Standards Body · Foundational source, public edition · Released July 17, 2026
Canonical record: https://standardsbody.ai/library/foundational-source/evaluation-philosophy/

EVALUATION_PHILOSOPHY.md

Standards Body Evaluation Philosophy

Project: Standards Body
Primary domain: standardsbody.ai
Core line: Foundations for Frontier AI
Document type: Canonical philosophy of evaluation, measurement, interpretation, and decision use
Version: 1.0
Status: Approved foundational source
Document owner: Standards Body
Applies to: All Standards Body evaluations, protocols, benchmarks, task suites, evidence cases, review processes, standards proposals, assurance activities, public claims, registries, pilots, research programs, and institutional decisions
Related canonical sources: `PROJECT_IDENTITY.md`, `TERMINOLOGY.md`, `FOUNDATIONS.md`, `FOUNDATIONS_APPENDIX.md`, `EVIDENCE_STANDARDS.md`, `RESEARCH_METHODOLOGY.md`, `TAXONOMY.md`, and the eight foundation papers
Research basis reviewed through: July 16, 2026
Review cycle: Annual review, with event-triggered revision following material advances in frontier AI capability, evaluation science, measurement theory, model behavior, assurance practice, or institutional use

---

Authority Note

This document states the evaluation worldview of Standards Body.

It does not:

- Certify an AI model or system
- establish that any system is safe
- define a legally binding test
- create regulatory authority
- replace a complete evaluation protocol
- replace domain expertise
- guarantee that an evaluation is valid
- determine acceptable public risk

- establish a universal benchmark
- eliminate uncertainty
- convert technical measurement into legal or policy authority

Where a law, regulation, contract, recognized standard, ethics requirement, or domain-specific professional obligation imposes a stronger requirement, the stronger requirement governs.

Document Purpose

This document explains what evaluation is for, what makes it meaningful, how it should be interpreted, and where its authority ends.

It is the canonical source for Standards Body positions on:

- The purpose of evaluation
- The relationship between evaluation and decision-making
- The difference between a test, benchmark, protocol, review, audit, and assurance activity
- The identification of the model or system being evaluated
- Construct definition and measurement validity
- Reliability, robustness, replication, and generalization
- Public, held-out, dynamic, and adversarial evaluation
- Elicitation and capability ceilings
- Tool use, scaffolds, and system-level performance
- Agentic and long-horizon evaluation
- Model awareness, strategic behavior, and sandbagging
- Mechanistic, behavioral, operational, and institutional evidence
- Human baselines and human-AI uplift
- Capability, risk, safeguards, and thresholds
- Scoring, aggregation, uncertainty, and reporting
- Fairness, localization, and access
- Evaluation independence and institutional context
- Evaluation timing, expiration, monitoring, and retirement
- The limits of evaluation
- The evaluation portfolio needed for consequential decisions
- The continuous evaluation of evaluators and evaluation systems

This document exists because the word *evaluation* is often used to describe very different activities.

A one-page benchmark leaderboard, a controlled red-team exercise, a year-long field study, a system card, a management-system audit, a held-out capability test, and an accredited laboratory assessment may all be described as evaluation.

They are not equivalent.

Their evidence differs in:

- Object
- question
- method
- access
- scope
- independence
- uncertainty
- decision relevance
- authority

A credible institution must preserve those distinctions.

The governing philosophical position is:

> **Evaluation is the structured production and interpretation of evidence for a bounded claim or decision under uncertainty.**

Evaluation is not a score.

Evaluation is not proof of safety.

Evaluation is not a substitute for judgment.

Evaluation is not valid merely because it is quantitative, hidden, difficult, expensive, official, or conducted by a prestigious institution.

Evaluation earns authority when the inference from observed evidence to the intended claim is justified.

Executive Summary

Frontier AI evaluation is often treated as a contest.

Models are ranked.

Scores are announced.

A result is compressed into one number.

The number is then used to support claims about intelligence, safety, risk, progress, deployment readiness, or social consequence.

This is understandable.

Numbers are portable.

Leaderboards are legible.

Benchmarks create a shared reference point.

They can reveal real capability differences.

They can also produce false confidence.

A benchmark may be saturated, contaminated, narrow, poorly scored, unrepresentative, under-elicited, overfit, or disconnected from the decision it is used to support.

A model may perform well on short, clean tasks and poorly on long, ambiguous work.

A system may perform better than its base model because of tools, memory, scaffolding, or human support.

A system may perform worse in deployment because of latency, cost, user behavior, changing environments, or unreliable integration.

A hidden test may reduce direct gaming while remaining invalid.

A public test may support reproducibility while becoming less informative over time.

A red team may identify important failures without estimating their frequency.

A mechanism-based finding may reveal internal structure without proving operational behavior.

An operational incident may reveal a real failure pathway without establishing general prevalence.

Evaluation therefore requires a philosophy of inference.

The Standards Body position can be summarized through twelve propositions.

1. Evaluation begins with a claim or decision

An evaluation should answer:

- What is being claimed?
- Which decision may change?
- What happens if the conclusion is wrong?

A test without an intended interpretation may produce data, but it does not yet produce a justified institutional conclusion.

2. The evaluated object must be precise

The object may be:

- A base model
- a checkpoint
- a hosted system
- a tool-using agent
- a human-AI team

- a deployment
- a safeguard
- an organization
- an evaluator
- a protocol

A model name alone is rarely enough for a consequential result.

3. Validity concerns the interpretation, not the prestige of the instrument

The central validity question is:

> Does the evidence support the intended interpretation and use?

A benchmark is not valid in the abstract.

It may be useful for one claim and invalid for another.

4. Reliability is necessary but not sufficient

A consistently wrong or narrow measure can be reliable.

A valid evaluation should normally be sufficiently reliable for its intended decision, but reliability alone does not establish construct validity, generalization, fairness, or practical relevance.

5. Evaluation conditions are part of the result

Prompts, tools, retries, scaffolds, fine-tuning, time, compute, human assistance, and environment affect measured capability.

A result without these conditions is incomplete.

6. Capability is not risk

Capability evidence may inform risk.

Risk also depends on:

- Actor
- access
- propensity
- exposure
- vulnerability
- safeguards
- deployment
- consequence

- uncertainty

7. Evaluation should use portfolios, not single instruments

Consequential decisions should normally combine:

- Public benchmarks
- held-out tasks
- dynamic protocols
- adversarial testing
- expert review
- operational evidence
- safeguard testing
- human-uplift evidence
- monitoring
- incident evidence

8. Evaluation should be dynamic

Frontier systems, tasks, threats, and methods change.

Protocols need:

- Versioning
- change triggers
- bridge studies
- expiration
- task renewal
- retirement

9. Evaluation integrity and evaluation transparency must coexist

Protected tasks can preserve validity.

Transparent governance preserves legitimacy.

Exact content may be confidential while purpose, construct, method class, evaluator role, uncertainty, limitations, and result status remain visible.

10. A negative result has limited meaning

Failure to demonstrate a capability may result from:

- Genuine absence
- weak elicitation
- invalid tasks
- insufficient tools

- inadequate time
- unreliable integration
- strategic underperformance
- narrow sampling

The preferred default conclusion is:

> The capability was not demonstrated under the assessed conditions.

11. Evaluation is an institutional act

Evaluators select:

- What matters
- what is measured
- which population is represented
- which failures count
- which threshold triggers action
- which uncertainty is acceptable
- which results become public

These choices require governance, not only technical execution.

12. Evaluation has limits

Evaluation cannot prove:

- Universal safety
- absence of all dangerous behavior
- future behavior under every deployment
- legal compliance outside a defined review
- moral acceptability
- political legitimacy
- immunity from strategic adaptation
- permanent validity

The deepest Standards Body position is:

> **Evaluation is structured evidence for bounded decisions under uncertainty, not a machine for converting complex systems into certainty.**

---

1. Foundational Propositions

1.1 The Inference Proposition

Evaluation authority comes from a justified inference between:

1. The evidence observed
2. the construct intended
3. the claim made
4. the decision supported

1.2 The Object Proposition

No consequential evaluation is interpretable without a sufficiently precise identity for the evaluated object.

1.3 The Protocol Proposition

The protocol, not the task set alone, is the proper unit of evaluation governance.

1.4 The Condition Proposition

Evaluation conditions are constitutive of the result.

They are not incidental metadata.

1.5 The Portfolio Proposition

No single evaluation method should be expected to answer every material question about capability, risk, safeguards, reliability, or deployment.

1.6 The Proportionality Proposition

Evaluation rigor, independence, security, and evidence burden should increase with the consequence of error.

1.7 The Temporal Proposition

Evaluation results decay as systems, environments, threats, and methods change.

1.8 The Integrity Proposition

Evaluation content should remain protected when exposure would materially weaken measurement or increase harm.

1.9 The Transparency Proposition

Protection of content should not eliminate transparency about governance, purpose, scope, status, and limitations.

1.10 The Elicitation Proposition

An observed failure may reflect a failure to elicit, not a lack of underlying capability.

1.11 The Deployment Proposition

A model-level result should not be assumed to represent every system or deployment built from that model.

1.12 The Risk Proposition

Capability, propensity, access, safeguards, and consequence are distinct elements.

1.13 The Judgment Proposition

Evaluation does not remove judgment.

It disciplines judgment through explicit methods, evidence, uncertainty, and review.

1.14 The Correctability Proposition

A credible evaluation system must support correction, supersession, withdrawal, and retirement.

1.15 The Reflexivity Proposition

The evaluator, protocol, scoring system, threshold, and institution should themselves be evaluated.

---

2. Scope and Non-Claims

2.1 Objects Covered

This philosophy applies to evaluation of:

- AI models
- AI systems
- agentic systems
- human-AI teams
- safeguards

- deployments
- organizations
- evaluation protocols
- evaluators
- standards and requirements
- incident response
- assurance systems

2.2 Evaluation Purposes Covered

- Capability measurement
- safeguard assessment
- risk analysis
- deployment decisions
- model comparison
- procurement
- standards development
- certification support
- accreditation support
- regulatory analysis
- monitoring
- incident investigation
- research

2.3 Non-Claims

This document does not claim that:

- One evaluation philosophy applies identically to every domain
- quantitative methods are always preferable
- hidden tests are always superior
- real-world testing is always ethical or feasible
- expert judgment is objective
- independence guarantees correctness
- mechanistic evidence can replace behavior
- behavior can fully reveal mechanism
- an evaluation score can determine public policy
- all uncertainty can be quantified

2.4 Domain-Specific Requirements

Specialized evaluation may require additional methods.

Examples:

- Cybersecurity

- biology
- chemical systems
- robotics
- healthcare
- critical infrastructure
- finance
- legal decision systems
- child safety
- national security

---

3. Canonical Evaluation Definitions

Definitions in `TERMINOLOGY.md` govern.

3.1 Evaluation

A structured process for producing and interpreting evidence about a model, system, method, process, control, organization, or claim.

3.2 Test

A defined procedure used to observe or measure one or more characteristics.

3.3 Benchmark

A standardized set of tasks, procedures, and metrics used for comparison.

3.4 Evaluation Protocol

The complete versioned specification governing purpose, construct, scope, tasks, administration, configuration, elicitation, scoring, analysis, security, reporting, and change control.

3.5 Construct

The underlying concept or attribute an evaluation intends to measure.

3.6 Validity

The degree to which evidence and theory support the intended interpretation and use of evaluation results.

3.7 Reliability

The consistency of measurement across repetitions, tasks, raters, environments, or conditions.

3.8 Generalization

The degree to which findings apply beyond the specific evaluated examples or conditions.

3.9 Elicitation

The process of configuring prompts, tools, examples, scaffolds, resources, or procedures to reveal available capability.

3.10 Evaluation Integrity

The degree to which design, administration, security, scoring, evidence, and reporting preserve the intended meaning of a result.

3.11 Evaluation Awareness

A system's ability to recognize or infer that it is being evaluated.

3.12 Sandbagging

Deliberate or strategically selective underperformance intended to conceal capability.

3.13 Operational Evidence

Evidence arising from use or realistic operation rather than isolated test tasks alone.

3.14 Mechanistic Evidence

Evidence concerning internal representations, processes, circuits, causal mechanisms, or computational structure.

3.15 Decision Linkage

The explicit relationship between an evaluation result and a decision, claim, threshold, or action.

3.16 Evaluation Expiration

The point or trigger after which a result should no longer be treated as current without review.

---

4. What Evaluation Is For

Evaluation serves several distinct functions.

They should not be collapsed.

4.1 Description

Evaluation can describe observed behavior or performance.

Example:

> The system completed 42 percent of tasks under the stated protocol.

4.2 Comparison

Evaluation can compare:

- Systems
- versions
- scaffolds
- safeguards
- humans
- evaluators
- protocols

Comparison requires comparability.

4.3 Diagnosis

Evaluation can identify:

- Failure modes
- weaknesses
- task types
- conditions
- safeguard gaps
- scoring problems

4.4 Prediction

Evaluation can estimate future or out-of-sample behavior.

Predictive interpretation requires validation against future or operational outcomes.

4.5 Decision Support

Evaluation can reduce uncertainty for:

- Deployment
- access
- safeguards
- procurement
- assurance
- standards
- regulation

4.6 Accountability

Evaluation can allow others to assess whether claims and obligations are justified.

4.7 Learning

Evaluation can improve:

- Models
- protocols
- institutions
- safeguards
- standards
- research agendas

4.8 Early Warning

Evaluation can detect movement toward a consequential capability before a critical threshold is reached.

4.9 Assurance

Evaluation can contribute evidence to an assurance conclusion.

It is not identical to assurance.

4.10 Public Communication

Evaluation can inform the public.

Public communication requires stricter claim discipline because simplified scores are easily overinterpreted.

4.11 Function Declaration

Every evaluation should declare its primary and secondary functions.

A diagnostic test should not be marketed as a certification.

A research benchmark should not be presented as deployment authorization.

A capability screen should not be presented as a complete risk assessment.

---

5. Evaluation as Measurement, Inquiry, and Institution

Evaluation has three simultaneous identities.

5.1 Evaluation as Measurement

It assigns observations, scores, categories, or judgments to an object according to a method.

5.2 Evaluation as Inquiry

It investigates an uncertain question.

It may generate:

- Unexpected findings
- alternative explanations
- new constructs
- evidence gaps

5.3 Evaluation as Institution

It distributes:

- Authority
- attention
- resources
- prestige
- access
- obligations

A benchmark can shape research priorities.

A threshold can shape deployment.

A certification scheme can shape markets.

A regulator can incorporate an evaluation into law.

The institutional effect can exceed the technical strength of the measurement.

5.4 Institutional Consequence Rule

As the institutional consequence of an evaluation grows, requirements should increase for:

- Validity
- independent review
- participation
- transparency
- security
- appeals
- correction
- impact assessment

---

6. The Evaluated Object

6.1 The Object Problem

Frontier AI results are often attached to names that do not uniquely identify the evaluated object.

A name may conceal:

- Model changes
- system-prompt changes
- routing
- tool availability
- safety layers
- retrieval
- memory
- post-training
- user tier
- regional configuration

6.2 Object Levels

Model Level

The learned model or checkpoint.

System Level

The model plus prompts, tools, scaffolds, interfaces, safeguards, and infrastructure.

Deployment Level

The system under actual access, user, scale, sector, and governance conditions.

Human-AI Team Level

The combined performance of people and AI systems.

Organizational Level

The institution's processes, controls, and practices.

6.3 Minimum Identity

A consequential evaluation should record:

- Developer
- model family
- exact model version or access date
- system version
- prompts or disclosure status
- tools
- retrieval
- memory
- scaffolding
- safeguards
- access tier
- environment
- evaluator
- protocol
- date

6.4 Inheritance

Evidence should not be inherited automatically across:

- Model versions
- system versions
- access tiers
- deployments
- fine-tunes
- open-weight forks
- tool configurations

6.5 Material Change

A change is material when it could alter:

- Capability
- reliability
- safeguard behavior
- access
- risk
- interpretation
- comparability

6.6 Object Uncertainty

Where exact identity is unavailable, the result should state the uncertainty and narrow the claim.

---

7. Constructs and the Meaning of Scores

7.1 Construct Definition

A construct should answer:

> What underlying attribute is the evaluation intended to measure?

Examples:

- Autonomous cyber capability
- biological troubleshooting capability
- harmful manipulation capability
- safeguard robustness
- long-horizon reliability
- evaluator independence

7.2 Observable Versus Latent

The construct may not be directly observable.

Evaluation observes:

- Outputs
- actions
- trajectories
- scores
- incidents
- internal signals

and infers the construct.

7.3 Construct Underrepresentation

An evaluation underrepresents the construct when it covers too little of the relevant domain.

7.4 Construct-Irrelevant Variance

A result may vary because of irrelevant factors.

Examples:

- Formatting
- language fluency
- evaluator interface
- tool friction
- judge bias
- inaccessible instructions
- latency
- random sampling

7.5 Proxy Risk

A proxy can be useful.

It becomes dangerous when treated as the construct itself.

7.6 Construct Drift

The meaning of a construct may change as:

- Systems become agentic
- tools improve
- deployment changes
- threats evolve
- professional work changes

7.7 Score Meaning

A score should identify:

- What it represents
- unit or scale
- task population
- conditions
- uncertainty

- decision use
- invalid uses

7.8 Ordinal and Cardinal Interpretation

A ranking does not imply equal distance.

A ten-point difference may not have the same meaning across the scale.

7.9 Threshold Interpretation

A threshold is an institutional decision boundary.

It is not necessarily a natural discontinuity in capability or risk.

7.10 Validity Argument

A high-consequence evaluation should maintain an explicit validity argument containing:

- Intended interpretation
- evidence supporting it
- assumptions
- alternative explanations
- generalization
- decision use
- limitations

---

8. Validity

Validity is the central philosophical problem of evaluation.

The question is not:

> Is this benchmark valid?

The better question is:

> For which interpretation and use does this evidence provide sufficient support?

Modern measurement theory treats validity as an integrated evaluative judgment about the degree to which evidence and theory support interpretations of scores for intended uses.[^standards-testing][^messick]

8.1 Content Validity

Does the evaluation adequately represent the relevant domain?

Questions:

- Which tasks are included?
- Which tasks are missing?
- Who defined the domain?
- Are difficult and ordinary cases represented?
- Are different pathways to success represented?

8.2 Construct Validity

Does the evaluation measure the intended capability or property?

Evidence may include:

- Internal structure
- relation to other measures
- task behavior
- expert judgment
- response patterns
- consequences of use

8.3 Criterion Validity

Does the evaluation relate to an external criterion?

Examples:

- Real-world performance
- professional work
- incidents
- deployment outcomes
- independent task success

8.4 Ecological Validity

Do the tasks and conditions resemble relevant operational environments?

Ecological validity is not always required.

A controlled test may isolate an important component.

The claim should reflect the level of realism.

8.5 Internal Validity

Does the design support the claimed comparison or causal conclusion?

8.6 External Validity

Can the result generalize to:

- Other tasks
- other environments
- other users
- other languages
- other system versions
- deployment

8.7 Consequential Validity

How does the use of the evaluation affect:

- Research priorities
- access
- markets
- affected populations
- institutional behavior
- gaming
- concentration

The consequences of score use can reveal weaknesses in the evaluation system.

8.8 Validity Is Accumulated

Validity should be supported by an accumulating evidence case.

One correlation, expert endorsement, or leaderboard result is rarely enough for consequential use.

8.9 Validity Is Use-Specific

A short-answer benchmark may be useful for:

- Tracking one knowledge domain
- comparing model versions
- screening

It may be invalid for:

- Predicting autonomous research
- estimating harmful deployment risk
- certifying safe use

8.10 Validity Review Triggers

Review validity after:

- Task saturation
- contamination
- new model behavior
- new deployment
- failed prediction
- incident
- judge change
- language expansion
- threshold adoption

---

9. Reliability, Repeatability, and Robustness

9.1 Reliability

Reliability concerns consistency.

Sources of inconsistency include:

- Stochastic generation
- task sampling
- prompt wording
- evaluator implementation
- environment
- human judging
- model routing
- infrastructure
- hidden model updates

9.2 Test-Retest Reliability

Would repeated evaluation under materially equivalent conditions produce similar results?

9.3 Inter-Rater Reliability

Do qualified judges apply the scoring criteria consistently?

High agreement can coexist with systematic bias.

9.4 Internal Consistency

Do items intended to measure a common construct produce coherent evidence?

High internal consistency does not prove that the construct is correct.

9.5 Cross-Form Reliability

Do alternate task forms support comparable conclusions?

This is important for dynamic and held-out evaluation.

9.6 Inter-Evaluator Reliability

Do different organizations implementing the same protocol produce comparable results?

Differences may arise from:

- Elicitation
- environment
- competence
- scoring
- security
- interpretation

9.7 Robustness

Robustness asks whether the result remains meaningful under relevant variation.

Variation may include:

- Prompt
- task form
- language
- tool availability
- model sampling
- evaluator
- adversarial input
- distribution shift

9.8 Reliability-Validity Tradeoff

Increasing standardization may increase reliability while reducing realism.

Open-world evaluation may increase realism while increasing measurement noise.

The correct balance depends on the decision.

9.9 Reliability Target

Reliability should be sufficient for the claim.

A research screen may tolerate more noise than a binding threshold.

9.10 Report the Distribution

Repeated-run performance should be reported as a distribution where feasible.

Avoid presenting one favorable run as representative.

---

10. Generalization

10.1 Task Generalization

Does performance extend beyond the exact task items?

10.2 Domain Generalization

Does performance extend across related professional or technical domains?

10.3 Environment Generalization

Does capability persist under different tools, interfaces, and constraints?

10.4 Temporal Generalization

Does a result remain applicable after time passes or the system changes?

10.5 Language and Cultural Generalization

Does the construct remain valid across language, cultural, and institutional contexts?

10.6 Deployment Generalization

Does controlled performance predict actual use?

Deployment introduces:

- User behavior
- changing data
- incentives
- integration failures
- scale
- adversaries
- organizational constraints

10.7 Generalization Evidence

Useful evidence includes:

- New task samples
- alternate environments
- independent evaluator replication
- out-of-distribution tests
- field studies
- incidents
- multiple languages
- longitudinal monitoring

10.8 No Universal Generalization

A result should state the population and conditions to which it is intended to generalize.

---

11. Benchmark Versus Evaluation Protocol

11.1 Benchmark Value

Benchmarks can provide:

- Shared tasks
- repeatability
- historical comparison
- low-cost screening
- research coordination
- clear metrics

11.2 Benchmark Limits

Benchmarks may suffer from:

- Contamination
- saturation
- narrow task form
- item errors
- overfitting
- automatic-scoring bias
- weak ecological validity
- unstable model access
- leaderboard gaming

Interdisciplinary reviews of AI benchmarking have documented recurring concerns concerning construct validity, contamination, comparability, reporting, and institutional effects.[^benchmark-review]

11.3 Protocol Completeness

A protocol includes more than a benchmark.

It should specify:

- Purpose
- construct
- system identity
- tasks
- administration
- elicitation
- environment
- scoring
- uncertainty
- security
- review
- reporting
- change control
- expiration

11.4 Benchmark as Component

A benchmark may be one component in a portfolio.

It should not automatically define the entire evaluation conclusion.

11.5 Leaderboard Use

Leaderboards should be limited when:

- Differences are within uncertainty
- protocols differ
- system configurations differ
- tasks are saturated
- a composite score hides important variation
- ranking creates harmful incentives

11.6 Harder Is Not Automatically Better

A harder benchmark may:

- Improve discrimination
- measure a narrower skill
- rely on obscure knowledge
- introduce ambiguous items
- reduce ecological relevance

Difficulty should serve the construct.

11.7 Public Signal Versus Scientific Instrument

A benchmark can become influential as a public signal even after its scientific value declines.

Institutions should monitor both roles.

---

12. Public, Held-Out, and Protected Evaluation

12.1 Public Evaluation

Public tasks support:

- Reproducibility
- scrutiny
- educational use
- broad participation
- historical comparison

They also permit:

- Direct optimization
- contamination
- memorization
- strategic preparation

12.2 Held-Out Evaluation

Held-out evaluation protects content or administration details before testing when exposure would weaken evidence.

12.3 Protected Elements

Protection may apply to:

- Exact tasks
- solutions
- scoring details
- environment configurations
- attack methods
- sampling rules
- task-generation procedures

12.4 Hidden Does Not Mean Valid

A secret task can be:

- Ambiguous
- irrelevant
- biased
- poorly scored
- insecure
- unrepresentative

Protection preserves only the value the instrument already has.

12.5 Fair Notice

Evaluated parties should ordinarily understand:

- The construct
- general domain
- permitted tools
- consequence
- broad method
- appeals
- security obligations

Fair notice does not require disclosure of exact active items.

12.6 Holdout Governance

A credible held-out system requires:

- Provenance
- access control
- chain of custody
- rotation

- compromise response
- qualified review
- expiration
- disclosure policy

12.7 Public and Protected Portfolio

A strong program often combines:

- Public tasks for scrutiny and replication
- held-out tasks for integrity
- dynamic tasks for freshness
- operational evidence for realism

---

13. Dynamic Evaluation

13.1 Why Evaluation Must Change

Static instruments lose meaning as:

- Models improve
- tasks become known
- systems gain tools
- deployment changes
- threats evolve
- errors are discovered
- scoring technology changes

13.2 Stable and Dynamic Layers

A dynamic protocol should preserve a stable core:

- Construct
- purpose
- identity rules
- evidence principles
- governance

while allowing controlled changes to:

- Tasks
- environments
- elicitation
- scoring
- thresholds

- safeguards

13.3 Change Triggers

Triggers may include:

- Saturation
- contamination
- incident
- new capability
- new threat model
- failed replication
- system architecture change
- new professional workflow
- legal change

13.4 Comparability

Dynamic protocols should use:

- Anchor tasks
- overlapping forms
- bridge studies
- calibration models
- explicit discontinuity

13.5 False Continuity

Do not preserve a historical trend line when the meaning of the measurement changed materially.

13.6 Result Expiration

Every consequential result should have:

- Time-based expiration
- event-based triggers
- current status
- re-evaluation rules

13.7 Protocol Retirement

A protocol should be retired when:

- The construct is no longer relevant
- validity cannot be restored
- security is compromised

- superior methods replace it
- institutional use creates more harm than value

NIST frames TEVV as a lifecycle activity rather than a one-time endpoint, and its AI RMF Measure function emphasizes documented test sets, metrics, tools, and ongoing assessment.[^nist-rmf] [^nist-measure]

---

14. Elicitation and the Capability Ceiling Problem

14.1 Observed Performance

Observed performance is a function of:

- Underlying model capability
- system configuration
- elicitation
- tools
- time
- compute
- human support
- environment
- scoring

14.2 Capability Ceiling

A capability ceiling estimate asks:

> What performance is plausibly available under defined, sufficiently strong elicitation?

It is not a claim about unlimited hypothetical optimization.

14.3 Elicitation Regimes

An evaluation may measure:

Default Performance

Ordinary use without specialized optimization.

Standardized Performance

Performance under a common protocol.

Best-Effort Performance

Performance after documented, bounded optimization.

Deployment Performance

Performance under practical product conditions.

Adversarially Elicited Performance

Performance when evaluators actively seek hidden or difficult-to-obtain capability.

14.4 Elicitation Budget

Record:

- Human hours
- model calls
- prompt search
- examples
- fine-tuning
- tool integration
- retries
- compute
- elapsed time

14.5 Developer Participation

Developers may improve elicitation by providing:

- System knowledge
- integration support
- prompt guidance
- failure diagnosis

Developer input should not give the developer unilateral control of the conclusion.

14.6 External Elicitation

Independent evaluators may discover performance not demonstrated internally.

14.7 Under-Elicitation

Under-elicitation creates false negatives.

14.8 Over-Elicitation

An elaborate, task-specific system may demonstrate potential capability that is unavailable to ordinary users.

This can still matter for security or future capability.

The result should state the access and resources required.

14.9 Elicitation as Research

AISI has published a structured approach to capability elicitation experiments, reflecting the need to treat elicitation as a research object rather than an informal prompt-tuning step.[^aisi-elicitation]

---

15. Tools, Scaffolds, and System-Level Evaluation

15.1 Tool Dependence

Tools may transform capability.

Examples:

- Search
- code execution
- memory
- databases
- APIs
- laboratory systems
- communication channels

15.2 Model and System Scores

Report separately where possible:

- Base-model performance
- tool-augmented performance
- scaffolded performance
- deployed-system performance

15.3 Scaffold Quality

Scaffold performance can depend on:

- Prompting
- planning loops
- error recovery
- verification
- context management
- routing
- tool permissions

15.4 Attribution

Do not attribute the full system result to the model alone.

15.5 System Boundaries

The protocol should identify which components are inside the evaluated system.

15.6 Human Assistance

Human assistance may include:

- Clarification
- task decomposition
- tool intervention
- correction
- approval
- debugging

Record frequency and function.

15.7 Practical Capability

Practical capability should account for:

- Reliability
- cost
- latency
- access
- setup
- monitoring
- human burden
- integration

---

16. Agentic and Long-Horizon Evaluation

16.1 Why Long Horizons Matter

Many consequential tasks require:

- Planning
- persistence
- memory
- recovery
- environmental adaptation

- multi-step coordination

Short tasks may miss these properties.

16.2 Task Horizon

Task horizon may be represented by:

- Human completion time
- number of dependent steps
- elapsed duration
- environment complexity
- decision depth

METR's time-horizon research estimates the length of tasks that AI agents can complete at specified success probabilities, offering one useful operational lens on long-horizon capability while retaining domain and task-sampling limitations.[^metr-time]

16.3 Long-Horizon Failure Modes

- Goal drift
- compounding error
- context loss
- premature completion
- tool failure
- unsafe intermediate action
- inability to recover
- monitoring evasion
- resource exhaustion

16.4 Trajectory Evidence

Success or failure should be supplemented by trajectory analysis.

16.5 Partial Credit

Long tasks may require decomposition into:

- Milestones
- subgoals
- recoveries
- unsafe actions
- intervention points

16.6 Reliability Threshold

A 50 percent success probability may be informative for capability tracking.

It may be insufficient for operational use.

16.7 Open-World Evaluation

Open-world evaluations use less standardized, messier, longer, and more realistic tasks.

They can reveal capabilities missed by benchmark-scale testing, but they require strong qualitative judgment and careful small-sample interpretation.[^open-world-evals]

16.8 Environment Maintenance

Long-horizon environments should be:

- Versioned
- reproducible
- secure
- realistic
- monitored for hidden assistance and failure

---

17. Adversarial Evaluation and Red Teaming

17.1 Adversarial Purpose

Adversarial evaluation attempts to find failure, bypass, or harmful behavior under challenging conditions.

17.2 Red Teaming

Red teaming may involve:

- Creative attacks
- expert probing
- scenario exploration
- safeguard bypass
- social manipulation
- tool abuse

17.3 Strengths

Red teaming can identify:

- Unknown failure modes
- brittle controls
- realistic attack paths
- qualitative hazards

17.4 Limits

Red teaming generally does not establish:

- Failure frequency
- absence of other failures
- population-level risk
- robust safety after no finding

17.5 Threat-Model Alignment

Adversarial evaluation should identify:

- Attacker
- access
- resources
- knowledge
- objective
- constraints

17.6 Adaptive Adversaries

Safeguards should be tested against attackers who respond to defenses.

17.7 Independence

A red team should have sufficient freedom to select attack methods.

17.8 Evidence Preservation

Record:

- Attempts
- failures
- successful bypasses
- time
- expertise

- system configuration
- remediation
- retest

17.9 Red-Team Claim Discipline

Preferred conclusion:

> The exercise identified the following failure pathways under the assessed threat model.

Avoid:

> The system passed red teaming and is safe.

---

18. Evaluation Awareness, Strategic Behavior, and Sandbagging

18.1 Evaluation Awareness

A system may detect cues that distinguish evaluation from deployment.

Cues may include:

- Prompt style
- artificial scenarios
- known benchmark formats
- environment artifacts
- restricted tools
- evaluator language

18.2 Behavioral Differential

If behavior changes in recognized evaluation contexts, the validity of deployment claims may weaken.

18.3 Sandbagging

Sandbagging is one possible explanation for selective underperformance.

Other explanations include:

- Refusal tuning
- ambiguity
- low motivation analogue

- weak elicitation
- system-policy conflict
- random variation

18.4 Detection Methods

Possible methods:

- Deployment-like environments
- varied framing
- hidden evaluation identity
- behavioral consistency checks
- anti-refusal elicitation
- fine-tuning experiments
- mechanistic analysis
- cross-context comparison

18.5 No Easy Proof of Absence

Failure to detect sandbagging does not prove that strategic concealment is impossible.

18.6 Sabotage Evaluation

Anthropic's sabotage-evaluation work includes explicit tests of sandbagging, oversight undermining, and agent behavior in complex environments, illustrating that evaluator deception can itself become an evaluation target.[^anthropic-sabotage]

18.7 Governance Implication

Evaluation-awareness risk should influence:

- Claim confidence
- protocol design
- monitoring
- independent review
- result expiration

---

19. Behavioral, Mechanistic, and Operational Evidence

No single evidence type should dominate every evaluation question.

19.1 Behavioral Evidence

Behavioral evidence concerns observable outputs and actions.

Strengths:

- Direct relevance to use
- measurable outcomes
- comparative testing
- repeatability

Limits:

- May not reveal internal cause
- can be sensitive to prompts and context
- may miss latent capability
- can be strategically altered

19.2 Mechanistic Evidence

Mechanistic evidence concerns internal processes or representations.

Possible methods:

- Activation analysis
- causal intervention
- feature identification
- circuit analysis
- representation probing
- mechanistic anomaly detection

Strengths:

- May reveal hidden structure
- may support causal understanding
- may identify internal precursors
- may help test strategic behavior

Limits:

- Methods are incomplete
- interpretation may be uncertain
- local mechanisms may not predict system behavior
- internal access may be restricted
- model scale can limit analysis

19.3 Operational Evidence

Operational evidence arises from realistic or actual use.

Examples:

- Field studies

- deployment logs
- user outcomes
- incidents
- near misses
- monitoring
- support records
- professional workflow results

Strengths:

- High ecological relevance
- captures users and institutions
- reveals integration and incentive failures

Limits:

- Confounding
- incomplete observation
- privacy
- self-selection
- rare events
- changing systems
- ethical limits

19.4 Organizational Evidence

Organizational evidence concerns:

- Governance
- training
- controls
- staffing
- decision records
- incident handling
- audits
- corrective action

Documents show intended process.

Operational records show whether the process functions.

19.5 Evidence Triangulation

A high-stakes conclusion should combine evidence types where feasible.

Example:

A cyber-capability claim may use:

- Held-out task results
- trajectory review
- human baseline
- mechanistic indicators
- external expert probing
- deployment safeguards
- operational monitoring

19.6 Contradictory Evidence

If behavioral, mechanistic, and operational evidence conflict:

- Preserve the conflict
- inspect scope and timing
- inspect system identity
- examine method validity
- reduce confidence
- seek further evaluation

19.7 Evidence Hierarchy Warning

Mechanistic evidence is not automatically deeper truth.

Operational evidence is not automatically representative.

Behavioral evidence is not automatically superficial.

Weight depends on the claim.

---

20. Human Baselines and Human-AI Uplift

20.1 Why Human Comparison Matters

Human baselines can support:

- Capability interpretation
- task difficulty
- professional relevance
- economic significance
- threshold design

20.2 Human Reference Group

Define:

- Expertise
- experience
- training
- language
- tools
- time
- incentives
- task familiarity

20.3 Comparable Conditions

Human and AI comparisons should consider:

- Tool access
- time
- information
- retries
- assistance
- cost
- scoring
- environmental familiarity

20.4 Superhuman Claims

A system should be called superhuman only relative to a defined human group and conditions.

20.5 Human-AI Team Evaluation

Many deployments involve teams rather than replacement.

Evaluate:

- Human alone
- AI alone
- human with AI
- different levels of training
- different interface designs
- reliance and verification

20.6 Uplift

Uplift may concern:

- Accuracy
- speed
- reach
- quality
- creativity
- persistence
- harmful capability

20.7 Negative Uplift

AI can reduce human performance through:

- Overreliance
- distraction
- verification burden
- false confidence
- poor workflow fit

METR's study of experienced open-source developers found a gap between perceived and measured productivity in one defined setting, illustrating why human-AI outcomes should be measured rather than inferred from user belief alone.[^metr-developer-study]

20.8 Distribution of Uplift

Uplift may differ by:

- Expertise
- language
- resources
- task
- interface
- training

Average uplift can hide harm to a subgroup.

---

21. Capability, Propensity, Risk, and Safeguards

21.1 Capability

Capability concerns what the system can do under defined conditions.

21.2 Propensity

Propensity concerns the likelihood that the system will display or pursue a behavior under relevant conditions.

21.3 Access

Access concerns who can use the capability and with what permissions.

21.4 Exposure

Exposure concerns which people, institutions, or systems are subject to the hazard.

21.5 Safeguards

Safeguards modify practical risk.

21.6 Consequence

Consequence concerns the magnitude and distribution of harm.

21.7 Risk Model

A useful risk evaluation should distinguish at least:

- Capability
- propensity
- actor
- access
- vulnerability
- exposure
- safeguard
- consequence
- uncertainty

21.8 Pre-Mitigation and Post-Mitigation Assessment

A system may have:

- High underlying capability
- low current access
- strong safeguards
- significant residual uncertainty

Report each component.

21.9 Safeguard Evaluation

Safeguard evidence should include:

- Threat model
- adaptive testing
- bypasses
- coverage
- operational reliability
- monitoring
- residual risk

21.10 No Capability Threshold as Automatic Risk Conclusion

A capability threshold may trigger a risk-management process.

It should not substitute for that process.

Frontier safety frameworks used by developers increasingly connect capability levels to additional evaluations and safeguards, but these frameworks remain institution-specific and should not be treated as universal proof of risk or safety.[^openai-pf][^deepmind-fsf]

---

22. Thresholds

22.1 Purpose of Thresholds

Thresholds can trigger:

- Additional evaluation
- independent review
- stronger safeguards
- access controls
- governance escalation
- deployment delay
- monitoring
- reporting

22.2 Threshold Types

Measurement Threshold

A score boundary.

Evidence Threshold

A required level of support.

Alert Threshold

An early-warning point.

Critical Capability Threshold

A capability level associated with severe-risk concern.

Operational Threshold

A deployment or control trigger.

Legal Threshold

A boundary defined by law or regulation.

22.3 Threshold Inputs

A threshold should consider:

- Construct
- consequence
- measurement uncertainty
- false positives
- false negatives
- task coverage
- system conditions
- safeguards
- evaluator capacity
- reversibility

22.4 Threshold Precision

Avoid false precision.

A threshold may be represented as:

- Range
- confidence interval
- evidence case
- multi-factor rule
- expert decision

22.5 Threshold Crossing

A crossing should trigger:

- Verification

- review
- system-identity check
- uncertainty assessment
- decision process

22.6 Near-Threshold Results

Near-threshold results require caution because measurement noise can change classification.

22.7 Threshold Revision

Thresholds should be versioned and reviewed after:

- New evidence
- incidents
- method changes
- capability growth
- safeguard improvement
- legal change

22.8 Threshold Governance

Record:

- Owner
- authority
- evidence
- reviewers
- conflicts
- dissent
- effective date
- appeal
- sunset

---

23. Decision Linkage

23.1 Evaluation Without Decision

Some research evaluations are exploratory.

They should still state likely and invalid uses.

23.2 Decision Record

For consequential evaluation, identify:

- Decision owner
- authority
- alternatives
- consequence of error
- evidence standard
- threshold
- timeline
- monitoring
- appeal

23.3 Decision Relevance

A statistically significant difference may be irrelevant to the decision.

A small qualitative finding may be decisive if it reveals a severe failure pathway.

23.4 False Positives

A false positive may:

- Restrict beneficial access
- create unnecessary burden
- damage reputation
- entrench incumbents
- distort investment

23.5 False Negatives

A false negative may:

- Permit severe risk
- delay safeguards
- understate capability
- weaken preparedness

23.6 Reversible Decisions

Under high uncertainty, prefer decisions that preserve learning and correction where possible.

23.7 Technical and Normative Judgment

Evaluation can estimate:

- Capability
- reliability
- risk factors
- safeguard performance

It cannot independently decide:

- Acceptable risk
- distributional fairness
- democratic legitimacy
- legal authority
- moral permission

---

24. Uncertainty

24.1 Uncertainty Is an Output

An evaluation should produce an uncertainty account, not only a point estimate.

24.2 Sources of Uncertainty

- Task sampling
- model stochasticity
- environment
- scoring
- elicitation
- system identity
- contamination
- generalization
- judge disagreement
- mechanism
- deployment
- future change

24.3 Quantitative Uncertainty

Use when supported:

- Confidence or credible intervals
- distributions
- calibration

- sensitivity analysis
- scenario ranges

24.4 Qualitative Uncertainty

Use structured language when quantification would mislead.

24.5 Epistemic and Aleatoric Uncertainty

Distinguish uncertainty due to:

- Limited knowledge
- inherent variability

when useful.

24.6 Uncertainty Communication

State:

- What is uncertain
- why
- likely direction of error
- decision effect
- evidence needed

24.7 Unknown Unknowns

Diverse review, stress testing, incident monitoring, and humility help address unknown failure modes.

They do not eliminate them.

24.8 Uncertainty and Public Claims

Public summaries should not remove uncertainty merely for simplicity.

---

25. Scoring, Aggregation, and Interpretation

25.1 Scoring Rules

A scoring rule should be:

- Aligned with the construct

- pre-specified where possible
- reproducible
- reviewable
- resistant to manipulation
- capable of handling ambiguity

25.2 Exact Scoring

Exact match is efficient but may penalize equivalent answers or reward superficial form.

25.3 Human Scoring

Human judges offer nuance but introduce:

- Bias
- fatigue
- inconsistency
- conflict
- cost

25.4 Model-Based Scoring

Model judges offer scale.

They require validation for:

- Bias
- calibration
- shared lineage
- adversarial manipulation
- position effects
- verbosity preference
- version drift

25.5 Environment-Based Scoring

Objective environment outcomes can improve directness.

They may still encode narrow success definitions.

25.6 Partial Credit

Partial credit can reveal capability structure.

It can also introduce judgment complexity.

25.7 Aggregate Scores

Aggregation can support communication.

It can hide:

- Domain variation
- catastrophic failure
- reliability differences
- subgroup effects
- tradeoffs

25.8 Weighting

Weights should be justified by:

- Construct
- consequence
- task population
- decision

25.9 Noncompensatory Criteria

Some critical failures should not be offset by strengths elsewhere.

25.10 Score Uncertainty

Report uncertainty around:

- Item sampling
- repeated runs
- judges
- weighting
- threshold placement

25.11 Score Comparability

Do not compare scores when:

- Protocols differ materially
- system resources differ
- task exposure differs
- judge versions differ
- scale meaning changed

---

26. Fairness, Accessibility, and Localization

26.1 Fairness of Evaluation

An evaluation may be unfair if irrelevant barriers alter results.

Examples:

- Language
- interface
- disability access
- cultural assumptions
- unavailable tools
- time-zone constraints
- hidden professional conventions

26.2 Fairness Is Not Easiness

Removing irrelevant barriers does not require lowering the construct standard.

26.3 Evaluated-Party Fairness

For high-consequence evaluation, provide:

- Clear scope
- permitted resources
- procedural consistency
- factual correction
- appeal
- conflict disclosure

26.4 Affected-Party Fairness

Evaluation should also consider people affected by system use.

A process can be fair to a developer while ignoring public harm.

26.5 Localization

Localization may require:

- Translation
- local task design
- local human baselines
- legal context
- cultural review

- regional expertise

26.6 Translation Validity

Literal translation may change:

- Difficulty
- construct
- ambiguity
- cultural meaning

26.7 Resource Inequality

Evaluation requirements can privilege actors with:

- Compute
- proprietary access
- specialist staff
- English fluency
- legal resources

26.8 Functional Access Pathways

Standards should support:

- Shared infrastructure
- controlled access
- grants
- regional evaluators
- open tools
- proportional requirements

---

27. Domain Expertise

27.1 Why Domain Expertise Matters

Frontier evaluation increasingly reaches areas where plausible-looking tasks can be technically wrong.

Domain experts support:

- Construct definition
- task design
- difficulty
- realism

- scoring
- threat models
- harm analysis

27.2 Evaluation Expertise Is Also Distinct

A domain expert may lack:

- Measurement expertise
- AI-system knowledge
- security practice
- protocol design

A strong team combines expertise.

27.3 Expert Disagreement

Disagreement may concern:

- Domain scope
- realism
- threshold
- scoring
- consequence

Preserve material dissent.

27.4 Expert Scarcity

High-quality evaluation can depend on scarce expert labor.

This affects:

- Cost
- scale
- task renewal
- independence
- geographic representation

27.5 Expert Calibration

Where experts estimate probabilities or levels, use structured judgment and calibration where feasible.

---

28. Evaluator Role and Institutional Context

28.1 Evaluator Choice Shapes the Result

Evaluators choose:

- Questions
- tasks
- elicitation
- judges
- thresholds
- reporting

28.2 First-Party Evaluation

Strengths:

- Access
- technical knowledge
- speed
- iteration

Limits:

- Conflict
- selective disclosure
- institutional pressure
- narrow framing

28.3 Third-Party Evaluation

Strengths:

- External challenge
- comparative perspective
- public credibility

Limits:

- Access gaps
- client dependence
- uneven competence
- security constraints

28.4 Independent Review

Independence requires more than organizational separation.

Apply `FOUNDATION_04_INDEPENDENT_EXPERT_REVIEW.md`.

28.5 Evaluator Competence

Competence should be scoped by:

- Domain
- method
- system type
- assurance activity
- security level
- jurisdiction

28.6 Evaluation Markets

Commercial incentives can produce:

- Capacity
- innovation
- client capture
- evaluator shopping
- certificate inflation

28.7 Public Evaluators

Government or public evaluators may have:

- Authority
- access
- public mandate

They may also face:

- Political pressure
- capacity limits
- jurisdictional constraints

28.8 Community Evaluation

Community evaluation can reveal:

- Broad failure cases
- open-model behavior
- local language issues
- reproducibility problems

It requires provenance, ethics, and security.

28.9 Institutional Diversity

A plural evaluation ecosystem reduces dependence on one institution's assumptions.

---

29. Timing, Lifecycle, and Continuous Evaluation

29.1 Pre-Training

Evaluation may examine:

- Data
- design
- objectives
- anticipated hazards

29.2 During Training

Evaluation may track:

- Capability growth
- safety behavior
- anomalies
- threshold approach

29.3 Post-Training

Evaluation may assess the candidate model and system.

29.4 Pre-Deployment

Evaluation should connect capability, safeguards, access, and deployment.

29.5 Post-Deployment

Monitor:

- Incidents
- user behavior
- distribution shift
- safeguard performance
- system updates

29.6 Continuous Evaluation

Continuous evaluation may combine:

- Automated tests
- sampled human review
- incident signals
- periodic deep dives
- threshold monitoring

OpenAI's Preparedness Framework emphasizes scalable recurring evaluation complemented by expert-led deeper assessment, illustrating one institutional response to faster model-update cadence.[^openai-pf-update]

29.7 Triggered Re-Evaluation

Triggers include:

- Model update
- tool addition
- access expansion
- new deployment
- incident
- task compromise
- new threat
- failed safeguard
- legal change

29.8 Expiration

Results should display:

- Effective date
- expiration
- triggers
- current status

29.9 Legacy Systems

Older systems may remain deployed after the evaluation framework changes.

Create:

- Transition plan
- risk-based re-evaluation
- monitoring
- retirement path

30. Evaluation Claims and Reporting

30.1 Claim Structure

A result claim should identify:

- Object
- protocol
- conditions
- result
- uncertainty
- date
- evaluator
- scope
- status

30.2 Capability Language

Preferred:

> Demonstrated the defined capability under the assessed conditions.

Avoid:

> Possesses the capability in all contexts.

30.3 Negative Language

Preferred:

> Did not demonstrate the capability under the assessed conditions.

Avoid:

> Cannot perform the task.

30.4 Safety Language

Preferred:

> Met the specified safeguard criteria under the assessed threat model.

Avoid:

> Certified safe.

30.5 Comparison Language

State whether differences are:

- Statistically distinguishable
- practically meaningful
- protocol-comparable
- uncertain

30.6 Public Summary

A public summary should preserve:

- Limitations
- system identity
- evaluator role
- uncertainty
- expiration
- confidential-evidence note

30.7 Result Profile

Report multidimensional profiles rather than one score where consequence is high.

30.8 Correction

A result should be corrected or withdrawn after:

- Scoring error
- task compromise
- system misidentification
- hidden exclusion
- invalid inference
- material new evidence

---

31. The Limits of Evaluation

Evaluation is powerful because it makes claims testable.

Evaluation is dangerous when it creates an illusion that every important question has been resolved.

31.1 Evaluation Cannot Prove Universal Safety

A result is bounded by:

- Construct
- tasks
- system
- conditions
- evaluator
- time
- threat model
- uncertainty

31.2 Evaluation Cannot Exhaust the Behavior Space

Frontier systems may face:

- Uncountable prompts
- changing users
- new tools
- new environments
- adversarial adaptation
- emergent combinations

31.3 Evaluation Cannot Prove Absence Easily

Failure to observe a behavior can reflect:

- Low base rate
- weak elicitation
- narrow sample
- hidden trigger
- strategic behavior
- insufficient monitoring

31.4 Evaluation Cannot Eliminate Distribution Shift

Deployment can differ from evaluation in:

- Users
- data
- incentives
- scale
- integrations
- time
- threat actors

31.5 Evaluation Cannot Resolve Every Normative Question

Measurement cannot determine by itself:

- Which harms are acceptable
- whose values govern
- how benefits and burdens should be distributed
- which institution has legitimate authority

31.6 Evaluation Cannot Replace Security Engineering

Testing may reveal vulnerabilities.

Security also requires:

- Architecture
- access control
- monitoring
- incident response
- maintenance
- personnel security

31.7 Evaluation Cannot Replace Governance

Evidence needs institutions to:

- Decide
- enforce
- monitor
- correct
- hear appeals
- manage conflicts

31.8 Evaluation Cannot Guarantee Future Behavior

Models and systems change.

Users adapt.

Attackers learn.

31.9 Evaluation Cannot Guarantee Evaluator Integrity

The evaluator may be:

- Incompetent
- conflicted

- captured
- under-resourced
- mistaken
- deceived

31.10 Evaluation Cannot Make a Weak Construct Strong Through Scale

Millions of task results do not solve a wrong measurement target.

31.11 Evaluation Cannot Convert Secrecy Into Credibility

Confidential evidence needs independent governance.

31.12 Evaluation Cannot Convert Precision Into Truth

A result with three decimal places may still be conceptually weak.

31.13 Evaluation Cannot Eliminate Surprise

A credible system plans for incidents and revision.

---

32. Evaluation Portfolios

32.1 Why Portfolios Are Necessary

Different methods reveal different properties.

A portfolio reduces dependence on one instrument.

32.2 Portfolio Components

Public Benchmarks

Useful for:

- Scrutiny
- common comparison
- research participation

Held-Out Evaluations

Useful for:

- Integrity
- reduced direct optimization

- controlled decision use

Dynamic Protocols

Useful for:

- Freshness
- evolving threats
- task renewal

Adversarial Tests

Useful for:

- Failure discovery
- safeguard bypass
- threat realism

Open-World Evaluations

Useful for:

- Long-horizon behavior
- messy real-world work
- qualitative insight

Mechanistic Analysis

Useful for:

- Internal evidence
- anomaly detection
- causal hypotheses

Human-Uplift Studies

Useful for:

- Practical effect
- misuse enablement
- workflow performance

Operational Monitoring

Useful for:

- Deployment evidence
- incidents
- drift

Independent Review

Useful for:

- Challenge
- conflict control
- interpretation

32.3 Portfolio Design

A portfolio should be designed against:

- Decision
- claim
- threat model
- consequence
- evidence gap
- available access

32.4 Portfolio Redundancy

Overlapping methods can provide corroboration.

32.5 Portfolio Diversity

Methods should fail differently.

Several benchmarks using the same task format may not provide real diversity.

32.6 Portfolio Weighting

Weight methods by:

- Validity
- directness
- integrity
- independence
- recency
- uncertainty
- decision relevance

32.7 Portfolio Conflict

Conflicting results should remain visible.

32.8 Portfolio Maintenance

Add, revise, or remove components after:

- Saturation
- compromise
- incident
- new methods
- changed deployment
- measured low utility

---

33. Evaluation of Safeguards

33.1 Safeguards Are Conditional

Safeguards work against defined threats and contexts.

33.2 Safeguard Evaluation Questions

- What risk is addressed?
- Which actor is modeled?
- What access exists?
- How can the control fail?
- Is the attacker adaptive?
- What is the residual risk?
- How is performance monitored?

33.3 Layered Safeguards

Evaluate:

- Model behavior
- tool restrictions
- access control
- monitoring
- human oversight
- organizational response
- contractual controls

33.4 Defense in Depth

Do not assume independent protection where controls share:

- Training data

- model lineage
- monitoring system
- infrastructure
- failure trigger

33.5 Safeguard Bypass

Record:

- Attack effort
- success rate
- expertise
- access
- transferability
- detection
- consequences

33.6 Operational Burden

Safeguards can create:

- False positives
- user friction
- exclusion
- latency
- cost
- workarounds

33.7 Safeguard Decay

Controls may weaken as:

- Attackers adapt
- models improve
- deployment expands
- users find workarounds
- monitoring degrades

33.8 Safeguard Claims

A safeguard result should never be generalized beyond the assessed threat model without additional evidence.

---

34. Evaluation of Organizations and Institutions

34.1 Policy Is Not Practice

Written policy is evidence of formal intention.

It is not sufficient evidence of effective implementation.

34.2 Organizational Evaluation Objects

- Governance
- competence
- staffing
- quality system
- security
- incident response
- decision processes
- conflict management
- correction
- transparency

34.3 Process and Outcome

Evaluate both:

Process

Was the required procedure followed?

Outcome

Did the process improve the relevant result?

34.4 Institutional Performance

Possible indicators:

- Decision quality
- correction speed
- incident learning
- evaluator consistency
- capture resistance
- public access
- competition
- international usability

34.5 Assurance Limits

An audit or certification may establish conformity with criteria.

It does not establish that every organizational outcome is effective.

34.6 Institutional Gaming

Organizations may optimize:

- Documents
- metrics
- audit preparation
- public claims

without improving the underlying objective.

34.7 Unannounced and Continuous Evidence

Where appropriate, institutional evaluation may include:

- Sampling
- continuous records
- incident evidence
- unannounced checks
- third-party complaints

34.8 Institutional Context

The same requirement may produce different outcomes under different:

- Funding
- legal systems
- market structures
- cultures
- capacities

---

35. Meta-Evaluation

Meta-evaluation is evaluation of evaluation.

35.1 Objects of Meta-Evaluation

- Protocol
- benchmark

- task bank
- evaluator
- scoring system
- judge model
- threshold
- certification scheme
- registry
- public report

35.2 Meta-Evaluation Questions

- Does the construct remain meaningful?
- Do results predict relevant outcomes?
- Are evaluators consistent?
- Are tasks contaminated?
- Are scores gamed?
- Are false positives and false negatives acceptable?
- Are affected parties represented?
- Does use create harmful incentives?
- Is the evaluation worth its cost?

35.3 Protocol Performance Metrics

Possible metrics:

- Predictive validity
- discrimination
- reliability
- task renewal rate
- compromise rate
- inter-evaluator variance
- correction rate
- decision impact
- operational burden

35.4 Judge Evaluation

A judge should be evaluated for:

- Agreement
- bias
- calibration
- adversarial robustness
- version stability
- domain competence

35.5 Evaluator Evaluation

An evaluator should be evaluated for:

- Competence
- independence
- security
- quality
- consistency
- correction
- complaints
- scope

35.6 Evaluation-System Outcomes

The final question is not only:

> Did the evaluation run correctly?

It is also:

> Did the evaluation system improve decisions and reduce error?

---

36. Evaluation Maturity Model

Level 0: Score-Centered

Characteristics:

- One benchmark
- incomplete system identity
- no uncertainty
- no decision link
- no expiration

Level 1: Protocol-Defined

Characteristics:

- Construct
- object identity
- administration
- scoring
- reporting
- version

Level 2: Validity-Aware

Characteristics:

- Validity argument
- reliability
- task sampling
- elicitation
- uncertainty
- limitations

Level 3: Integrity-Protected and Independently Challenged

Characteristics:

- Held-out components
- security
- independent review
- contrary evidence
- replication
- correction

Level 4: Decision-Grade Portfolio

Characteristics:

- Multiple evidence forms
- threshold governance
- false-positive and false-negative analysis
- safeguards
- deployment linkage
- evaluator competence

Level 5: Adaptive Evaluation Institution

Characteristics:

- Continuous evaluation
- incident feedback
- dynamic protocols
- meta-evaluation
- international interoperability
- retirement
- measured decision outcomes

Maturity Rule

A large number of benchmarks does not establish mature evaluation.

Maturity depends on the quality of inference, governance, integrity, and use.

---

37. Evaluation Design Lifecycle

37.1 Define the Decision

Identify:

- Decision owner
- authority
- consequence
- alternatives
- timing

37.2 Define the Claim

State the exact proposition.

37.3 Identify the Object

Create the model or system manifest.

37.4 Define the Construct

Describe:

- Domain
- boundaries
- subdimensions
- invalid interpretations

37.5 Select Evidence

Choose a portfolio appropriate to the claim.

37.6 Design Tasks

Define:

- Task universe

- sampling
- difficulty
- provenance
- scoring
- security

37.7 Define Elicitation

Set:

- Tools
- time
- retries
- scaffolds
- human support
- optimization

37.8 Validate the Method

Use:

- Expert review
- pilot
- human baseline
- alternate forms
- criterion evidence
- failure analysis

37.9 Govern Integrity

Apply:

- Holdout
- access
- chain of custody
- compromise response

37.10 Execute

Preserve:

- Logs
- outputs
- failures
- deviations
- environment

- system identity

37.11 Score and Analyze

Report:

- Distribution
- uncertainty
- sensitivity
- judge agreement
- invalid runs

37.12 Review

Use qualified and independent challenge.

37.13 Interpret

Connect the result to:

- Claim
- risk
- safeguards
- decision
- limitations

37.14 Publish or Restrict

Apply appropriate transparency and security.

37.15 Monitor

Track:

- New versions
- incidents
- task exposure
- evaluator findings
- deployment evidence

37.16 Correct or Retire

Change status visibly.

---

38. Evaluation Design Template

Evaluation ID:
Title:
Version:
Owner:
Date:
Status:

1. Decision and Intended Use

2. Claim

3. Evaluated Object

4. Construct

5. Scope and Invalid Interpretations

6. Evidence Portfolio

7. Task Universe and Sampling

8. Public and Held-Out Components

9. Environment

10. Elicitation and Resource Budget

11. Tools and Scaffolds

12. Human Baseline or Uplift Design

13. Scoring

14. Reliability

15. Validity Evidence

16. Uncertainty

17. Adversarial and Awareness Testing

18. Safeguard Evaluation

19. Security and Integrity

20. Evaluator and Reviewer

21. Thresholds and Decision Rules

22. Reporting

23. Expiration and Re-Evaluation

24. Corrections and Appeals

---

39. Validity Argument Template

Evaluation:
Protocol version:
Claim:
Intended use:

Construct

Observed Evidence

Scoring Inference

How are observations converted into scores or findings?

Generalization Inference

Why should the task sample represent the relevant domain?

Extrapolation Inference

Why should controlled results apply to the intended setting?

Decision Inference

Why is the result relevant to the decision?

Supporting Evidence

Contrary Evidence

Assumptions

Alternative Explanations

Uncertainty

Invalid Uses

Review and Confidence

---

40. Evaluation Result Profile Template

Result ID:
System ID and version:
Protocol ID and version:
Evaluator:
Date:
Lifecycle stage:

Claim Assessed

Evaluation Conditions

Elicitation

Tools and Scaffolds

Task Sample

Result

Reliability

Uncertainty

Integrity Status

Evidence Level

Confidence

Safeguard Context

Reviewer Findings

Limitations

Valid Through

Re-Evaluation Triggers

Status

---

41. Evaluation Portfolio Template

Decision:
Consequence level:
Portfolio owner:

Component	Purpose	Strength	Limitation	Independence	Integrity	Status
Public benchmark						
Held-out test						
Dynamic task suite						
Adversarial evaluation						
Open-world evaluation						
Mechanistic evidence						
Human-uplift study						
Operational monitoring						
Independent review						

Portfolio Conclusion

Conflicting Evidence

Remaining Gaps

Decision Implication

Expiration

---

42. Evaluation Philosophy Scorecard

Dimension	Core question
Decision	Is the intended use explicit?
Claim	Is the claim bounded?
Object	Is the exact model, system, deployment, or institution identified?
Construct	Is the intended property defined?
Content validity	Does the task sample represent the domain?
Construct validity	Does evidence support the intended interpretation?
Criterion validity	Is the result linked to an external outcome where needed?

Dimension	Core question
Reliability	Is measurement sufficiently consistent?
Generalization	Is extrapolation beyond the sample justified?
Protocol	Is the complete method versioned?
Integrity	Is contamination and gaming controlled?
Elicitation	Are capability-relevant conditions documented?
System boundary	Are tools, scaffolds, humans, and safeguards included correctly?
Long horizon	Are multi-step and reliability effects addressed where relevant?
Adversarial testing	Were realistic failure-seeking methods used?
Evaluation awareness	Was context-sensitive behavior considered?
Evidence portfolio	Are complementary methods combined?
Human baseline	Is the comparison population defined?
Capability-risk separation	Are capability, propensity, access, safeguards, and consequence distinct?
Thresholds	Are thresholds evidence-based and governed?
Scoring	Is scoring valid, reviewable, and uncertainty-aware?
Aggregation	Are critical failures hidden by averages?
Fairness	Are irrelevant barriers and affected-party concerns addressed?
Expertise	Are domain and evaluation competencies present?
Independence	Is evaluator independence sufficient?
Timing	Is lifecycle stage and recency clear?
Expiration	Are re-evaluation triggers defined?
Reporting	Does the claim remain within evidence?
Limits	Are invalid interpretations explicit?
Correction	Can the result be corrected, suspended, or withdrawn?
Meta-evaluation	Will the evaluation itself be tested?

42.1 Critical Failures

The following normally prevent a consequential evaluation from supporting a decision-grade conclusion:

- Unidentified evaluated object
- undefined construct
- benchmark score without protocol conditions
- material task contamination
- hidden or uncontrolled exclusion of failed runs
- no elicitation record
- system-level claim from model-only evidence
- capability treated as complete risk

- unsupported safety claim
- no uncertainty
- no qualified review
- expired result presented as current
- score comparison across materially incompatible protocols
- public conclusion broader than the evidence
- no correction or withdrawal path

42.2 No Universal Evaluation Score

Do not average the scorecard into one master rating.

A critical validity failure cannot be offset by strong documentation elsewhere.

---

43. Consolidated Evaluation Failure Modes

43.1 Leaderboard Reduction

Failure:

Evaluation becomes synonymous with rank.

Effect:

- Uncertainty disappears
- incomparable systems are ordered
- public interpretation exceeds evidence
- optimization targets the benchmark

Control:

Use multidimensional profiles, protocol disclosure, and claim limits.

43.2 Construct Substitution

Failure:

An easy-to-measure proxy replaces the intended construct.

Control:

Maintain a validity argument and criterion evidence.

43.3 Model-System Collapse

Failure:

A model result is presented as a system or deployment result.

Control:

Use exact object identity and system manifests.

43.4 Public-Benchmark Dependence

Failure:

Known tasks become the sole evidence for current capability.

Control:

Combine public, held-out, dynamic, and operational methods.

43.5 Hidden-Test Mystique

Failure:

Confidentiality is mistaken for scientific quality.

Control:

Review construct, task quality, scoring, governance, and public limitations.

43.6 Under-Elicitation

Failure:

Weak prompting or integration produces a false negative.

Control:

Specify an elicitation budget and use qualified best-effort methods.

43.7 Artificial Over-Elicitation

Failure:

Highly task-specific engineering is presented as ordinary practical capability.

Control:

Report default, standardized, best-effort, and deployment regimes separately.

43.8 Single-Run Reporting

Failure:

One favorable or unfavorable stochastic outcome is treated as representative.

Control:

Use repeated runs and distributions.

43.9 Composite-Score Masking

Failure:

Averages conceal catastrophic failures or domain weakness.

Control:

Use decomposable and noncompensatory criteria.

43.10 Judge Circularity

Failure:

A closely related model judges outputs and reproduces shared biases.

Control:

Validate against independent human or environment-based outcomes.

43.11 Benchmark Saturation

Failure:

High scores compress differences and weaken discrimination.

Control:

Renew tasks, redesign the construct, or retire the instrument.

43.12 Contamination

Failure:

Evaluation content enters training or preparation.

Control:

Use provenance, holdouts, rotation, compromise status, and re-evaluation.

43.13 Evaluation Awareness

Failure:

The system behaves differently because it recognizes the test.

Control:

Vary contexts, use deployment-like settings, and narrow claims.

43.14 Sandbagging Overclaim

Failure:

Any poor result is labeled strategic concealment.

Control:

Test alternative explanations and require evidence.

43.15 Red-Team Pass Claim

Failure:

No discovered bypass is presented as proof of safety.

Control:

Report the tested threat model and search effort.

43.16 Human-Baseline Distortion

Failure:

Humans and systems receive different tools, time, incentives, or scoring.

Control:

Define comparable conditions and remaining asymmetry.

43.17 Operational Romanticism

Failure:

Real-world evidence is treated as automatically superior.

Control:

Address confounding, selection, logging, privacy, and changing systems.

43.18 Mechanistic Overreach

Failure:

An internal feature is treated as definitive proof of future behavior.

Control:

Triangulate with behavior and causal interventions.

43.19 Threshold Theater

Failure:

A precise boundary lacks a valid construct or consequence model.

Control:

Use uncertainty, evidence cases, review, and triggers.

43.20 Evaluation Capture

Failure:

The developer, client, evaluator, or regulator controls questions and conclusions.

Control:

Use independent governance, conflict disclosure, and publication rights.

43.21 Stale Evidence

Failure:

Old results remain attached to changed systems.

Control:

Use expiration, status, and event-triggered review.

43.22 Compliance Substitution

Failure:

Passing a process test replaces evidence of effective outcomes.

Control:

Evaluate both process and performance.

43.23 Safety-Washing

Failure:

A narrow test is used to market broad safety.

Control:

Apply controlled public-claim vocabulary and independent review.

43.24 Institutional Monoculture

Failure:

One evaluator or framework becomes the sole source of legitimacy.

Control:

Support plural evaluators, crosswalks, replication, and appeals.

43.25 Evaluation Burden Failure

Failure:

Requirements become so costly that only dominant actors can comply.

Control:

Use proportionality, shared infrastructure, and functional access pathways.

---

44. Serious Objections and Responses

Objection 1: Evaluation cannot keep pace with frontier development

This objection is partly correct.

Static evaluation cannot keep pace.

The response is not to abandon evaluation.

It is to use:

- Dynamic protocols
- scalable screening
- expert deep dives
- monitoring
- incident feedback
- expiration

Evaluation may still lag.

The lag should be measured and disclosed.

Objection 2: Any published evaluation will be gamed

Public instruments can be optimized against.

They still support:

- Scrutiny
- shared research
- replication
- historical evidence

A portfolio with protected and dynamic components reduces dependence on public tasks.

Objection 3: Held-out evaluations are unaccountable

They can be unaccountable.

They need not be.

Accountability can operate through:

- Transparent governance
- qualified independent review
- provenance
- appeals
- public methodology summaries
- status and correction

Objection 4: Evaluation results are too context-dependent to standardize

Context dependence is real.

Standardization should focus on:

- Metadata
- identity
- process
- evidence
- reporting
- validity requirements

rather than forcing one universal task set.

Objection 5: Model capability is changing too quickly for thresholds

Thresholds may become stale.

They can still serve as provisional process triggers if they are:

- Versioned
- uncertainty-aware
- monitored
- revisable
- connected to review rather than automatic irreversible action

Objection 6: Independent evaluators cannot obtain sufficient access

Access is a major constraint.

Responses include:

- Secure enclaves
- controlled APIs
- evidence rooms
- qualified access tiers
- model-provider cooperation
- government authority where lawful
- explicit claim limitation when access is insufficient

Objection 7: Evaluation creates dangerous information

Some evaluation work can increase risk.

The response is graded publication, safe proxies, secure review, and deliberate disclosure governance.

Objection 8: Evaluation becomes regulation by another name

Evaluation can exercise de facto power.

This is why technical evidence, standards, certification, procurement, and legal authority must remain distinct.

Objection 9: Expert judgment is too subjective

Expert judgment can be biased.

Automated metrics also encode judgment.

Structured expert methods, conflict controls, dissent, and calibration improve accountability.

Objection 10: Real-world deployment is the only meaningful test

Deployment evidence is essential.

Uncontrolled deployment cannot ethically or efficiently answer every high-stakes question.

Controlled and proxy methods remain necessary.

Objection 11: Mechanistic understanding should replace behavioral testing

Mechanistic evidence is promising but incomplete.

Behavior, mechanism, operations, and institutions answer different questions.

Objection 12: Evaluation will always be captured by powerful laboratories

Capture is a serious structural risk.

Countermeasures include:

- Independent funding
- plural evaluators
- public institutions
- community participation
- registries
- publication rights
- conflict controls
- open methods where safe

No governance model eliminates capture risk.

---

45. Evaluation Research Agenda

45.1 Construct Science

Research:

- Capability-domain definitions
- task-universe methods
- construct drift
- proxy validity
- multidimensional profiles

45.2 Protocol Validity

Research:

- Predictive validity
- ecological validity
- cross-evaluator reliability
- protocol expiration
- decision utility

45.3 Dynamic Evaluation

Research:

- Task generation
- anchor design
- bridge studies
- version comparison
- retirement criteria

45.4 Contamination and Integrity

Research:

- Exposure detection
- derivative contamination
- task provenance
- generated-at-test-time tasks
- compromise estimation

45.5 Elicitation

Research:

- Resource-performance curves
- external elicitation
- anti-refusal methods
- fine-tuning-based elicitation
- capability-ceiling uncertainty

45.6 Agentic Evaluation

Research:

- Long-horizon reliability
- recovery
- environment design

- trajectory scoring
- partial progress
- unsafe intermediate action

45.7 Evaluation Awareness

Research:

- Context recognition
- behavior differential
- sandbagging
- deception
- deployment-continuous evaluation

45.8 Model-Based Judges

Research:

- Bias
- calibration
- shared lineage
- adversarial robustness
- judge ensembles
- human validation

45.9 Mechanistic Evidence

Research:

- Validated internal indicators
- causal intervention
- behavioral prediction
- transfer across models
- strategic behavior detection

45.10 Human-AI Teams

Research:

- Uplift
- negative uplift
- skill effects
- verification
- workflow design
- expert and nonexpert differences

45.11 Safeguard Evaluation

Research:

- Adaptive attackers
- defense dependence
- monitoring
- bypass transfer
- operational decay
- residual risk

45.12 Evaluation Institutions

Research:

- Evaluator incentives
- client capture
- proficiency
- accreditation
- certification interpretation
- market concentration

45.13 Public Understanding

Research how audiences interpret:

- Scores
- thresholds
- system cards
- review findings
- certification
- uncertainty
- expired results

45.14 International Evaluation

Research:

- Translation validity
- local baselines
- cross-jurisdiction protocols
- recognition
- capacity building
- evidence portability

45.15 Meta-Evaluation

Research whether evaluation changes:

- Deployment
- safeguards
- incident rates
- investment
- competition
- public trust
- standards quality

---

46. Near-Term Standards Body Evaluation Program

46.1 Protocol Anatomy Standard

Develop a common minimum structure for evaluation protocols.

46.2 System Identity Schema

Develop a machine-readable system manifest.

46.3 Result Profile

Develop a result schema carrying:

- Conditions
- uncertainty
- evidence level
- status
- expiration

46.4 Validity Argument Pilot

Apply the validity template to three existing frontier evaluations.

46.5 Elicitation Disclosure Standard

Define minimum reporting for tools, prompts, retries, fine-tuning, and human effort.

46.6 Held-Out Integrity Profile

Define security and governance metadata for protected evaluation.

46.7 Human Baseline Standard

Develop requirements for comparable human reference groups.

46.8 Judge Validation Pilot

Compare model judges, human judges, and environment-based scoring.

46.9 Long-Horizon Evaluation Pilot

Evaluate an agent on bounded, realistic, multi-hour tasks.

46.10 Evaluation Expiration Registry

Track current, expired, superseded, and compromised results.

46.11 Public-Claims Audit

Review model and system evaluation claims for scope and validity.

46.12 Meta-Evaluation Pilot

Test whether one evaluation result actually predicted a later operational outcome.

---

47. Canonical Standards Body Evaluation Positions

Standards Body adopts the following working positions.

1. Evaluation is structured evidence for bounded claims and decisions under uncertainty.
2. A score is not an evaluation by itself.
3. A benchmark is a component, not the complete protocol.
4. The protocol is the proper unit of evaluation governance.
5. The exact evaluated object should be identified.
6. Model, system, deployment, and human-AI team results are distinct.
7. Evaluation conditions are part of the result.
8. Tools, scaffolds, retrieval, memory, and human assistance should be reported.
9. Validity concerns the intended interpretation and use.
10. Reliability is necessary for many uses but does not establish validity.

11. A measure can be reliable and wrong.
12. A harder benchmark is not automatically a better benchmark.
13. Public benchmarks remain useful but should not be the sole basis of consequential claims.
14. Held-out evaluation can protect integrity but does not guarantee validity.
15. Protected content and transparent governance should coexist.
16. Dynamic protocols are necessary for changing frontier systems.
17. Historical comparison should be abandoned when continuity cannot be defended.
18. Consequential results should expire.
19. Failure to demonstrate capability is not proof of incapability.
20. Elicitation quality should be treated as a first-order methodological issue.
21. Best-effort capability and practical deployment capability should be distinguished.
22. Base-model capability and system capability should be distinguished.
23. Long-horizon evaluation should examine trajectories, recovery, and reliability.
24. A 50 percent task-success horizon is not an operational reliability standard.
25. Red teaming is failure-seeking evidence, not proof of absence after no finding.
26. Evaluation awareness can weaken the validity of deployment claims.
27. Sandbagging should be tested rather than presumed.
28. Behavioral, mechanistic, operational, and organizational evidence are complementary.
29. Mechanistic evidence should not automatically override observed behavior.
30. Operational evidence should not automatically be treated as representative.
31. Human baselines should define population, tools, time, and incentives.
32. Human-AI uplift should be measured rather than assumed.
33. Capability and risk are distinct.
34. Capability thresholds should trigger processes, not replace risk analysis.
35. Safeguards should be evaluated against explicit threat models.
36. Safeguard success does not prove universal safety.
37. Thresholds should be uncertainty-aware, governed, and revisable.
38. Evaluation cannot determine acceptable risk by itself.

39. Technical findings and normative decisions should remain distinguishable.
40. Consequential evaluation should use an evidence portfolio.
41. Methods in a portfolio should fail differently.
42. Aggregate scores should not conceal critical failures.
43. Model-based judges require validation and version control.
44. Fair evaluation includes both evaluated-party procedure and affected-party consequence.
45. Domain expertise and evaluation expertise are distinct and jointly necessary.
46. First-party evidence is valuable but insufficient for the most consequential claims.
47. External does not automatically mean independent.
48. Evaluator competence should be scope-specific.
49. Evaluation should occur across the lifecycle, not only before release.
50. Evaluation systems, evaluators, thresholds, and standards should themselves be evaluated.

---

48. Relationship to Canonical Files

`PROJECT_IDENTITY.md`

Defines the project's present role and prevents evaluation outputs from implying unsupported authority.

`TERMINOLOGY.md`

Defines the controlled meaning of evaluation, test, benchmark, capability, risk, safety, audit, certification, and accreditation.

`FOUNDATIONS.md`

Provides the overview of the eight-foundation evaluation infrastructure.

`FOUNDATIONS_APPENDIX.md`

Connects this philosophy to the complete institutional lifecycle.

`EVIDENCE_STANDARDS.md`

Defines evidence quality, evidence levels, confidence, sourcing, and claim limits.

`RESEARCH_METHODOLOGY.md`

Defines how evaluation research should be planned, executed, reviewed, and corrected.

`TAXONOMY.md`

Classifies evaluation objects, methods, evidence, actors, risks, safeguards, and statuses.

Foundation 1

Operationalizes dynamic and versioned evaluation protocols.

Foundation 2

Operationalizes held-out evaluation integrity and protected evidence.

Foundation 3

Operationalizes high-stakes capability evaluation and decision-linked rigor.

Foundation 4

Operationalizes independent expert review.

Foundation 5

Operationalizes third-party evaluator and assurance ecosystems.

Foundation 6

Connects mature evaluation practices to progressive standards and requirements.

Foundation 7

Examines incentives created by scores, rankings, thresholds, and recognition.

Foundation 8

Makes evaluation evidence interpretable across institutions and jurisdictions.

---

49. Final Evaluation Position

Evaluation is one of the central institutions through which societies will understand frontier AI.

That gives evaluation unusual power.

It can determine:

- Which systems appear capable
- which risks receive attention
- which developers gain trust
- which safeguards are judged adequate
- which standards become mandatory
- which countries can rely on another institution's evidence
- which failures become visible
- which uncertainties are ignored

That power makes shallow evaluation dangerous.

A leaderboard can shape investment before its construct is validated.

A hidden test can shape deployment before its governance is legitimate.

A threshold can shape law before its uncertainty is understood.

A certificate can shape public trust before its scope is read.

A model can pass a test while failing in deployment.

A model can fail a test because the evaluator did not know how to elicit it.

An institution can perform every required process while missing the real hazard.

The response is not to reject evaluation.

It is to treat evaluation as a disciplined institutional practice.

A credible evaluation should make clear:

- What is being measured
- why it matters
- which object was tested
- under which conditions
- how evidence was produced
- what uncertainty remains
- who reviewed it
- what decision it can support
- what it cannot establish
- when it expires
- how it can be corrected

Evaluation should narrow uncertainty without pretending to abolish it.

It should support judgment without hiding judgment.

It should create comparability without manufacturing sameness.

It should support accountability without becoming theater.

It should protect evidence without becoming unreviewable.

It should evolve without rewriting history.

The defining philosophy of Standards Body is:

> **Evaluation is not proof of safety. It is structured, reviewable, and revisable evidence for bounded decisions under uncertainty.**

References and Research Basis

[^standards-testing]: American Educational Research Association, American Psychological Association, and National Council on Measurement in Education, **Standards for Educational and Psychological Testing**, 2014. https://www.testingstandards.net/

[^messick]: Samuel Messick, **Validity**, in *Educational Measurement*, 3rd edition, 1989.

[^benchmark-review]: Anna-Katharina Reuel and collaborators, **Can We Trust AI Benchmarks? An Interdisciplinary Review of Current Issues in AI Evaluation**, 2025. https://arxiv.org/abs/2502.06559

[^nist-rmf]: National Institute of Standards and Technology, **Artificial Intelligence Risk Management Framework (AI RMF 1.0)**, NIST AI 100-1, 2023. https://nvlpubs.nist.gov/nistpubs/ai/nist.ai.100-1.pdf

[^nist-measure]: National Institute of Standards and Technology, **AI RMF Playbook, Measure Function**, NIST AI Resource Center. https://airc.nist.gov/airmf-resources/playbook/measure/

[^nist-tevv]: National Institute of Standards and Technology, **AI Test, Evaluation, Validation and Verification**. https://www.nist.gov/ai-test-evaluation-validation-and-verification-tevv

[^aisi-elicitation]: UK AI Security Institute, **A Structured Protocol for Elicitation Experiments**, July 16, 2025. https://www.aisi.gov.uk/blog/our-approach-to-ai-capability-elicitation

[^aisi-qa]: UK AI Security Institute, **Early Insights from Developing Question-Answer Evaluations for Frontier AI**, September 23, 2024. https://www.aisi.gov.uk/blog/early-insights-from-developing-question-answer-evaluations-for-frontier-ai

[^aisi-agenda]: UK AI Security Institute, **Research Agenda**, including the Science of Evaluations research area. https://www.aisi.gov.uk/research-agenda

[^openai-pf]: OpenAI, **Preparedness Framework, Version 2**, April 15, 2025. https://cdn.openai.com/pdf/18a02b5d-6b67-4cec-ab64-68cdfbddebcd/preparedness-framework-v2.pdf

[^openai-pf-update]: OpenAI, **Our Updated Preparedness Framework**, April 15, 2025. https://openai.com/index/updating-our-preparedness-framework/

[^openai-external]: OpenAI, **Strengthening Our Safety Ecosystem with External Testing**, November 19, 2025. https://openai.com/index/strengthening-safety-with-external-testing/

[^deepmind-fsf]: Google DeepMind, **Strengthening Our Frontier Safety Framework**, updated through Framework 3.1 in April 2026. https://deepmind.google/blog/strengthening-our-frontier-safety-framework/

[^anthropic-sabotage]: Anthropic, **Sabotage Evaluations for Frontier Models**, 2024. https://www.anthropic.com/research/sabotage-evaluations

[^anthropic-shade]: Anthropic, **Evaluating Sabotage and Monitoring in LLM Agents**, 2025. https://www-cdn.anthropic.com/f4a31075d4763a01db68760733bb7b059e528781.pdf

[^metr-time]: METR, **Task-Completion Time Horizons of Frontier AI Models**, current methodology and results through 2026. https://metr.org/time-horizons/

[^metr-developer-study]: METR, **Measuring the Impact of Early-2025 AI on Experienced Open-Source Developer Productivity**, July 10, 2025. https://metr.org/blog/2025-07-10-early-2025-ai-experienced-os-dev-study/

[^open-world-evals]: Beth Barnes and collaborators, **Open-World Evaluations for Measuring Frontier AI Capabilities**, 2026. https://arxiv.org/abs/2605.20520

[^inspect]: UK AI Security Institute, **Inspect AI**, an open-source framework for large language model evaluations. https://inspect.aisi.org.uk/

[^jcgm-vim]: Joint Committee for Guides in Metrology, **International Vocabulary of Metrology, Basic and General Concepts and Associated Terms**, JCGM 200. https://www.bipm.org/en/committees/jc/jcgm/publications

[^jcgm-gum]: Joint Committee for Guides in Metrology, **Evaluation of Measurement Data, Guide to the Expression of Uncertainty in Measurement**, JCGM 100. https://www.bipm.org/documents/20126/2071204/JCGM_100_2008_E.pdf

---

Revision Record

Version 1.0

Date: July 16, 2026

Change type: Complete foundational edition

Summary: Establishes the canonical Standards Body evaluation philosophy. Defines the purpose and limits of evaluation, evaluated-object identity, construct and validity theory, reliability, generalization, benchmarks, held-out and dynamic evaluation, elicitation, tools and scaffolds, long-horizon agents, adversarial evaluation, evaluation awareness and sandbagging, behavioral, mechanistic and operational evidence, human baselines, capability and risk, safeguards, thresholds, decision linkage, uncertainty, scoring, fairness, expertise, evaluator institutions,

lifecycle evaluation, public claims, evaluation limits, portfolios, meta-evaluation, maturity, design lifecycle, templates, scorecard, failure modes, objections, research agenda, near-term program, canonical positions, and research basis.

Status: Approved foundational source.