented judgement.

Its objective is not to determine whether a decision was correct. Its objective is to make the evaluation process itself inspectable.

1.2 Methodology Character

CREF is a methodology. It is not software, legislation, legal procedure, or institutional policy. Software may implement CREF; organisations may adopt it; researchers may analyse it. None of these implementations alter the methodology itself, which exists independently of any individual implementation.

1.3 Deterministic Evaluation

Every output produced by CREF derives from declared methodology rather than undocumented reasoning. Each evaluation follows declared stages, visible metadata, explicit dependencies, and inspectable outputs, allowing independent observers to examine both an outcome and the pathway that produced it.

Grounding note: in the reference implementation, determinism is not merely asserted — it is achieved mechanically, through SHA-256 content hashing of evidence and outputs and through immutable, path-based storage of preserved artefacts. Chapter 18 walks through this concretely.

Chapter 2. Scope

CREF applies to the evaluation of civic records — documents, correspondence, administrative records, public decisions, policies, reports, metadata, and structured record collections — where the goal is a structured, inspectable evaluation rather than a determination of truth, liability, or authority.

CREF does not itself decide legal questions, certify institutional compliance, or validate the authenticity of evidence submitted to it. These remain outside the methodology's scope in every stage.

Chapter 3. Definitions

- Evidence — observable material presented to the evaluation process without alteration.
- Evaluation — the deterministic application of declared rules to a declared evidence set.
- Inspection — the transformation of evaluation outputs into inspectable reports.
- Governance — the maintenance of the methodology's own structure, continuity, and boundaries.
- Preservation — the maintenance of methodological artefacts for future inspection without reinterpretation.
- Reflection — the framework's description of its own character using its own declared outputs.
- Completion — the formal declaration that an implemented version's stage sequence is finished.
- Implementation — software applying CREF's principles; distinct from the methodology itself.

Chapter 4. Design Principles

- Determinism: identical evidence and rules produce identical outputs.
- Separation of evidence and evaluation: evidence is never altered by the act of evaluating it.
- Visibility of dependency: nothing an output relies on is hidden from inspection.
- Boundary discipline: the framework states, and keeps to, what it does not do.
- Version independence: each published version is a complete, self-contained specification.

Chapter 5. Methodological Boundaries

CREF does not determine truth, legal liability, institutional authority, or factual correctness. It does not validate evidence authenticity, resolve conflicting evidence, or infer missing evidence. These boundaries apply across every stage and are restated, where load-bearing, in Part III.

Part II — Framework Architecture

Chapter 6. Overall Framework Architecture

CREF is organised into three phases comprising forty declared stages:

- Phase I — Evaluation Methodology (Stages 1–19): how evidence becomes a sealed evaluation output.
- Phase II — Reflexive Inspection Infrastructure (Stages 20–30): how sealed outputs become inspectable, replayable reports.
- Phase III — Methodology Governance (Stages 31–40): how the methodology itself is kept continuous, transparent, and eventually declared complete.

The full stage-by-stage list appears in Appendix A. Chapter 10 gives a condensed overview of how the three phases connect.

Chapter 7. Phase I — Evaluation Methodology (Stages 1–19)

Phase I governs the path from raw evidence intake to a sealed, boundary-stated evaluation output. It establishes identity, boundary, relationship, and preservation of evidence before any rule is applied, then binds a versioned rule set, applies it deterministically, and seals the result against silent modification.

Chapter 8. Phase II — Reflexive Inspection Infrastructure (Stages 20–30)

Phase II governs how a sealed Phase I output becomes something a third party can examine. It declares report modes, exposes dependencies and provenance, and specifies a replay mechanism so that inspection is not a claim of trustworthiness but a demonstrable, repeatable procedure.

Chapter 9. Phase III — Methodology Governance (Stages 31–40)

Phase III governs the framework rather than any single record. It establishes continuity across versions, a change register, a stewardship declaration, a preservation architecture for the methodology's own artefacts, a meta-framework reflection (Stage 39), and the formal completion statement (Stage 40).

Chapter 10. Forty-Stage Framework Overview

The table below summarises the three phases; full detail is in Appendix A.

Stage	Name	Function
1–19	Phase I — Evaluation Methodology	Evidence intake through sealed evaluation output.
20–30	Phase II — Reflexive Inspection Infrastructure	Report modes, provenance, and deterministic replay.
31–40	Phase III — Methodology Governance	Continuity, stewardship, reflection, and completion.

Part III — Methodological Behaviour

Chapter 11. Deterministic Evaluation

Every output produced by CREF derives from declared methodology. Given identical evidence and an identical bound rule set, evaluation is reproducible without exception.

Load-bearing boundary: determinism describes the evaluation pathway, not the correctness of the evidence it is applied to. CREF does not certify that the record evaluated is itself accurate.

Chapter 12. Evidence Handling

CREF evaluates records through declared evidence structures. It defines how evidence is identified, preserved, related, inspected, and referenced — it does not determine whether evidence is true or false, and it does not reconstruct missing material.

Evidence remains separate from evaluation: the methodology does not alter, annotate, or replace source material. Framework outputs reference evidence; they do not become it. Relationships between evidence items (sequence, dependency, provenance, citation) are declared explicitly and never inferred silently.

Load-bearing boundary: CREF does not validate authenticity, resolve conflicting evidence, or create evidence. Its purpose is a deterministic structure through which evidence may be examined, not a verdict on the evidence itself.

Chapter 13. Inspection Behaviour

Inspection transforms structured evidence into inspectable methodology outputs and is deterministic, reproducible, bounded, and non-mutating. Report presentation may differ across report modes; evaluation behaviour underneath does not.

Inspection exposes declared dependencies and preserves provenance for every output, and supports deterministic replay: given identical evidence and methodology, replay reproduces the same evaluation pathway, demonstrating methodological consistency rather than evidence correctness.

Load-bearing boundary: inspection does not decide legal questions, determine liability or intent, or validate evidence authenticity. It makes methodology visible; interpretation remains outside the framework.

Chapter 14. Governance Behaviour

Governance ensures the methodology itself remains inspectable throughout its lifecycle. Unlike evaluation, it does not assess records submitted for inspection — it evaluates and preserves the structure, continuity, integrity, and declared boundaries of the methodology, operating at the framework level rather than the case level.

Governance rests on five principles: continuity (an unbroken methodological sequence across versions), transparency (governance outputs are explicitly documented, not hidden in implementation), traceability (changes remain attributable to declared revisions), stewardship (preservation rather than ownership), and boundary preservation (scope cannot expand by implication).

Load-bearing boundary: governance does not create legal authority, approve adoption, or certify institutional compliance. Its function is to preserve methodological continuity, nothing more.

Chapter 15. Preservation Behaviour

Preservation ensures methodological artefacts remain available for future inspection without modifying their original meaning. It applies to implemented stages, declared methodology, governance outputs, inspection structures, and framework metadata, and preserves the structural relationships between these components — phase relationships, stage sequence, dependency structure, and lineage.

Legacy preservation documents methodology independently of software implementation, so the framework remains understandable even if a given implementation evolves or ceases operation.

Load-bearing boundary: preservation does not guarantee permanent storage or future hosting, and does not certify software portability. It maintains declared methodology only.

Chapter 16. Reflection Behaviour

Reflection enables the framework to describe its own methodological character using its declared outputs, operating on visible framework artefacts rather than external interpretation. It is deterministic, evidence-derived, bounded, and introduces no additional evidence.

Stage 39 introduced a formal reflection layer describing CREF as a methodology: identifying its character, documenting the separation from implementation, recording adjacent-domain relationships, and preserving declared boundaries. Reflection documents what the framework demonstrably is — it does not speculate about what it may become.

Load-bearing boundary: reflection does not evaluate external methodologies, certify research quality, or infer undocumented characteristics. It describes; it does not validate.

Chapter 17. Completion Behaviour

Completion formally declares the implemented framework complete within its declared scope — the final methodological output of the implemented framework lifecycle. Within Version 3.1, completion confirms that all forty declared stages have been implemented, the three phases are complete, and governance, preservation, and reflection architectures are complete as methodology.

Completion applies only to the version it is declared for. Future versions may extend the methodology without altering the declared completion of a prior version; each completed version remains independently inspectable.

Load-bearing boundary: completion does not certify adoption, validate external implementations, or determine framework superiority. It documents the implemented methodology, nothing more.

Chapter 18. Worked Example

This chapter follows one record through the framework, end to end, to ground Chapters 11–17 in a concrete pathway rather than abstract description. The example uses the vocabulary of the reference implementation, the Civic Decision Engine (CDE), without depending on it — an alternative implementation would produce the same methodological pathway through different mechanics.

Step 1 — Evidence Intake and Boundary (Stages 1–5)

A public decision document is submitted for evaluation. It is assigned a stable identifier, and its declared boundary states explicitly that only the submitted document — not any related but unsubmitted correspondence — is within scope. A SHA-256 hash is computed over the document's byte content at intake; this hash becomes the evidence's fixed reference point for every later stage.

Step 2 — Deterministic Evaluation (Stages 6–19)

The record is classified, bound to a specific version of the evaluation rule set, and evaluated mechanically against that rule set. The output — for example, a structured finding about which declared procedural steps are documented in the record — is generated, stamped with the rule-set version and timestamp, and linked back to the intake hash. A non-mutation check confirms the source document's hash is unchanged, and the evaluation record is sealed: any future alteration to the sealed output is itself detectable, because it would change the output's own hash.

Step 3 — Inspection (Stages 20–30)

A reviewer requests the public report mode. The report exposes the evaluation's dependencies — which rule-set version, which evidence hash, which prior stage outputs it relied on — and its provenance. The reviewer, or an independent auditor, can request replay: re-running the same rule set against the same evidence hash and confirming the output matches. A match demonstrates the evaluation pathway is reproducible; it does not, on its own, establish that the original document's contents were accurate.

Step 4 — Governance and Preservation (Stages 31–40)

Separately from the record's own evaluation, the framework's governance layer logs that this evaluation was produced under rule-set version N, and that version N is part of an unbroken lineage back to version 1. If the rule set is later revised, the change register documents the revision; the sealed evaluation above remains valid under the version it was produced with, and is not silently re-evaluated. Preservation ensures that even if the reference implementation's storage layer changes, the evaluation's metadata, hash linkages, and governance record remain reconstructable.

What this example does and does not show: it demonstrates how determinism, provenance, and replay become concrete via hashing and sealed metadata. It does not demonstrate legal validity, evidentiary weight, or institutional acceptance of the underlying document — those remain, by design, outside CREF's scope.

Part IV — Reference Implementation

Chapter 19. Reference Implementation

CREF is a methodology; methodologies require implementations before they can be applied to records. The Civic Decision Engine (CDE) serves as the reference implementation of Version 3.1, demonstrating how the methodology may be realised through deterministic software while preserving its declared boundaries. The implementation illustrates the methodology; it does not define it.

The reference implementation demonstrates deterministic evaluation, structured evidence handling, inspectable outputs, reproducible behaviour, explicit governance, preservation architecture, methodological reflection, and completion declaration — each derived from the methodology rather than being implementation-specific.

Implementation Status Note

This specification describes CREF's methodology as complete through Stage 40. The reference implementation's build state is a separate fact and, as of this revision, is not at parity with the full methodology: CDE has completed v12 Stage 1, which introduced file-attachment support (the `record_attachments` schema, migration, SHA-256 hashing, and immutable storage path helpers). Stage 2 of that build — the admin upload route and its authentication mechanism (environment-variable bearer token versus network-level protection) — remains an open decision and is not yet implemented. Readers comparing this specification against the live deployment should treat governance, preservation, and reflection behaviours described here as methodologically specified, not as claims about every feature currently operating in production.

Chapter 20. Methodological Conformance

Conformance concerns methodological behaviour, not technical architecture. A conforming implementation should demonstrate deterministic evaluation, declared evidence boundaries, inspectable outputs, explicit limitations, replay capability, provenance preservation, governance visibility, preservation behaviour, reflection behaviour, and completion behaviour.

Implementations may vary in interface, storage mechanism, deployment environment, programming language, authentication model, and presentation style without affecting conformance. An implementation is not conformant if it alters declared boundaries, removes determinism, suppresses provenance or replay, introduces undocumented inference, mutates evidence, or obscures governance outputs.

This specification does not certify implementations; it specifies the characteristics against which implementations may be examined. Certification, if performed, remains external to CREF.

Chapter 21. Adoption

CREF is designed as a portable methodology; adoption does not require use of the Civic Decision Engine. Adopters should preserve declared methodology, deterministic behaviour, evidence boundaries, governance architecture, preservation architecture, and explicit limitations. Local adaptation — terminology, interface, workflow, jurisdiction-specific metadata — is possible provided it does not alter declared methodological behaviour.

Adoption does not imply legal compliance, regulatory approval, institutional endorsement, or certification. It signifies only that an organisation has chosen to implement the methodology.

Chapter 22. Versioning

Each published version of CREF is a complete methodological specification with a unique identifier, publication date, declared scope, declared boundaries, documented changes, and preserved lineage. Future versions do not modify prior specifications; historical versions remain valid records of their declared methodology. Version numbers identify methodology, not software releases — implementations may release independently while remaining methodologically conformant.

Part V — Supporting Material

Glossary

- Bounded — limited to an explicitly declared scope.
- Deterministic — producing identical outputs from identical inputs and rules.
- Non-mutating — an operation that does not alter its source material.
- Provenance — a record of how an output originated.
- Replay — re-execution of a methodology pathway to verify reproducibility.
- Sealed — closed against further silent modification.

References

CREF is a self-contained specification. Where implementations reference external standards (e.g. cryptographic hash functions), those standards are referenced by name only and are not reproduced here.

Version History

- Version 3.0 — Complete forty-stage methodology, documented across Stages 1–40.
- Version 3.1 — Adds the full Appendix A stage list, Chapter 18 (Worked Example), condensed boundary statements in Part III, and an Implementation Status note in Chapter 19.

Appendix A — Forty-Stage Summary

Phase I — Evaluation Methodology (Stages 1–19)

Stage	Name	Function
1	Evidence Intake Declaration	Declares how evidence enters the framework and what form it must take.
2	Evidence Identification	Assigns stable identifiers to discrete evidence items.
3	Evidence Boundary Declaration	States explicitly what the evidence set does and does not include.
4	Evidence Relationship Mapping	Records declared relationships (sequence, dependency, citation) between items.
5	Evidence Preservation Lock	Fixes the evidence set against modification once evaluation begins.
6	Record Classification	Assigns the record to a declared evaluation category.
7	Declared Rule Set Binding	Binds the evaluation to a specific, versioned rule set.
8	Deterministic Rule Application	Applies the bound rules mechanically to the evidence set.
9	Dependency Declaration	Records which prior outputs or evidence the evaluation depends on.
10	Evaluation Pathway Construction	Assembles the ordered sequence of steps that produced the output.
11	Evaluation Output Generation	Produces the primary evaluation output.

Stage	Name	Function
12	Output Metadata Attachment	Attaches timestamps, rule-set version, and identifiers to the output.
13	Evidence-Output Linkage	Creates an inspectable reference from output back to source evidence.
14	Non-Mutation Verification	Confirms the evaluation did not alter underlying evidence.
15	Evaluation Boundary Statement	Declares what the evaluation output does not claim to establish.
16	Cross-Reference Resolution	Resolves references between this and other evaluations.
17	Evaluation Consistency Check	Verifies the output is internally consistent with its declared pathway.
18	Evaluation Record Sealing	Closes the evaluation record against further silent modification.
19	Phase I Closure Statement	Formally closes the evaluation methodology phase for the record.

Phase II — Reflexive Inspection Infrastructure (Stages 20–30)

Stage	Name	Function
20	Inspection Interface Declaration	Declares the interface(s) through which outputs may be examined.
21	Report Mode Definition	Defines the distinct report views available over the same evaluation.
22	Dependency Visibility Layer	Exposes declared dependencies to inspecting parties.
23	Provenance Recording	Records the origin pathway of every declared output.
24	Replay Mechanism Specification	Specifies how identical evidence and rules reproduce identical outputs.
25	Replay Verification	Executes replay and compares results against the sealed output.
26	Inspection Boundary Enforcement	Prevents inspection outputs from exceeding declared scope.
27	Cross-Report Consistency	Verifies that differing report modes do not contradict one another.
28	Public Inspection Access Rules	Declares which outputs are publicly inspectable and which are restricted.
29	Inspection Audit Trail	Logs inspection activity itself as an inspectable record.
30	Phase II Closure Statement	Formally closes the reflexive inspection infrastructure phase.

Phase III — Methodology Governance (Stages 31–40)

Stage	Name	Function
31	Governance Charter Declaration	States the scope and authority boundaries of framework governance.
32	Continuity Record Establishment	Establishes an unbroken record of framework state across versions.
33	Change Register Initialization	Opens a register documenting every methodological change.
34	Version Lineage Documentation	Documents how each version relates to its predecessor.
35	Stewardship Declaration	Declares who preserves the methodology and under what constraints.
36	Preservation Architecture Specification	Specifies how methodological artefacts are kept inspectable over time.
37	Legacy Packaging	Packages methodology documentation independent of any software implementation.

Stage	Name	Function
38	Self-Containment Check	Verifies the specification is understandable without external dependency.
39	Meta-Framework Reflection	Formally describes CREF's own methodological character.
40	Framework Completion Statement	Formally declares the implemented framework complete within its scope.

Appendix B — Framework Relationships

Phase I outputs are the sole inputs to Phase II inspection. Phase II outputs (reports, replay results, provenance records) are what Phase III governance tracks continuity and lineage over — governance does not act on Phase I evidence directly.

Appendix C — Governance Outputs

- Continuity Records, Change Registers, Governance Statements
- Version Lineage, Lifecycle Reviews, Self-Containment Checks
- Stewardship Declarations, Legacy Packages
- Meta-Framework Reflection, Completion Statements

Appendix D — Verification Summary

Verification within CREF refers to replay-based confirmation that a sealed output is reproducible from its declared evidence and rule set (Stage 25). It is not verification of the underlying record's factual accuracy.

Appendix E — Validation Metrics

CREF does not itself define quantitative validation metrics for adopting institutions; any such metrics are an implementation-level or institutional-level concern and fall outside this specification's declared scope.

Framework Completion Statement

The Civic Record Evaluation Framework (CREF) Version 3.1 documents a complete deterministic methodology for civic record evaluation, comprising forty implemented stages organised into three phases. The methodology has been documented, inspected, governed, preserved, reflected upon, and declared complete within its stated scope and limitations. This specification constitutes the canonical description of CREF Version 3.1.

End of Specification.