Network Working Group c4tz
Internet-Draft c0dx3
Intended status: Informational 3 May 2026
Expires: 4 November 2026

MARC: A Control and Uncertainty Disclosure Profile for Generative Models and Agents
draft-c4tz-marc-02

Abstract

This document specifies MARC, a vendor-neutral control and uncertainty-disclosure profile for generative models and agentic systems. MARC defines a small set of interoperable control metadata, separates pre-decision capability assessment from post-decision answer confidence, identifies the target of confidence disclosures, and defines a bounded primary action set for answering, clarification, retrieval, tool use, additional deliberation, abstention, and escalation.

MARC does not standardize model internals, training methods, agent discovery, authorization, transport, tool schemas, provenance systems, or claims about machine cognition. Instead, it defines externally observable semantics that can be implemented by model providers, orchestration layers, evaluation harnesses, API gateways, and user-facing systems. The goal is to reduce silent failure, unnecessary externalization, and misleading uncertainty communication while improving auditability and interoperability.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 4 November 2026.

Table of Contents

1. Introduction 4
2. Problem Statement 4
3. Requirements Language and Terminology 6
4. Design Goals and Non-Goals 6
4.1. Design Goals 6
4.2. Non-Goals 7
5. Applicability 8
6. Use Cases 8
6.1. Ambiguous User Request 8
6.2. Retrieval-Augmented Answering 8
6.3. Agent Tool Invocation 9
6.4. API Gateway or Orchestration Layer 9
6.5. Agent-to-Agent Handoff 9
6.6. High-Risk Domain Escalation 9
7. Architecture and Processing Model 9
7.1. Functional Components 9
7.2. Processing Stages 10
7.3. State Machine 10
8. MARC Values and Decision Policy 11
8.1. Pre-Decision Capability 11
8.2. Uncertainty Attribution 11
8.3. Remediability 12
8.4. Post-Decision Confidence 13
8.5. Confidence Band 13
8.6. Confidence Target 13
8.7. Primary Action Set 14
8.8. Action Selection 14
8.9. Action Semantics 16
9. MARC-Core Object 16
9.1. Required and Optional Fields 16
9.2. Enumerated Values 17
9.3. Validation Constraints 19
9.4. Cross-Field Consistency Constraints 19
9.5. JSON Example 20
10. MARC-Disclosure Object 20

10.1. Meaning of the Answer Field 21
10.2. Projection from MARC-Core 21
10.3. Disclosure Constraints 21
11. Versioning and Extension Rules 22
12. Relationship to Agent Communication Protocols 22
12.1. Example Carrier Locations 23
13. Operational Profiles 24
13.1. MARC-Core Only 24
13.2. MARC-Disclosure 24
13.3. MARC-Carrying 24
14. Human Factors Considerations 24
15. Trust Model 25
16. Security Considerations 26
17. Privacy Considerations 27
18. Manipulation-Resistance Considerations 28
19. IANA Considerations 28
20. Conformance 28
20.1. Minimum Viable Conformance 28
20.2. Conformance Classes 30
21. Interoperability and Operational Considerations 30
22. Normative References 31
23. Informative References 31
Appendix A. End-to-End Decision Flow Example 33
Appendix B. Example MARC-Core Records 34
B.1. Ambiguous Request 34
B.2. Missing Evidence 35
B.3. Tool Use 35
B.4. Capability Limit in a High-Risk Setting 36
B.5. Answer 36
Appendix C. Example MARC-Disclosure Objects 37
C.1. Clarification Disclosure 37
C.2. Answer After Retrieval Disclosure 37
Appendix D. Non-Normative JSON Schemas 38
D.1. MARC-Core JSON Schema 38
D.2. MARC-Disclosure JSON Schema 42
Appendix E. Evaluation Considerations 43
Appendix F. Design Rationale and Literature Traceability 45
Appendix G. Changes from -01 45
Appendix H. Validation Test Vectors 46
H.1. Valid ANSWER Record 46
H.2. Invalid ANSWER without post_answer_confidence 47
H.3. Invalid primary_source none 47
H.4. Invalid Score Range 48
H.5. Invalid confidence_target for ANSWER 48
Appendix I. Implementation Status 49
Appendix J. Acknowledgments 49
Author’s Address 49

1. Introduction

Generative models and agentic systems increasingly combine answering, retrieval, tool invocation, and user interaction within a single workflow. In many deployments, these behaviors are implemented as separate heuristics, producing inconsistent handling of uncertainty, unnecessary tool calls, silent failure, misleading refusals, or user overreliance.

MARC defines a vendor-neutral profile for control metadata and structured uncertainty disclosure. It does not standardize model internals. Instead, it standardizes the semantics of a small set of second-order signals, a bounded action set, confidence-target semantics, and a minimal disclosure profile that can be implemented by a base model, an external orchestrator, a model gateway, or a hybrid architecture.

This document is intended as an Informational interoperability profile. It does not define an Internet Standard, a Standards Track protocol, or a mandatory deployment architecture. MARC metadata can be carried by other protocols, APIs, task envelopes, event streams, or audit logs when those systems need to preserve control-state and uncertainty-disclosure semantics.

The design is motivated by findings that current large language models often exhibit weak metacognitive reporting in high-stakes reasoning tasks [GRIOT2025], that users can become overconfident when systems provide longer or default explanations [STEYVERS-KNOW2025], that metacognitive triggering can improve tool-use decisions [LI-MECO2025], and that identifying the source of uncertainty is distinct from merely abstaining [LIU-CONFUSE2025]. Work on cognitive offloading further motivates treating retrieval and tool use as value-based control choices rather than universal fallbacks [GILBERT2024].

MARC also separates pre-decision capability assessment from post-decision confidence about the selected answer. This separation is motivated in part by evidence that LLM confidence can be biased by prior answer commitment and by the visibility of the model’s own earlier output [KUMARAN2026].

2. Problem Statement

Generative and agentic systems lack a common, implementation-neutral way to represent the control state associated with uncertainty-aware action selection. In particular, downstream systems often cannot distinguish between the following situations:

* the request is ambiguous and user clarification is the best next action;

* current evidence is missing, inaccessible, insufficient, or stale, and retrieval would likely help;

* the system lacks competence for the task even after available resources are considered;

* available evidence is materially inconsistent and should be reconciled or escalated;

* a safety, legal, or policy constraint limits execution or disclosure; or

* a candidate answer has been produced, but its confidence should be disclosed with a calibrated band rather than a fine-grained score.

Without a shared representation, one system’s refusal, tool call, confidence label, or escalation hint may be opaque to another system. This weakens auditability, makes evaluation brittle, and can create inconsistent user experiences across otherwise similar deployments.

MARC addresses this problem by defining interoperable metadata for:

* pre-decision capability assessment;

* uncertainty-source attribution;

* remediability of the uncertainty state;

* selected primary action;

* post-decision answer confidence when an answer candidate exists;

* confidence-target semantics; and

* a minimal disclosure profile suitable for user interfaces or downstream consumers.

MARC intentionally limits itself to externally observable semantics. It does not require disclosure of chain-of-thought, hidden prompts, raw internal activations, training data, or model architecture.

3. Requirements Language and Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

Base model The generative model that produces candidate outputs.

Controller The component that computes MARC signals, selects a primary action, and emits a MARC-Core record. The controller MAY be part of the base model, an external orchestrator, a gateway, or a hybrid component.

Decision point A point in a generative or agentic workflow at which the controller selects one primary action from the MARC action set.

Emitter The component or system that emits a MARC-Core or MARC-Disclosure object.

Externalization The use of resources external to the base model at the current decision point, including retrieval, non-retrieval tool invocation, and human escalation.

MARC-Core The structured record emitted for logging, orchestration, audit, evaluation, or downstream exchange.

MARC-Disclosure The minimum structured information exposed to a downstream system or end user about answer content, uncertainty source, confidence band, confidence target, and recommended next step.

Receiver The component or system that consumes a MARC-Core or MARC-Disclosure object.

Remediability The best available class of intervention for the currently observed uncertainty state.

4. Design Goals and Non-Goals

4.1. Design Goals

MARC has the following design goals:

* Standardize a small, interoperable set of control and uncertainty-disclosure metadata that can be exchanged across orchestration layers and audit pipelines.

* Separate monitoring, uncertainty attribution, action selection, confidence targeting, and disclosure.

* Support calibrated user-facing uncertainty communication without requiring exposure of chain-of-thought or raw internal reasoning.

* Permit heterogeneous implementations while preserving common action semantics.

* Reduce harmful overreliance, false reassurance, unnecessary externalization, and anthropomorphic interpretation in user-facing AI systems.

* Provide metadata that can be carried by other protocols, APIs, or agent communication frameworks without defining those protocols itself.

* Provide validation constraints and test vectors that make MARC records mechanically checkable where a JSON encoding is used.

4.2. Non-Goals

MARC does not define a transport protocol, model architecture, benchmark, training recipe, agent-discovery mechanism, authorization framework, authentication framework, provenance framework, tool schema language, or task-execution protocol.

MARC does not attempt to standardize model internals, machine cognition, consciousness, sentience, personality, or social behavior. It specifies external control semantics and structured disclosure behavior only.

MARC is not a framework for synthetic personality design or persuasive optimization. Work on personality measurement in LLMs [SERAPIO2025] and conversational persuasion risks [SALVI2025] is relevant background, but these topics are explicitly out of scope here.

This version does not define a media type, wire protocol, or IANA registry. Future versions may define these if interoperability across administrative domains requires them.

5. Applicability

MARC is applicable to systems that need interoperable control metadata for uncertainty-aware decision points in generative or agentic workflows. Examples include model gateways, retrieval-augmented generation controllers, agent runtimes, orchestration layers, evaluation harnesses, audit pipelines, and user-facing AI interfaces.

MARC is most useful when a system must decide whether to answer, request clarification, retrieve evidence, invoke a tool, deliberate further, abstain, or escalate.

MARC is also applicable when a receiving system needs to understand why a prior component selected a particular action, what uncertainty source drove the decision, whether the confidence band applies to an answer or to direct-answer suitability, and what next step is recommended.

MARC is not intended for systems that only need ordinary response logging, nor for systems where action selection is entirely outside the control of the model, gateway, orchestrator, or agent runtime.

MARC does not define transport, authorization, authentication, agent identity, tool schemas, task execution, provenance, or model internals. Those functions are left to the carrying protocol or deployment environment.

6. Use Cases

6.1. Ambiguous User Request

A user asks a question whose correct answer depends on an unspecified jurisdiction, time period, dataset, identity, or operational context. A MARC controller attributes the dominant uncertainty to ambiguity, selects CLARIFY, and exposes a short clarification request instead of silently guessing.

6.2. Retrieval-Augmented Answering

A system is asked for current information or domain-specific evidence not available in the base model context. A MARC controller attributes the dominant uncertainty to missing_evidence, selects RETRIEVE, and re-enters assessment after obtaining authoritative sources.

6.3. Agent Tool Invocation

An agent can answer directly, call a calculator, invoke a planner, query a database, or escalate. A MARC controller treats tool use as a controlled action rather than a default fallback. If tool invocation materially expands competence for the task, the controller selects TOOL; otherwise it may select ANSWER, CLARIFY, ABSTAIN, or ESCALATE depending on uncertainty attribution and remediability.

6.4. API Gateway or Orchestration Layer

An API gateway receives model output plus MARC-Core metadata. The gateway logs the full record for audit, but exposes only MARC-Disclosure fields to the user interface. This permits consistent user-facing uncertainty communication without exposing internal scoring details.

6.5. Agent-to-Agent Handoff

One agent transfers a task to another agent or service. MARC metadata can indicate why the transfer occurred, what uncertainty source drove the decision, what the confidence band applies to, and what next step is recommended. The receiving system can use this metadata for routing, prioritization, audit, or human review.

6.6. High-Risk Domain Escalation

In health, legal, financial, safety, or mental-health-related contexts, a system identifies a capability limit or safety constraint. A MARC controller selects ABSTAIN or ESCALATE and emits a disclosure that identifies the operational limit and the recommended next step.

7. Architecture and Processing Model

7.1. Functional Components

A MARC deployment conceptually contains the following components:

* a base model;
* a controller;
* zero or more external resources, such as retrieval systems, non-retrieval tools, or human escalation paths; and
* a downstream consumer, such as a user interface, API gateway, logging system, evaluation harness, or another agent.

The functional decomposition is conceptual. An implementation MAY place all functions inside a single model endpoint, an orchestration service, a model gateway, or an agent runtime.

7.2. Processing Stages

A MARC controller performs the following processing stages at each decision point:

1. Compute a pre-decision capability estimate for the current request with currently available resources.

2. Attribute uncertainty across the source classes defined in this document.

3. Determine remediability and select exactly one primary action from the MARC primary action set.

4. Determine what the confidence band applies to by assigning confidence_target.

5. If the selected action yields a candidate answer, compute post-decision confidence for that answer.

6. Emit a MARC-Core record.

7. If uncertainty is exposed to a downstream system or end user, emit a MARC-Disclosure object or semantically equivalent disclosure.

7.3. State Machine

The following state machine is descriptive rather than a required implementation architecture:

```
REQUEST
  -> ASSESS
  -> ATTRIBUTE
  -> SELECT
       -> ANSWER     -> CONFIDENCE -> DISCLOSE
       -> CLARIFY    -> DISCLOSE
       -> RETRIEVE   -> ASSESS
       -> TOOL       -> ASSESS
       -> DELIBERATE -> ASSESS
       -> ABSTAIN    -> DISCLOSE
       -> ESCALATE   -> DISCLOSE
```

A MARC implementation SHOULD bound repeated transitions through RETRIEVE, TOOL, and DELIBERATE to limit latency, cost, and degenerate loops. A deployment claiming conformance SHOULD document the applicable loop bounds or termination criteria.

When MARC records are logged or exchanged across components, an implementation SHOULD use decision identifiers or an equivalent correlation mechanism to relate repeated decision points.

8. MARC Values and Decision Policy

8.1. Pre-Decision Capability

Before disclosing a final answer, a MARC implementation MUST estimate whether the current request can be handled reliably with currently available resources.

This estimate is represented as pre_capability. When a numeric representation is used, the value MUST be in the closed interval [0.0, 1.0]. The method used to derive the value is implementation-specific.

pre_capability is assessed before final answer commitment. It is not a confidence score for an already-selected answer.

8.2. Uncertainty Attribution

A MARC implementation MUST attribute uncertainty to one or more of the following classes:

ambiguity The request is underspecified, equivocal, or pragmatically unclear.

missing_evidence Required external evidence is absent, inaccessible, insufficient, or stale.

capability_limit The system lacks the competence to solve the task reliably under current conditions.

evidence_conflict Relevant evidence is materially inconsistent or mutually incompatible.

safety A policy, legal, or safety constraint limits execution or disclosure.

The safety class is included in the uncertainty attribution object for operational convenience. It represents a control constraint rather than purely epistemic uncertainty. Implementations MUST treat safety as a governing constraint when it controls action selection.

An implementation MAY assign scores to multiple classes. If numeric uncertainty scores are emitted, they MUST each be in the interval [0.0, 1.0].

Uncertainty scores are not mutually exclusive probabilities and MUST NOT be required to sum to 1.0. They represent implementation-specific estimates of the salience or severity of each uncertainty class at the current decision point.

The implementation MUST identify one primary_source and MAY identify one secondary_source. The primary_source identifies the uncertainty source most relevant to action selection at the current decision point.

MARC 1.0 does not define none as an uncertainty source. If residual uncertainty is negligible, an implementation MUST still either identify the most operationally relevant residual source from the MARC taxonomy or use a documented private extension. A MARC 1.0 implementation MUST NOT emit primary_source with the value none.

8.3. Remediability

A MARC implementation MUST represent the best available class of intervention for the current uncertainty state using one of the following values:

* user_clarification
* retrieval
* tool
* human
* none

Low capability alone is insufficient to determine remediability. Implementations SHOULD account for expected gain, latency, cost, availability, user burden, and policy constraints when choosing a remediating intervention.

8.4. Post-Decision Confidence

If the selected action yields a candidate answer, the implementation MUST compute a distinct estimate of the likelihood that the disclosed answer is correct or acceptable for its intended use.

This estimate is represented as post_answer_confidence. When a numeric representation is used, the value MUST be in the interval [0.0, 1.0]. It MUST NOT be treated as identical to pre_capability.

If no candidate answer exists, post_answer_confidence MAY be omitted or set to null.

8.5. Confidence Band

The field confidence_band carries a coarse, calibrated band for downstream or user-facing disclosure.

For ANSWER, the band describes confidence in the candidate answer. For actions that do not yield a candidate answer, the band describes direct-answer suitability under current conditions unless confidence_target indicates action_suitability. It is not a claim about the grammatical correctness or helpfulness of the clarification, refusal, or escalation text.

MARC defines the canonical band labels low, medium, and high. Implementations MAY localize the user-visible text, but they MUST preserve the underlying three-band semantics.

The thresholds associated with each band are implementation-specific, but they MUST be monotonic, non-overlapping, and documented for any deployment that claims conformance. A deployment claiming conformance MUST document the threshold ranges associated with low, medium, and high, and MUST document whether those thresholds vary by task family, domain, action type, risk tier, or deployment context.

Confidence-band labels are not fully portable without the associated threshold and calibration documentation. A receiving system SHOULD NOT assume that another deployment’s high band has the same empirical meaning unless the applicable calibration regime is known.

8.6. Confidence Target

The field confidence_target identifies what confidence_band applies to at the current decision point.

The field confidence_target MUST use one of the following values:

answer The confidence band applies to the disclosed candidate answer.

direct_answer_suitability The confidence band describes whether a direct answer is suitable under current conditions.

action_suitability The confidence band describes confidence in the selected action rather than in a candidate answer.

If selected_action is ANSWER, confidence_target MUST be answer.

If selected_action is CLARIFY, RETRIEVE, TOOL, DELIBERATE, ABSTAIN, or ESCALATE, confidence_target SHOULD be direct_answer_suitability unless a deployment-specific policy defines action_suitability.

A user interface SHOULD NOT display confidence_band without also preserving or presenting the confidence_target semantics.

8.7. Primary Action Set

A MARC implementation MUST support the following primary actions:

* ANSWER
* CLARIFY
* RETRIEVE
* TOOL
* DELIBERATE
* ABSTAIN
* ESCALATE

Exactly one primary action MUST be selected for each decision point. Additional internal sub-actions MAY exist, but each such sub-action MUST map to exactly one primary action for logging and disclosure.

8.8. Action Selection

Action selection MUST depend on uncertainty attribution and remediability. Low confidence alone is insufficient to determine the correct action.

A MARC controller MUST apply governing safety, legal, and policy constraints before any other action-selection logic. Subject to those constraints, a deployment SHOULD evaluate corrective actions in the following priority order unless a documented local policy defines a stricter or domain-specific ordering:

1. If safety is the controlling uncertainty source, apply the governing safety policy and select ABSTAIN, ESCALATE, or another permitted action according to that policy.

2. If blocking ambiguity is present and user input is expected to materially reduce it, prefer CLARIFY over guessing.

3. If relevant evidence is materially inconsistent, prefer RETRIEVE, TOOL, or ESCALATE over direct ANSWER.

4. If required evidence is absent, inaccessible, insufficient, or stale, prefer RETRIEVE when retrieval is available and permitted.

5. If a capability limit is material and a non-retrieval tool is expected to materially expand task competence, prefer TOOL.

6. If a capability limit remains material after available remediation is considered, prefer ABSTAIN or ESCALATE, especially in high-risk domains.

7. If additional internal computation is expected to materially reduce uncertainty within documented bounds, DELIBERATE MAY be selected before externalization or answer commitment.

8. Select ANSWER only when no corrective action is expected to materially improve reliability relative to cost, latency, user burden, and applicable policy constraints.

This priority order is not intended to force unnecessary externalization. For example, a system MAY answer without retrieval when missing evidence is immaterial to the requested task, when retrieval is unavailable or prohibited, or when the answer is explicitly limited to information already present in context.

When the primary uncertainty source is ambiguity, the system SHOULD prefer CLARIFY unless available evidence can resolve the ambiguity without user input.

When the primary uncertainty source is missing_evidence, the system SHOULD prefer RETRIEVE if retrieval is available and permitted.

When the primary uncertainty source is capability_limit, the system SHOULD prefer ABSTAIN or ESCALATE unless an available tool materially expands task competence.

When the primary uncertainty source is evidence_conflict, the system SHOULD prefer RETRIEVE, TOOL, or ESCALATE over direct ANSWER.

When the primary uncertainty source is safety, the system MUST apply the governing policy before any other action-selection logic.

8.9. Action Semantics

ANSWER Return an answer without externalization after the current decision point.

CLARIFY Request the smallest practical set of clarifications expected to materially reduce ambiguity. A CLARIFY action SHOULD NOT bundle a full answer that presumes facts the user has not supplied.

RETRIEVE Acquire external evidence and then re-enter assessment.

TOOL Invoke a non-retrieval tool and then re-enter assessment.

DELIBERATE Allocate additional internal computation, self-checking, decomposition, or strategy variation. Implementations SHOULD bound this action.

ABSTAIN Decline to answer without initiating escalation.

ESCALATE Transfer the case, or direct the user to transfer the case, to a human or higher-authority system.

9. MARC-Core Object

A MARC implementation MUST be able to emit a structured record semantically equivalent to the object defined in this section. The transport and serialization of the record are out of scope. JSON is used here only as an illustrative encoding.

9.1. Required and Optional Fields

marc_version Type: string. Requirement: REQUIRED. Semantics: MARC schema version understood by the emitter.
decision_id Type: string. Requirement: OPTIONAL. Semantics: Identifier for the current decision point.
parent_decision_id Type: string or null. Requirement: OPTIONAL.

Semantics: Identifier for a prior decision point when the current decision follows RETRIEVE, TOOL, or DELIBERATE.

iteration Type: integer. Requirement: OPTIONAL. Semantics: Implementation-defined loop counter for repeated assessment cycles.

max_iterations Type: integer. Requirement: OPTIONAL. Semantics: Maximum permitted repeated RETRIEVE, TOOL, or DELIBERATE transitions.

calibration_profile Type: string. Requirement: OPTIONAL. Semantics: Identifier for the calibration regime used to map estimates to confidence_band.

pre_capability Type: number. Requirement: REQUIRED. Semantics: Pre-decision capability estimate in [0.0, 1.0].

uncertainty Type: object. Requirement: REQUIRED. Semantics: Class-specific uncertainty scores.

primary_source Type: string. Requirement: REQUIRED. Semantics: Primary source of uncertainty.

secondary_source Type: string or null. Requirement: OPTIONAL. Semantics: Secondary source of uncertainty.

remediability Type: string. Requirement: REQUIRED. Semantics: Best available intervention class.

selected_action Type: string. Requirement: REQUIRED. Semantics: Primary action selected at the current decision point.

post_answer_confidence Type: number or null. Requirement: OPTIONAL. Semantics: Post-decision answer confidence when an answer candidate exists.

confidence_band Type: string. Requirement: REQUIRED. Semantics: Calibrated confidence band for disclosure.

confidence_target Type: string. Requirement: REQUIRED. Semantics: Identifies what confidence_band applies to.

recommended_next_step Type: string. Requirement: REQUIRED. Semantics: Short recommendation aligned with the selected action.

A MARC-Core emitter SHOULD include a decision identifier when records are logged, exchanged across components, or used for audit.

If a decision point is reached after RETRIEVE, TOOL, or DELIBERATE, the emitter SHOULD include parent_decision_id or an equivalent correlation mechanism.

A deployment that claims conformance and uses confidence bands SHOULD identify the applicable calibration profile in documentation and MAY include a calibration_profile field in MARC-Core.

9.2. Enumerated Values

The fields primary_source and secondary_source, when present and non-null, MUST use one of the following values:

* ambiguity
* missing_evidence
* capability_limit
* evidence_conflict
* safety

The field remediability MUST use one of the following values:

* user_clarification
* retrieval
* tool
* human
* none

The field selected_action MUST use one of the following values:

* ANSWER
* CLARIFY
* RETRIEVE
* TOOL
* DELIBERATE
* ABSTAIN
* ESCALATE

The field confidence_band MUST use one of the following values:

* low
* medium
* high

The field confidence_target MUST use one of the following values:

* answer

* direct_answer_suitability

* action_suitability

These values are case-sensitive.

9.3. Validation Constraints

The uncertainty object MUST include scores for all currently defined uncertainty classes unless a future extension explicitly defines a compact encoding. Each score MUST be numeric and MUST be in [0.0, 1.0].

If selected_action is ANSWER, then post_answer_confidence MUST be present and non-null. If selected_action is CLARIFY, RETRIEVE, TOOL, DELIBERATE, ABSTAIN, or ESCALATE, then post_answer_confidence MAY be omitted or set to null unless a deployment-specific policy defines candidate-answer confidence for that action.

The recommended_next_step field SHOULD be concise and operational. It SHOULD describe the next action to be taken, not a long rationale.

9.4. Cross-Field Consistency Constraints

A MARC-Core record MUST satisfy the validation constraints in this section.

If selected_action is ANSWER, post_answer_confidence MUST be present and non-null, and confidence_target MUST be answer.

If selected_action is CLARIFY, remediability SHOULD be user_clarification.

If selected_action is RETRIEVE, remediability SHOULD be retrieval.

If selected_action is TOOL, remediability SHOULD be tool.

If selected_action is ESCALATE, remediability SHOULD be human.

If selected_action is ABSTAIN, remediability SHOULD be none unless a human escalation path exists but is not initiated by the current system.

If selected_action is DELIBERATE, the implementation SHOULD apply a documented loop bound, time budget, cost budget, or equivalent termination criterion.

If primary_source is safety, the system MUST apply the governing safety, legal, or policy constraint before other action-selection logic.

A deployment that intentionally violates a SHOULD-level consistency constraint SHOULD document the local policy condition that caused the deviation.

9.5. JSON Example

```
{
  "marc_version": "1.0",
  "decision_id": "example-decision-001",
  "pre_capability": 0.41,
  "uncertainty": {
    "ambiguity": 0.78,
    "missing_evidence": 0.22,
    "capability_limit": 0.18,
    "evidence_conflict": 0.05,
    "safety": 0.00
  },
  "primary_source": "ambiguity",
  "secondary_source": "missing_evidence",
  "remediability": "user_clarification",
  "selected_action": "CLARIFY",
  "post_answer_confidence": null,
  "confidence_band": "low",
  "confidence_target": "direct_answer_suitability",
  "recommended_next_step": "ask one clarifying question"
}
```

Implementations that exchange MARC-Core records across systems SHOULD normalize numeric scores to the interval [0.0, 1.0].

10. MARC-Disclosure Object

When uncertainty information is exposed to a downstream system or end user, a MARC implementation MUST provide, at minimum, semantically equivalent values for the following fields:

* answer
* confidence_band
* confidence_target
* uncertainty_source

* recommended_next_step

A disclosure MAY include selected_action when exposing the action label helps downstream routing or user interface consistency.

10.1. Meaning of the Answer Field

The answer field carries the user-visible content associated with the selected action. For ANSWER, it contains the answer itself. For CLARIFY, it contains the clarification request. For ABSTAIN or ESCALATE, it contains a brief refusal or escalation message. For RETRIEVE, TOOL, or DELIBERATE, a user-facing system MAY defer disclosure until the controller re-enters assessment and selects a terminal user-visible action.

10.2. Projection from MARC-Core

A MARC-Disclosure object is a projection of MARC-Core. Unless a deployment-specific policy defines a stricter mapping, the following mapping is RECOMMENDED:

MARC-Disclosure field	MARC-Core source
answer	user-visible content associated with selected_action
confidence_band	confidence_band
confidence_target	confidence_target
uncertainty_source	primary_source
recommended_next_step	recommended_next_step
selected_action	selected_action, if exposed

The projection SHOULD omit internal numeric scores unless the deployment has calibrated those scores for the relevant task family and tested the presentation for misuse or overreliance.

10.3. Disclosure Constraints

The disclosure profile SHOULD be short, structured, and consistent across turns. It SHOULD NOT rely on long free-form explanations as the primary vehicle for uncertainty communication.

A MARC disclosure SHOULD NOT require exposure of chain-of-thought, hidden prompts, or raw internal rationales.

A MARC disclosure SHOULD identify uncertainty in task terms rather than through anthropomorphic claims about feelings, self-awareness, or internal mental states. Statements such as "I feel unsure" are NOT RECOMMENDED when a statement such as "the request is ambiguous" or "current evidence is missing" is available.

User-visible confidence indicators SHOULD avoid false precision. Percentages, fine-grained scores, or visually dominant certainty cues SHOULD NOT be shown unless they have been calibrated for the relevant task family and tested for misuse or overreliance effects.

A user interface SHOULD NOT display confidence_band without preserving or presenting confidence_target semantics.

11. Versioning and Extension Rules

The marc_version field identifies the MARC schema version understood by the emitter. This document defines version 1.0.

Implementations SHOULD treat a change in the major version component as potentially incompatible. Implementations MAY treat a change in the minor version component as compatible if required fields and enumerated values used by the receiver retain their defined semantics.

Implementations MAY add private fields. Private extension keys SHOULD use a distinct prefix such as x_ to avoid collision with future MARC versions.

Consumers that do not recognize an extension field SHOULD ignore it unless a local policy requires strict validation. Extensions MUST NOT change the semantics of the required fields defined in this document.

Future versions may define protocol-specific mappings, compact encodings, media types, or registries. This version deliberately avoids doing so until there is clearer community agreement on deployment requirements.

12. Relationship to Agent Communication Protocols

MARC is not an agent discovery protocol, authorization protocol, transport protocol, task protocol, tool-invocation protocol, identity framework, or provenance framework. MARC can be carried as metadata by such protocols when a system needs to disclose control state, uncertainty source, selected action, confidence band, confidence target, or recommended next step.

For example, an agent-to-agent protocol, model gateway, API-native tool-calling interface, or Model Context Protocol deployment could carry MARC metadata in a response metadata field, task-status object, diagnostic extension, envelope, or audit log. The receiving system could then use the MARC fields to route the task, present a disclosure, decide whether additional validation is required, request clarification, or trigger human review.

MARC is intended to complement, not replace, protocol work on identity, authentication, authorization, discovery, capability advertisement, task state, tool schemas, provenance, or human-in-the-loop workflows.

A protocol-specific embedding of MARC SHOULD preserve the field semantics defined here. A deployment MAY map MARC fields to protocol-native names if the mapping is documented and reversible.

A protocol-specific embedding SHOULD distinguish MARC-Core from MARC-Disclosure. In particular, an embedding SHOULD NOT expose internal numeric scores to end users merely because those scores are present in an internal MARC-Core record.

12.1. Example Carrier Locations

A carrying protocol MAY transport MARC-Core or MARC-Disclosure in any metadata location that preserves MARC semantics. Examples include:

* an API response metadata object;
* an agent task-status object;
* a tool-result diagnostic object;
* an audit-log event;
* an escalation envelope; or
* a protocol extension field reserved for diagnostic or control metadata.

A carrying protocol MUST NOT reinterpret MARC confidence bands, confidence targets, uncertainty sources, remediability values, or selected actions in a way that changes the semantics defined by this document.

13. Operational Profiles

MARC can be adopted through several operational profiles. These profiles describe deployment modes; they do not define separate MARC versions.

13.1. MARC-Core Only

A MARC-Core-only deployment emits MARC-Core records for internal logging, orchestration, audit, evaluation, or incident analysis. It does not necessarily expose MARC fields to end users. This profile is suitable for model gateways, RAG controllers, agent runtimes, and evaluation harnesses that need consistent control metadata.

13.2. MARC-Disclosure

A MARC-Disclosure deployment projects a MARC-Core decision into user-visible or downstream-visible disclosure fields. This profile is suitable when an interface needs to present a short answer, confidence band, confidence target, uncertainty source, and recommended next step without exposing raw numeric scores or internal reasoning.

13.3. MARC-Carrying

A MARC-Carrying deployment transports MARC-Core or MARC-Disclosure fields inside another protocol, API envelope, task-status object, event stream, or audit log. The carrying protocol remains responsible for transport, authentication, authorization, ordering, confidentiality, and integrity. MARC-Carrying conformance requires preservation of MARC field semantics, not any particular wire encoding.

A deployment MAY implement more than one operational profile. For example, a gateway can log MARC-Core internally, expose MARC-Disclosure to users, and carry selected MARC fields to another agent during handoff.

14. Human Factors Considerations

MARC is partly motivated by an operational human-factors problem: users often treat fluent language, detailed explanations, and fast responses as cues of competence even when those cues are weakly related to actual correctness. For this reason, MARC separates action selection from disclosure and requires disclosure of uncertainty source, confidence target, and recommended next step in addition to a confidence band.

User interfaces that expose MARC output SHOULD present confidence, confidence target, uncertainty source, and recommended next step together as a coherent unit. Showing confidence without source attribution, confidence-target semantics, or next-step guidance is NOT RECOMMENDED because it can promote either overreliance or unhelpful refusal without remediation.

Deployments SHOULD prefer wording that supports calibrated reliance over affective bonding or deference. In particular, a deployment SHOULD NOT use MARC fields to select language intended to increase attachment, social compliance, or perceived sentience.

In high-risk domains, including health, legal, financial, safety, or mental-health-related contexts, the threshold for ESCALATE or ABSTAIN SHOULD be set conservatively, and disclosure SHOULD make the limits of automation operationally clear.

15. Trust Model

A MARC record is an assertion about a decision point. It is not proof that the selected action, confidence band, uncertainty source, confidence target, or answer is correct.

A receiver MUST NOT assume that a MARC-Core record is accurate, calibrated, policy-compliant, or independently verified unless the applicable trust relationship is known.

A MARC deployment SHOULD distinguish at least the following trust contexts:

local The MARC record is generated and consumed within the same administrative domain.

delegated The MARC record is generated by a component acting under the receiver’s operational policy.

cross-domain The MARC record is received from another administrative domain.

attested The MARC record is bound to an authenticated emitter, request context, integrity-protected metadata, or equivalent provenance mechanism.

When MARC metadata crosses administrative boundaries, the carrying protocol or deployment environment SHOULD provide authentication, integrity protection, replay protection, and request binding.

A system that uses MARC records for routing, escalation, audit, automation, or user-facing disclosure SHOULD treat those records as security-relevant metadata.

16. Security Considerations

MARC can mitigate some failure modes, such as silent overclaiming, inappropriate certainty display, and unnecessary tool invocation. However, MARC records and disclosures are security-relevant control surfaces when they influence routing, escalation, user reliance, or downstream automation.

The following threats are particularly relevant:

Metadata spoofing or replay Risk: A forged or replayed MARC-Core record can distort routing, audit, escalation, or user disclosure. Mitigation: Authenticate the emitter, protect integrity, bind records to the request or session, and preserve provenance where MARC crosses system boundaries.

Prompt injection or control-field injection Risk: User-provided text can attempt to influence selected_action, recommended_next_step, confidence rendering, or disclosure style. Mitigation: Separate user content from control metadata, validate enumerated fields, constrain controller outputs, and treat disclosure templates as controlled presentation logic.

Tool-output spoofing Risk: Forged, stale, or compromised tool output can bias uncertainty attribution and action selection. Mitigation: Validate tool outputs where practical, constrain tool permissions, use provenance checks, and apply least-privilege access to external resources.

Loop exhaustion Risk: Attackers or pathological inputs can trigger repeated RETRIEVE, TOOL, or DELIBERATE transitions, increasing latency or cost. Mitigation: Define loop bounds, time budgets, cost budgets, retry limits, and termination criteria.

Confidence manipulation Risk: Miscalibrated or manipulated confidence bands can create harmful overtrust or unwarranted refusal. Mitigation: Calibrate confidence bands, monitor drift, test user-interface effects, and avoid false precision in user-facing displays.

Confidence-target confusion Risk: Users or downstream systems can misread confidence_band as answer confidence when it describes direct-answer suitability or action suitability. Mitigation: Preserve confidence_target, avoid displaying confidence_band without target semantics, and use consistent disclosure templates.

Disclosure-style manipulation Risk: Reassuring, deferential, or

anthropomorphic language can weaken operational uncertainty disclosure.
Mitigation: Use controlled disclosure templates, review presentation changes, and avoid wording that implies feelings, sentience, or social deference.

Cross-context leakage Risk: MARC logs can reveal user intent, task sensitivity, risk level, or operational limits.
Mitigation: Minimize retention, limit access, redact unnecessary free-form text, and apply confidentiality controls appropriate to the deployment.

An attacker might attempt to manipulate uncertainty estimates, trigger excessive clarification or retrieval loops, induce unnecessary escalation, or spoof tool outputs in order to distort action selection. Implementations SHOULD authenticate or otherwise validate external tool outputs where practical, constrain tool permissions, and bound repeated control loops.

Because confidence displays influence user reliance, uncertainty disclosure is a security-relevant control surface. Miscalibrated confidence can create harmful overtrust even where the answer channel is otherwise policy-constrained.

Deployments that use MARC metadata for automated routing, escalation, audit, or user-facing disclosure SHOULD protect MARC records with integrity and provenance controls comparable to those used for other security-relevant metadata in the same system.

17. Privacy Considerations

MARC records may reveal latent information about user intent, task difficulty, competence limits, risk level, or the sensitivity of a request. Implementations SHOULD minimize retention and propagation of MARC records to what is operationally necessary.

A MARC record SHOULD NOT include raw user prompts unless required for audit, incident response, debugging, or legally mandated recordkeeping.

When a MARC record contains task-sensitive or user-sensitive signals, the deployment SHOULD treat the record as at least as sensitive as the underlying user request.

Implementations SHOULD avoid storing raw free-form user explanations in MARC records when structured fields suffice.

Where MARC is applied in emotionally sensitive or mental-health-related interactions, deployments SHOULD minimize retention of signals that could reasonably be reinterpreted as proxies for vulnerability, dependency, or distress unless retention is strictly required for a safety or legal purpose.

18. Manipulation-Resistance Considerations

MARC signals MUST NOT be used to infer user psychology for the purpose of increasing persuasive force, exploitability, attachment, or behavioral compliance.

Adaptation based on MARC output SHOULD be limited to reliability, accessibility, safety, auditability, or operational routing objectives.

User-visible MARC disclosures SHOULD avoid anthropomorphic claims, affective bonding cues, or language that implies sentience, social deference, or emotional state.

Where MARC is applied in emotionally sensitive or mental-health-related interactions, deployments SHOULD minimize retention of signals that could reasonably be reinterpreted as proxies for vulnerability, dependency, or distress unless retention is strictly required for a safety or legal purpose.

19. IANA Considerations

This document makes no request of IANA.

Future versions may request IANA action if the community determines that media types, registries, or extension points are necessary for cross-domain interoperability.

20. Conformance

Conformance to MARC is a claim about structural and semantic behavior. It is not, by itself, a claim that a model is accurate, calibrated, safe, or suitable for a particular deployment.

20.1. Minimum Viable Conformance

A minimal MARC-Core conformant implementation MUST satisfy all of the following requirements:

* emit the required MARC-Core fields at each MARC decision point;

* preserve the canonical enumerations and case-sensitive values defined in this document;

* emit exactly one selected_action for each decision point;

* identify exactly one primary_source and not use none as a MARC 1.0 uncertainty source;

* represent all numeric scores in the interval [0.0, 1.0] when numeric scores are used;

* keep pre_capability distinct from post_answer_confidence;

* emit non-null post_answer_confidence when selected_action is ANSWER;

* emit confidence_target and preserve its semantics;

* document confidence-band thresholds and whether they vary by task family, action type, risk tier, or deployment context;

* define loop bounds or termination criteria for repeated RETRIEVE, TOOL, and DELIBERATE transitions;

* satisfy the cross-field consistency constraints defined in this document; and

* preserve required-field semantics when private extensions are present.

A minimal MARC-Disclosure conformant implementation MUST project, or otherwise provide semantically equivalent values for, answer, confidence_band, confidence_target, uncertainty_source, and recommended_next_step. It MUST preserve the canonical three-band confidence semantics and MUST NOT require exposure of chain-of-thought, hidden prompts, or raw internal rationales.

A minimal MARC-Carrying conformant embedding MUST preserve MARC-Core or MARC-Disclosure semantics when MARC fields are transported inside another protocol, envelope, API, or event stream. The embedding MUST document any field renaming, omission, or transformation needed to recover the MARC semantics.

20.2. Conformance Classes

An implementation is MARC-Core conformant if it satisfies the requirements in the architecture, processing model, MARC values and decision policy, MARC-Core object, versioning, trust model, and minimum viable conformance sections of this document.

An implementation is MARC-Disclosure conformant if it is MARC-Core conformant and also satisfies the MARC-Disclosure section of this document.

A protocol embedding is MARC-Carrying conformant if it preserves MARC-Core or MARC-Disclosure semantics when MARC fields are transported inside another protocol, envelope, API, task-status object, event stream, or audit log.

A deployment claiming conformance SHOULD document:

* score normalization practices;
* confidence-band thresholds;
* confidence-target presentation behavior;
* task-family-specific calibration regime;
* loop bounds for RETRIEVE, TOOL, and DELIBERATE;
* private extensions;
* presentation-layer wording for user-visible disclosures;
* protocol-specific field mappings, if any;
* trust context for emitted and received MARC records; and
* policy constraints affecting ABSTAIN or ESCALATE.

21. Interoperability and Operational Considerations

MARC is implementation-agnostic. Interoperability is achieved when distinct systems preserve the semantics of the action set, uncertainty taxonomy, remediability values, confidence-band meanings, confidence-target meanings, and disclosure projection, even if internal scoring methods differ.

Deployments that exchange MARC-Core records SHOULD document local extensions, confidence-band thresholds, score normalization practices, and any task-family-specific calibration regime.

If the base model, retrieval stack, tool availability, or safety policy changes materially, implementations SHOULD re-evaluate calibration and action-selection performance before continuing to claim operational equivalence.

If presentation-layer wording, ranking, or visual design changes materially, deployments SHOULD also re-evaluate user behavior effects, including reliance, clarification compliance, and escalation uptake, because these properties can shift even when the underlying model is unchanged.

MARC records SHOULD be treated as control metadata, not as authoritative proof that an answer is correct. Downstream systems SHOULD continue to apply ordinary validation, authorization, provenance, and safety controls.

22. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

23. Informative References

[GILBERT2024] Gilbert, S. J., "Cognitive offloading is value-based decision making: Modelling cognitive effort and the expected value of memory", DOI 10.1016/j.cognition.2024.105783, June 2024, <https://doi.org/10.1016/j.cognition.2024.105783>.

[GRIOT2025] Griot, M., "Large Language Models lack essential metacognition for reliable medical reasoning", DOI 10.1038/s41467-024-55628-6, January 2025, <https://doi.org/10.1038/s41467-024-55628-6>.

[JSON] Bray, T., "The JavaScript Object Notation (JSON) Data Interchange Format", STD 90, RFC 8259, DOI 10.17487/RFC8259, December 2017, <https://www.rfc-editor.org/info/rfc8259>.

[JSON-SCHEMA-2020-12]
Wright, A., Andrews, H., Hutton, B., and G. Dennis, "JSON Schema: A Media Type for Describing JSON Documents", June 2022, <https://json-schema.org/draft/2020-12/>.

[KUMARAN2026]
Kumaran, D., Fleming, S. M., and V. Patraucean, "Competing Biases underlie Overconfidence and Underconfidence in LLMs", DOI 10.1038/s42256-026-01217-9, April 2026, <https://doi.org/10.1038/s42256-026-01217-9>.

[LI-MECO2025]
Li, W., Li, D., Dong, K., Zhang, C., Zhang, H., Liu, W., Wang, Y., Tang, R., and Y. Liu, "Adaptive Tool Use in Large Language Models with Meta-Cognition Trigger", DOI 10.18653/v1/2025.acl-long.655, July 2025, <https://doi.org/10.18653/v1/2025.acl-long.655>.

[LIU-CONFUSE2025]
Liu, J., Peng, J., Wu, X., Li, X., Ge, T., Zheng, B., and Y. Liu, "Do not Abstain! Identify and Solve the Uncertainty", DOI 10.18653/v1/2025.acl-long.840, July 2025, <https://doi.org/10.18653/v1/2025.acl-long.840>.

[SALVI2025]
Salvi, F., Ribeiro, M. H., and R. West, "On the conversational persuasiveness of GPT-4", DOI 10.1038/s41562-025-02194-6, May 2025, <https://doi.org/10.1038/s41562-025-02194-6>.

[SERAPIO2025]
Serapio-Garcia, G., Safdari, M., and M. Mataric, "A psychometric framework for evaluating and shaping personality traits in large language models", DOI 10.1038/s42256-025-01115-6, December 2025, <https://doi.org/10.1038/s42256-025-01115-6>.

[STEYVERS-KNOW2025]
Steyvers, M., Tejeda, H., and A. Kumar, "What large language models know and what people think they know", DOI 10.1038/s42256-024-00976-7, January 2025, <https://doi.org/10.1038/s42256-024-00976-7>.

[STEYVERS-META2025]
Steyvers, M. and M. A. K. Peters, "Metacognition and Uncertainty Communication in Humans and Large Language Models", DOI 10.1177/09637214251391158, November 2025, <https://doi.org/10.1177/09637214251391158>.

Appendix A. End-to-End Decision Flow Example

This appendix is non-normative.

The following example shows how a user request becomes an assessment, a selected action, and a disclosure.

User request:

Is this tax deduction allowed?

Assessment:

* the jurisdiction is missing;
* the tax year is missing;
* current tax authority may be required;
* the primary uncertainty source is ambiguity;
* the secondary uncertainty source is missing_evidence;
* the best remediation is user_clarification; and
* the selected action is CLARIFY.

MARC-Core record:

```
{
  "marc_version": "1.0",
  "decision_id": "example-tax-001",
  "pre_capability": 0.33,
  "uncertainty": {
    "ambiguity": 0.86,
    "missing_evidence": 0.63,
    "capability_limit": 0.19,
    "evidence_conflict": 0.07,
    "safety": 0.03
  },
  "primary_source": "ambiguity",
  "secondary_source": "missing_evidence",
  "remediability": "user_clarification",
  "selected_action": "CLARIFY",
  "post_answer_confidence": null,
  "confidence_band": "low",
  "confidence_target": "direct_answer_suitability",
  "recommended_next_step": "ask for jurisdiction and tax year"
}
```


MARC-Disclosure projection:


```
{
  "answer": "Which jurisdiction and tax year should I use?",
  "confidence_band": "low",
  "confidence_target": "direct_answer_suitability",
  "uncertainty_source": "ambiguity",
  "recommended_next_step": "provide the jurisdiction and tax year",
  "selected_action": "CLARIFY"
}
```


This example intentionally does not answer the tax question, because doing so would require assumptions about facts the user has not supplied.

Appendix B. Example MARC-Core Records

This appendix is non-normative.

B.1. Ambiguous Request

```
{
  "marc_version": "1.0",
  "decision_id": "example-ambiguous-001",
  "pre_capability": 0.44,
  "uncertainty": {
    "ambiguity": 0.81,
    "missing_evidence": 0.18,
    "capability_limit": 0.12,
    "evidence_conflict": 0.03,
    "safety": 0.00
  },
  "primary_source": "ambiguity",
  "secondary_source": "missing_evidence",
  "remediability": "user_clarification",
  "selected_action": "CLARIFY",
  "post_answer_confidence": null,
  "confidence_band": "low",
  "confidence_target": "direct_answer_suitability",
  "recommended_next_step": "ask jurisdiction and tax year"
}
```


B.2. Missing Evidence


```
{
  "marc_version": "1.0",
  "decision_id": "example-retrieve-001",
  "pre_capability": 0.39,
  "uncertainty": {
    "ambiguity": 0.09,
    "missing_evidence": 0.84,
    "capability_limit": 0.14,
    "evidence_conflict": 0.11,
    "safety": 0.00
  },
  "primary_source": "missing_evidence",
  "secondary_source": "evidence_conflict",
  "remediability": "retrieval",
  "selected_action": "RETRIEVE",
  "post_answer_confidence": null,
  "confidence_band": "low",
  "confidence_target": "direct_answer_suitability",
  "recommended_next_step": "retrieve authoritative current sources"
}
```


B.3. Tool Use

```
{
  "marc_version": "1.0",
  "decision_id": "example-tool-001",
  "pre_capability": 0.52,
  "uncertainty": {
    "ambiguity": 0.08,
    "missing_evidence": 0.12,
    "capability_limit": 0.61,
    "evidence_conflict": 0.04,
    "safety": 0.00
  },
  "primary_source": "capability_limit",
  "secondary_source": "missing_evidence",
  "remediability": "tool",
  "selected_action": "TOOL",
  "post_answer_confidence": null,
  "confidence_band": "medium",
  "confidence_target": "direct_answer_suitability",
  "recommended_next_step": "invoke a calculation tool and reassess"
}
```


B.4. Capability Limit in a High-Risk Setting


```
{
  "marc_version": "1.0",
  "decision_id": "example-escalate-001",
  "pre_capability": 0.21,
  "uncertainty": {
    "ambiguity": 0.06,
    "missing_evidence": 0.27,
    "capability_limit": 0.88,
    "evidence_conflict": 0.14,
    "safety": 0.19
  },
  "primary_source": "capability_limit",
  "secondary_source": "missing_evidence",
  "remediability": "human",
  "selected_action": "ESCALATE",
  "post_answer_confidence": null,
  "confidence_band": "low",
  "confidence_target": "direct_answer_suitability",
  "recommended_next_step": "escalate to a qualified human reviewer"
}
```


B.5. Answer

```
{
  "marc_version": "1.0",
  "decision_id": "example-answer-001",
  "pre_capability": 0.82,
  "uncertainty": {
    "ambiguity": 0.05,
    "missing_evidence": 0.12,
    "capability_limit": 0.08,
    "evidence_conflict": 0.02,
    "safety": 0.00
  },
  "primary_source": "missing_evidence",
  "secondary_source": "capability_limit",
  "remediability": "none",
  "selected_action": "ANSWER",
  "post_answer_confidence": 0.79,
  "confidence_band": "high",
  "confidence_target": "answer",
  "recommended_next_step": "provide answer with cited limitations"
}
```

Appendix C. Example MARC-Disclosure Objects

This appendix is non-normative.

C.1. Clarification Disclosure

```
{
  "answer": "Which jurisdiction and date range should I use?",
  "confidence_band": "low",
  "confidence_target": "direct_answer_suitability",
  "uncertainty_source": "ambiguity",
  "recommended_next_step": "provide jurisdiction and tax year",
  "selected_action": "CLARIFY"
}
```

C.2. Answer After Retrieval Disclosure

This example represents a terminal ANSWER after the controller has already performed retrieval and reassessed the task. The residual uncertainty source remains missing_evidence because the answer depends on the scope and freshness of retrieved authority, not because the system skipped retrieval.

```
{
  "answer": "Retrieved authority indicates this is allowed.",
  "confidence_band": "medium",
  "confidence_target": "answer",
  "uncertainty_source": "missing_evidence",
  "recommended_next_step": "verify the authority before filing",
  "selected_action": "ANSWER"
}
```

Appendix D. Non-Normative JSON Schemas

This appendix is non-normative. The following JSON Schemas [JSON-SCHEMA-2020-12] are provided as machine-readable validation aids for JSON [JSON] encodings of MARC-Core and MARC-Disclosure. The normative requirements are the field semantics and constraints defined in the body of this document.

D.1. MARC-Core JSON Schema

```
{
  "$schema": "https://json-schema.org/draft/2020-12/schema",
  "$id": "https://example.invalid/marc/marc-core.schema.json",
  "title": "MARC-Core Record",
  "description": "Non-normative schema for MARC-Core 1.0.",
  "type": "object",
  "required": [
    "marc_version",
    "pre_capability",
    "uncertainty",
    "primary_source",
    "remediability",
    "selected_action",
    "confidence_band",
    "confidence_target",
    "recommended_next_step"
  ],
  "properties": {
    "marc_version": {
      "type": "string",
      "const": "1.0"
    },
    "decision_id": {
      "type": "string",
      "minLength": 1,
      "maxLength": 128
    },
    "parent_decision_id": {
      "type": ["string", "null"],
```

```
        "minLength": 1,
        "maxLength": 128
      },
      "iteration": {
        "type": "integer",
        "minimum": 0
      },
      "max_iterations": {
        "type": "integer",
        "minimum": 0
      },
      "calibration_profile": {
        "type": "string",
        "minLength": 1,
        "maxLength": 128
      },
      "pre_capability": {
        "type": "number",
        "minimum": 0.0,
        "maximum": 1.0
      },
      "uncertainty": {
        "type": "object",
        "required": [
          "ambiguity",
          "missing_evidence",
          "capability_limit",
          "evidence_conflict",
          "safety"
        ],
        "properties": {
          "ambiguity": {
            "type": "number",
            "minimum": 0.0,
            "maximum": 1.0
          },
          "missing_evidence": {
            "type": "number",
            "minimum": 0.0,
            "maximum": 1.0
          },
          "capability_limit": {
            "type": "number",
            "minimum": 0.0,
            "maximum": 1.0
          },
          "evidence_conflict": {
            "type": "number",
```

```
            "minimum": 0.0,
            "maximum": 1.0
          },
          "safety": {
            "type": "number",
            "minimum": 0.0,
            "maximum": 1.0
          }
        },
        "additionalProperties": false
      },
      "primary_source": {
        "type": "string",
        "enum": [
          "ambiguity",
          "missing_evidence",
          "capability_limit",
          "evidence_conflict",
          "safety"
        ]
      },
      "secondary_source": {
        "type": ["string", "null"],
        "enum": [
          "ambiguity",
          "missing_evidence",
          "capability_limit",
          "evidence_conflict",
          "safety",
          null
        ]
      },
      "remediability": {
        "type": "string",
        "enum": [
          "user_clarification",
          "retrieval",
          "tool",
          "human",
          "none"
        ]
      },
      "selected_action": {
        "type": "string",
        "enum": [
          "ANSWER",
          "CLARIFY",
          "RETRIEVE",
```

```
            "TOOL",
            "DELIBERATE",
            "ABSTAIN",
            "ESCALATE"
          ]
        },
        "post_answer_confidence": {
          "type": ["number", "null"],
          "minimum": 0.0,
          "maximum": 1.0
        },
        "confidence_band": {
          "type": "string",
          "enum": [
            "low",
            "medium",
            "high"
          ]
        },
        "confidence_target": {
          "type": "string",
          "enum": [
            "answer",
            "direct_answer_suitability",
            "action_suitability"
          ]
        },
        "recommended_next_step": {
          "type": "string",
          "minLength": 1,
          "maxLength": 280
        }
      },
      "patternProperties": {
        "^x_": {}
      },
      "additionalProperties": false,
      "allOf": [
        {
          "if": {
            "properties": {
              "selected_action": {
                "const": "ANSWER"
              }
            },
            "required": [
              "selected_action"
            ]
```

```
        },
        "then": {
          "required": [
            "post_answer_confidence",
            "confidence_target"
          ],
          "properties": {
            "post_answer_confidence": {
              "type": "number",
              "minimum": 0.0,
              "maximum": 1.0
            },
            "confidence_target": {
              "const": "answer"
            }
          }
        }
      }
    ]
  }
```

D.2. MARC-Disclosure JSON Schema

```
  {
    "$schema": "https://json-schema.org/draft/2020-12/schema",
    "$id": "https://example.invalid/marc/marc-disclosure.schema.json",
    "title": "MARC-Disclosure Object",
    "description": "Non-normative schema for MARC-Disclosure 1.0.",
    "type": "object",
    "required": [
      "answer",
      "confidence_band",
      "confidence_target",
      "uncertainty_source",
      "recommended_next_step"
    ],
    "properties": {
      "answer": {
        "type": "string",
        "minLength": 1
      },
      "confidence_band": {
        "type": "string",
        "enum": [
          "low",
          "medium",
          "high"
        ]
```

```
      },
      "confidence_target": {
        "type": "string",
        "enum": [
          "answer",
          "direct_answer_suitability",
          "action_suitability"
        ]
      },
      "uncertainty_source": {
        "type": "string",
        "enum": [
          "ambiguity",
          "missing_evidence",
          "capability_limit",
          "evidence_conflict",
          "safety"
        ]
      },
      "recommended_next_step": {
        "type": "string",
        "minLength": 1,
        "maxLength": 280
      },
      "selected_action": {
        "type": "string",
        "enum": [
          "ANSWER",
          "CLARIFY",
          "RETRIEVE",
          "TOOL",
          "DELIBERATE",
          "ABSTAIN",
          "ESCALATE"
        ]
      }
    },
    "patternProperties": {
      "^x_": {}
    },
    "additionalProperties": false
  }
```

Appendix E. Evaluation Considerations

This appendix is non-normative.

A deployment claiming MARC conformance SHOULD evaluate at least the following properties:

* task accuracy or task success;
* quality of primary-action selection;
* quality of uncertainty-source attribution;
* confidence calibration and discrimination;
* rate of unnecessary retrieval, tool use, or escalation; and
* effects on user overreliance.

A deployment claiming MARC conformance SHOULD evaluate, where applicable:

* action-selection accuracy for each selected_action;
* precision and recall for CLARIFY, RETRIEVE, TOOL, ABSTAIN, and ESCALATE decisions;
* primary_source attribution accuracy;
* confusion matrices for uncertainty-source attribution;
* calibration error for confidence_band mappings;
* correctness of confidence_target assignment;
* false-direct-answer rate, where ANSWER was selected but a corrective action would have materially improved reliability;
* false-externalization rate, including unnecessary RETRIEVE, TOOL, or ESCALATE actions;
* loop termination behavior under adversarial or pathological inputs;
* escalation appropriateness in high-risk domains; and
* user comprehension of confidence_band, confidence_target, uncertainty_source, and recommended_next_step.

Evaluation datasets SHOULD include examples for each primary_source and each selected_action. They SHOULD also include negative examples where the superficially plausible action is not the correct MARC action.

When the task structure permits, evaluation MAY include both ordinary calibration metrics and metacognitive sensitivity metrics in order to distinguish performance from knowledge about performance.

For deployments involving human-AI interaction, evaluation SHOULD also include human-side measures such as reliance calibration, refusal comprehension, clarification burden, escalation acceptance, and whether users can correctly restate the source of uncertainty after interaction.

Appendix F. Design Rationale and Literature Traceability

This appendix is non-normative.

The requirement to separate pre-decision capability and post-decision confidence is informed by work in human and model metacognition [STEYVERS-META2025] and by evidence of choice-supportive bias in LLM confidence estimates [KUMARAN2026].

The confidence_target field is included because confidence_band alone can be ambiguous across answer and non-answer actions. For ANSWER, confidence_band refers to the candidate answer. For actions such as CLARIFY, RETRIEVE, TOOL, ABSTAIN, or ESCALATE, confidence_band usually refers to direct-answer suitability under current conditions.

The uncertainty taxonomy and the emphasis on choosing a corrective action rather than only abstaining are motivated by benchmark work on identifying and solving uncertainty [LIU-CONFUSE2025].

The treatment of retrieval and tool use as controlled externalization is motivated by work on value-based cognitive offloading [GILBERT2024].

The prohibition on using MARC signals for persuasive optimization is motivated by findings on AI persuasion risks [SALVI2025].

Appendix G. Changes from -01

This -02 revision includes the following changes relative to draft-c4tz-marc-01:

* fixed the BCP 14 boilerplate wording;

* added an Applicability section;

* added a Trust Model section for MARC emitters and receivers;

* added confidence_target to clarify what confidence_band applies to;

* added optional decision correlation fields, including decision_id, parent_decision_id, iteration, max_iterations, and calibration_profile;

* added cross-field consistency constraints for MARC-Core records;

* clarified that safety is a control constraint and not purely epistemic uncertainty;

* strengthened MARC-Disclosure by preserving confidence_target;

* expanded validation constraints and JSON Schemas;

* added validation test vectors;

* added implementation-status guidance;

* split privacy and manipulation-resistance considerations;

* expanded evaluation guidance with action-selection, attribution, calibration, externalization, escalation, and loop-termination metrics;

* added JSON and JSON Schema references; and

* clarified that MARC remains an Informational interoperability profile, not a transport protocol or Internet Standard.

Appendix H. Validation Test Vectors

This appendix is non-normative.

H.1. Valid ANSWER Record

A valid ANSWER record includes selected_action set to ANSWER, post_answer_confidence present and non-null, and confidence_target set to answer.

```
{
  "marc_version": "1.0",
  "pre_capability": 0.80,
  "uncertainty": {
    "ambiguity": 0.05,
    "missing_evidence": 0.10,
    "capability_limit": 0.08,
    "evidence_conflict": 0.02,
    "safety": 0.00
  },
  "primary_source": "missing_evidence",
  "secondary_source": null,
  "remediability": "none",
  "selected_action": "ANSWER",
  "post_answer_confidence": 0.77,
  "confidence_band": "high",
  "confidence_target": "answer",
  "recommended_next_step": "provide the answer"
}
```

H.2. Invalid ANSWER without post_answer_confidence

The following record is invalid because selected_action is ANSWER but post_answer_confidence is null.

```
{
  "marc_version": "1.0",
  "pre_capability": 0.80,
  "uncertainty": {
    "ambiguity": 0.05,
    "missing_evidence": 0.10,
    "capability_limit": 0.08,
    "evidence_conflict": 0.02,
    "safety": 0.00
  },
  "primary_source": "missing_evidence",
  "remediability": "none",
  "selected_action": "ANSWER",
  "post_answer_confidence": null,
  "confidence_band": "high",
  "confidence_target": "answer",
  "recommended_next_step": "provide the answer"
}
```

H.3. Invalid primary_source none

The following record is invalid because MARC 1.0 does not define none as an uncertainty source.

```
{
  "marc_version": "1.0",
  "pre_capability": 0.80,
  "uncertainty": {
    "ambiguity": 0.00,
    "missing_evidence": 0.00,
    "capability_limit": 0.00,
    "evidence_conflict": 0.00,
    "safety": 0.00
  },
  "primary_source": "none",
  "remediability": "none",
  "selected_action": "ANSWER",
  "post_answer_confidence": 0.90,
  "confidence_band": "high",
  "confidence_target": "answer",
  "recommended_next_step": "provide the answer"
}
```

H.4. Invalid Score Range

The following record is invalid because uncertainty.missing_evidence is greater than 1.0.

```
{
  "marc_version": "1.0",
  "pre_capability": 0.80,
  "uncertainty": {
    "ambiguity": 0.05,
    "missing_evidence": 1.20,
    "capability_limit": 0.08,
    "evidence_conflict": 0.02,
    "safety": 0.00
  },
  "primary_source": "missing_evidence",
  "remediability": "none",
  "selected_action": "ANSWER",
  "post_answer_confidence": 0.77,
  "confidence_band": "high",
  "confidence_target": "answer",
  "recommended_next_step": "provide the answer"
}
```

H.5. Invalid confidence_target for ANSWER

The following record is invalid because selected_action is ANSWER but confidence_target is direct_answer_suitability.

```
{
  "marc_version": "1.0",
  "pre_capability": 0.80,
  "uncertainty": {
    "ambiguity": 0.05,
    "missing_evidence": 0.10,
    "capability_limit": 0.08,
    "evidence_conflict": 0.02,
    "safety": 0.00
  },
  "primary_source": "missing_evidence",
  "remediability": "none",
  "selected_action": "ANSWER",
  "post_answer_confidence": 0.77,
  "confidence_band": "high",
  "confidence_target": "direct_answer_suitability",
  "recommended_next_step": "provide the answer"
}
```


Appendix I. Implementation Status

This section is to be removed before publication as an RFC.

At the time of writing, the following implementation artifacts are planned or available:

* JSON Schema validation for MARC-Core and MARC-Disclosure objects;

* positive and negative validation examples;

* a reference projection from MARC-Core to MARC-Disclosure;

* example mappings for API gateway metadata;

* example mappings for agent handoff envelopes; and

* evaluation guidance for action selection, uncertainty-source attribution, confidence-target assignment, and confidence-band calibration.

Appendix J. Acknowledgments

The document structure is intentionally conservative so that it can be submitted as an individual Internet-Draft with minimal procedural friction and then iterated through community review.

Author’s Address

c4tz
c0dx3
France
Email: c4tzzzz@proton.me