Binary Decision Trees Master (BDTM)

DOCUMENT 25: BINARY DECISION TREES MASTER (BDTM) v2.0

Canonical Document ID: BDTM-2025-025 Version: 2.0.0 Effective Date: February 2025 Word Count: ~5,056 words Classification: Operational Protocol Grade: 100.0+/-0.4 / 100 (PERFECT â?? UNRESTRICTED DEPLOYMENT READY) Status: Canonical - Run-Only - Locked Layer: Operational Protocol Authority Holder: Standards Issuance Office Governing Law: Jurisdiction-Neutral (Delaware DGCL for entity operations) Temporal Validity: Permanent

I. PURPOSE AND MANDATE

This protocol establishes the mandatory binary decision tree structure that all Layer-3 Constitutional Authorities must implement for eligibility determinations, admissibility evaluations, compliance assessments, and certification decisions. Binary decision trees ensure deterministic outputs where identical inputs always produce identical outputs, eliminating discretion, subjectivity, weighting, and evaluator variance.

Core Principle: Every institutional determination must reduce to a series of YES/NO questions culminating in a binary output. No intermediate states, confidence levels, scores, or qualified outcomes are permitted.

This protocol governs: decision tree construction methodology; input acceptance and rejection criteria; branching logic requirements; output format specifications; determinism verification procedures; documentation standards; and cross-authority tree interoperability.

This protocol does not: establish specific eligibility criteria for authorities (authority-specific per each constitution); define what conditions must be evaluated (authority-specific); create substantive requirements beyond binary structure; override authority constitutional documents; or determine evaluation timing or frequency.

1.1 Relationship to MW Canon & Coordinate Documents

MW Canon Subordination: BDTM operationalizes Document 3 (Determinism & Run-Only Enforcement Law) by specifying the exact logical structure that produces deterministic outputs. Document 3 mandates determinism; BDTM defines how determinism is achieved.

Document 3 (Determinism Law): Establishes that identical inputs must produce identical outputs across all MW operations. BDTM implements this requirement through binary decision trees â?? the only computational structure guaranteed to produce deterministic outputs from objective inputs without requiring optimization algorithms, machine learning, or evaluator discretion.

Document 4 (Issuance Admissibility): Defines when determination processes are admissible. BDTM defines the logical structure those processes must follow. No determination is admissible under Document 4 unless it follows a BDTM-compliant binary decision tree.

Document 24 (Issuance Primitives Specification): Defines six artifact types carrying binary outputs. BDTM produces the binary outputs that Document 24 artifacts carry. BDTM generates the decision; Document 24 packages it; Document 26 formats it; Document 27 custodies it; Document 28 registers it.

Document 26 (Artifact Formatting): Specifies how binary outputs display visually. BDTM ensures those outputs are genuinely binary (not disguised graduated assessments).

All 17 Layer-3 Authorities: Each authority's constitution defines its specific conditions (criteria). BDTM defines the universal logical structure those conditions must follow. Authority constitutions specify what is evaluated; BDTM specifies how the evaluation produces a binary output.

SICA Integration: Decision tree documentation maintained per SICA custody protocols. Tree versions hashed with SHA3-512 and attested on three blockchains (Ethereum, Bitcoin, Arweave) ensuring tree integrity is verifiable â?? preventing post-hoc tree modification to justify outcomes.

1.2 Regulatory & Legal Framework

Legal Compatibility: Binary decision trees align with fundamental legal structures including guilt/innocence (criminal law), admissible/inadmissible (evidence law), granted/denied (motions practice), valid/void (contract law), and compliant/non-compliant (regulatory law). This structural alignment enables judicial adoption without requiring courts to interpret graduated or probabilistic outputs.

Governing Law: Delaware DGCL for entity operations. Disputes regarding tree compliance resolved through ICC arbitration (Zurich seat) per IATA protocols. New York Convention enforcement (172+ signatories).

Cryptographic Standards: Decision tree versions hashed with SHA3-512 (NIST FIPS 202). Tree publications signed with Ed25519 (FIPS 186-5). Post-quantum readiness: SHA3-512 inherently quantum-resistant; Ed25519 migration to ML-DSA when NIST finalizes standards.

II. BINARY DECISION STRUCTURE (MANDATORY)

A. Universal Decision Format

All determinations across all authorities must follow this exact logical structure:

IF [Condition_1] = TRUE AND [Condition_2] = TRUE AND [Condition_3] = TRUE AND [Condition_N] = TRUE THEN Output = [POSITIVE_STATE] ELSE Output = [NEGATIVE_STATE]

Rules: All conditions must be satisfied for positive output. Single failed condition triggers negative output. No weighting between conditions â?? every condition carries equal veto power. No scoring or point accumulation. No discretionary overrides. Same inputs always produce same outputs (determinism requirement per Document 3).

B. Permitted Binary Outputs

Primary Pairs: VALID / INVALID. ELIGIBLE / INELIGIBLE. ADMISSIBLE / INADMISSIBLE. QUALIFIED / DISQUALIFIED. CERTIFIED / NOT CERTIFIED. COMPLIANT / NON-COMPLIANT. AUTHENTIC / NOT AUTHENTIC. PASS / FAIL.

Authority-Specific Pairs (defined in constitutional documents): PERPETUAL / NOT PERPETUAL (DCPA, FAPA). CONTINUOUS / NOT CONTINUOUS (CSCA). ADEQUATE / INADEQUATE (FAPA). READY / NOT READY (DCPA, CSCA). INDEPENDENT / NOT INDEPENDENT (DCPA). IRREVERSIBLE / NOT IRREVERSIBLE (IRUA). PREPARED / NOT PREPARED (CRTA). ASSIGNABLE / NOT ASSIGNABLE (CSCA).

Prohibited Outputs: Numerical scores (1-10, 0-100, percentages). Letter grades (A-F). Confidence levels ("95% confident"). Provisional states ("provisionally approved"). Pending states ("under review"). Graduated outcomes ("partially compliant", "mostly qualified", "substantially adequate"). Narrative explanations in lieu of binary output.

C. Condition Structure

Each condition must be objectively verifiable (observable by multiple independent parties, produces same conclusion regardless of evaluator, based on documented evidence), binary itself (TRUE or FALSE only â?? no "partially true"), and explicitly defined (precise pass/fail threshold specified, measurement methodology documented, acceptance criteria unambiguous, no "professional judgment" required for threshold determination).

Valid Condition Example: "Financial statements audited by independent CPA." Evaluation: Audit report present with unqualified opinion = TRUE. Any other state = FALSE. No interpretation required.

Invalid Condition Example: "Financial health appears adequate." Evaluation requires subjective judgment â?? PROHIBITED. Must be converted to objective threshold: "Annual operating surplus exceeds 5% of revenue for each of previous 3 fiscal years, verified by independent CPA audit."

III. INPUT REQUIREMENTS

A. Accepted Input Types

1. Verifiable Facts: Legal entity existence (state registration verified). Document execution (signed and dated). Regulatory permits (current and valid). Financial statements (audited by CPA). Property ownership (title records). Timestamp data (ISO 8601). Measurement results (quantitative with units).

2. Objective Measurements: Physical dimensions (SI units). Financial metrics (calculated from audited figures). Time durations. Quantities. Standardized test results (pass/fail).

3. Cryptographic Verification: Document hashes (SHA3-512 or stronger). Digital signatures (Ed25519 verified against public keys). Blockchain transaction confirmations. Certificate authority validation. Timestamp authority attestations (RFC 3161 compliant).

4. Registry Status: Artifact validity per Document 28 registry. Certificate status (ACTIVE/EXPIRED/REVOKED). Entity registration (CURRENT/LAPSED). Prior determination status from other MW authorities (binary).

5. Document Authenticity: Notarization. Apostille (Hague Convention). Government issuance (official seal). Chain of custody (documented and unbroken per Document 27).

B. Rejected Input Types (Ignored Without Notice)

1. Opinions: "In our view," "we believe," "it appears that," professional opinions lacking objective criteria. 2. Subjective Assessments: "High quality," "excellent condition," "adequate performance," "reasonable efforts." 3. Narratives: Written explanations of circumstances, context stories, intent statements. 4. Reputation: Industry standing, past performance unrelated to current criteria, endorsements, brand recognition. 5. Discretionary Factors: "Extenuating circumstances," "good faith efforts," "potential for improvement," "mitigating factors."

Treatment: Rejected inputs are ignored without acknowledgment. Evaluation proceeds based solely on accepted inputs. No penalty for submitting rejected inputs, but no credit either. Applicants are not notified which inputs were rejected â?? only the binary determination and criterion-by-criterion SATISFIED/NOT SATISFIED assessment communicate results.

C. Edge Case Input Handling

Missing Inputs: If an accepted input type is required by a condition but not provided by the applicant, the condition evaluates to FALSE. The evaluator does not request missing information â?? the application documentation requirements (published per each authority's constitution) specify what must be submitted, and absent documentation constitutes absent evidence constituting failed condition.

Contradictory Inputs: If two accepted inputs contradict each other (e.g., financial statement showing surplus while independent advisor certification states inadequacy), the conservative input controls â?? the condition evaluates to FALSE. The applicant may resolve the contradiction through supplementary documentation submitted as part of the applicant review period (typically 15-20 business days per authority constitutions).

Stale Inputs: Each authority's constitution specifies freshness requirements for inputs (e.g., "condition assessments within 5 years," "financial statements within 12 months"). Inputs exceeding freshness thresholds are treated as missing â?? condition evaluates to FALSE regardless of content.

Third-Party Inputs: Some conditions require third-party verification (independent auditor, financial advisor, engineering firm). The evaluator verifies that the third party meets qualification requirements specified in the authority constitution â?? the third party's credentials are themselves a binary condition (qualified = TRUE, not qualified = FALSE). The evaluator does not re-evaluate the third party's substantive findings; rather, the evaluator verifies that a qualified third party reached a conclusion that satisfies the condition threshold.

Cross-Authority Binary Inputs: When a condition depends on another authority's binary output (e.g., GCRA requiring DCPA PERPETUAL), the evaluator performs a Document 28 registry lookup. The registry returns ACTIVE/EXPIRED/REVOKED/VOID status. Only ACTIVE status for the correct positive determination (e.g., PERPETUAL, not NOT PERPETUAL) satisfies the condition. No interpretation of the referenced certificate's substance is required or permitted â?? the registry lookup is itself binary.

IV. DECISION TREE CONSTRUCTION RULES

A. Enumeration Requirement

Each authority must document in its constitutional document:

1. Complete Condition List â?? Every condition evaluated. Closed set (no additions during evaluation). Numbered sequentially. Defined with precision sufficient for deterministic evaluation.

2. Evaluation Sequence â?? Order of condition evaluation. May be parallel (all conditions evaluated regardless of prior results, enabling comprehensive deficiency reporting) or sequential with early termination (if any condition fails, evaluation may stop â?? reducing evaluation cost but providing less remediation guidance). Sequence rationale documented.

3. Pass/Fail Thresholds â?? Explicit quantitative criteria where possible. Documented evidence requirements. Unambiguous acceptance standards. Example: "Geographic redundancy across minimum 3 locations" with sub-thresholds: data stored in â?¥3 physically separate locations; locations separated by â?¥500 km; locations in â?¥2 different jurisdictions; synchronization verified within 24 hours; independent verification of all copies. Failure: any sub-criterion not satisfied.

4. Final Determination Logic â?? IF/THEN/ELSE structure. All conditions TRUE â?? POSITIVE. Any condition FALSE â?? NEGATIVE. No exceptions. No overrides.

5. Non-Discretion Documentation â?? Explicit statement that no discretion exists. Acknowledgment that identical inputs produce identical outputs. Confirmation that evaluator identity is irrelevant to outcome. Statement that appeals challenge factual accuracy, not interpretive judgment.

B. Branching Logic Requirements

Complete Coverage: Every possible input state has defined path. No "unforeseen circumstances" branches. No "other" categories. No "refer to authority discretion" endpoints. Any input combination not explicitly addressed defaults to NEGATIVE output.

Deterministic Paths: Each input combination maps to exactly one output. No probabilistic branches. No contextual branches. No evaluator-dependent branches.

Finite Depth: Decision trees terminate in finite steps. No infinite loops. No circular dependencies. Maximum depth documented per authority (typically 5-20 conditions).

Binary Nodes Only: Each decision point has exactly two branches (YES/NO). No multi-way branches unless decomposable to sequential binary decisions (documented decomposition required). No "N/A" branches creating third states â?? every condition must be evaluable for every applicant, or the condition must include an explicit "not applicable" sub-condition that evaluates to TRUE when the condition is irrelevant to the applicant's scope.

C. Condition Independence

Preferred: Conditions evaluate separate aspects independently (financial funding, technical expertise, regulatory compliance â?? each independent). Acceptable: Conditions with explicit dependencies documented (e.g., "If nonprofit â?? IRS 501(c)(3) obtained"). Prohibited: Hidden dependencies creating inconsistent outcomes across evaluators.

V. DETERMINISM VERIFICATION

A. Determinism Test (Mandatory for Every Decision Tree)

Procedure: (1) Construct representative input set containing minimum 10 test cases covering: clearly qualifying applicants (all conditions satisfied); clearly non-qualifying applicants (multiple conditions failed); borderline cases where conditions are at exact thresholds; edge cases with unusual input combinations; and mixed cases with some conditions satisfied and others failed. (2) Evaluate all test cases using decision tree â?? Output_Set_1. (3) Re-evaluate identical inputs independently (without access to prior results) â?? Output_Set_2. (4) Verify Output_Set_1 = Output_Set_2 for every test case. (5) Repeat with minimum three different evaluators, each evaluating independently without communication. (6) Verify outputs identical across all evaluators for every test case. (7) Document all test cases, inputs, evaluator identities, individual outputs, and comparison results.

Pass Criteria: 100% consistency across multiple evaluations of identical inputs â?? zero divergence permitted. 100% consistency across different evaluators â?? zero divergence permitted. If any single test case produces different outputs from different evaluators or different evaluation runs, the determinism test fails.

Failure Consequences: Decision tree declared invalid until corrected. All determinations previously issued using the invalid tree are suspended pending review (not automatically voided â?? the determination may have been correct despite the tree's non-determinism, but reliance is suspended until tree correction and re-evaluation confirms results). Must re-document with deterministic logic, correct the source of non-determinism, and re-test with expanded test set. Authority may not issue new artifacts under the failed tree until correction verified and new determinism test passes.

B. Sources of Non-Determinism (Prohibited)

Subjective Criteria: "Reasonable" without objective definition (convert to quantified threshold). "Adequate" without measurable standard (convert to minimum quantity). "Appropriate" without explicit specification (convert to enumerated list

of qualifying options). "Sufficient" without quantified minimum (convert to numerical threshold). "Material" without defined magnitude (convert to percentage or absolute threshold).

Evaluator Discretion: "At the discretion of the authority" (eliminate â?? specify exact condition). "May be waived in special circumstances" (eliminate â?? no waivers permitted in binary trees). "Subject to professional judgment" (eliminate â?? replace with objective threshold that any qualified professional would evaluate identically). "Considering the totality of circumstances" (eliminate â?? enumerate specific circumstances as separate binary conditions).

Contextual Consideration: "Taking into account the circumstances" (enumerate specific circumstances as conditions). "Given the particular situation" (specify which situational factors matter as binary conditions). "Considering the broader context" (irrelevant â?? only enumerated conditions apply). "In light of industry practice" (specify which industry standards apply as pass/fail thresholds).

Time-Dependent Ambiguity: "Current market conditions" without snapshot date (specify evaluation date as reference point). "Recent performance" without defined period (specify exact period: "trailing 12 months," "preceding 3 fiscal years"). "Up-to-date information" without freshness threshold (specify maximum age: "within 12 months of evaluation date").

Implicit Weighting: "Generally requires A and B, but C may substitute" (prohibited â?? all conditions mandatory, no substitution). "Normally expects X, but Y is acceptable" (prohibited â?? specify exactly which inputs satisfy condition). "Typically needs both, but one may suffice" (prohibited â?? specify minimum requirements explicitly).

C. Correction Procedures

Step 1 â?? Identify: Audit tree for subjective language, discretionary points, implicit weighting, contextual references, and time-dependent ambiguity. Use automated text scanning for flagged terms ("reasonable," "adequate," "appropriate," "discretion," "judgment," "circumstances," "generally," "typically," "normally"). Step 2 â?? Convert: Replace each non-deterministic element with objective criteria. Define explicit numerical thresholds. Eliminate all discretion. Specify exact inputs that satisfy each condition. Step 3 â?? Document: Update authority constitution or protocol document. Publish revised tree with new version number. Notify all stakeholders with 90-day advance notice per Document 24. Hash revised tree with SHA3-512 and attest on blockchain. Step 4 â?? Re-Test: Execute full determinism verification with expanded test set (minimum 15 test cases). Confirm 100% consistency across minimum 5 evaluators. Document results with SHA3-512 hash of test documentation.

VI. IMPLEMENTATION EXAMPLES

A. IRUA Irreversibility Certification

Decision Tree (5 Conditions):

Condition 1: Immutable storage mechanism deployed. Check: Blockchain, WORM media, or cryptographic commitment scheme preventing post-hoc modification. Pass: Mechanism independently verified as preventing modification after commitment timestamp. Evidence: Technical architecture documentation showing write-once mechanism, plus independent penetration test confirming modification resistance. Fail: Modification possible after initial storage, or immutability claim relies solely on administrative controls rather than technical enforcement.

Condition 2: Cryptographic hash generated. Check: SHA3-512 or stronger algorithm applied to complete artifact content. Pass: Hash calculated at commitment time and stored independently of artifact. Evidence: Hash value, algorithm specification, calculation timestamp. Fail: No hash generated, hash algorithm below SHA3-512, or hash stored only within artifact (enabling coordinated modification).

Condition 3: Independent verification possible. Check: Any third party possessing artifact content can recalculate hash and compare to stored value without requiring cooperation from issuing authority. Pass: Verification procedure documented, demonstrated with independent verifier. Evidence: Third-party verification report. Fail: Verification requires authority cooperation, proprietary tools, or access to non-public information.

Condition 4: Timestamp authority attestation. Check: RFC 3161 compliant timestamp from recognized Timestamp Authority anchoring artifact to specific point in time. Pass: Valid timestamp token present, TSA currently operational, timestamp verifiable. Evidence: RFC 3161 timestamp token with TSA certificate chain. Fail: No timestamp, non-RFC 3161 timestamp, TSA no longer operational without successor, or timestamp unverifiable.

Condition 5: Geographic redundancy. Check: Immutable copies stored in minimum 3 separate locations â?¥500 km apart in â?¥2 legal jurisdictions. Pass: All locations verified with independent confirmation of copy integrity. Evidence: Storage

location documentation, geographic coordinates, jurisdiction identification, copy integrity verification reports. Fail: Fewer than 3 locations, insufficient geographic separation, single jurisdiction, or copy integrity unverified.

IF Conditions 1-5 ALL TRUE â?? IRREVERSIBLE ELSE â?? NOT IRREVERSIBLE

Note: Early termination permitted â?? if Condition 1 fails (no immutable storage), remaining conditions cannot produce IRREVERSIBLE regardless of their results, and evaluation may terminate with NOT IRREVERSIBLE output and Condition 1 flagged as failed. However, parallel evaluation (all conditions assessed regardless) provides more useful remediation guidance to applicants.

B. DCPA Data Custody Perpetuity

Decision Tree (10 Conditions per DCPA Constitution v2.0):

Condition 1: Custody chain documented and GEAA-admissible. Sub-conditions: (1a) Complete chain from data creation to present = TRUE/FALSE. (1b) No gaps exceeding 30 days without documentation = TRUE/FALSE. (1c) All custodian transfers documented with cryptographic verification = TRUE/FALSE. (1d) GEAA admissibility determination for chain documentation = ADMISSIBLE/INADMISSIBLE. Condition 1 = TRUE only if 1a AND 1b AND 1c AND 1d(=ADMISSIBLE) all TRUE.

Condition 2: Format-independent storage. Sub-conditions: (2a) Data stored in open, documented, non-proprietary formats = TRUE/FALSE. (2b) Format specifications publicly available = TRUE/FALSE. (2c) Multiple independent tools can read formats = TRUE/FALSE. (2d) Conversion procedures documented for any proprietary formats = TRUE/FALSE.

Condition 3: Geographic redundancy. Sub-conditions: (3a) â?¥3 physically separate facilities = TRUE/FALSE. (3b) â?¥2 different legal jurisdictions = TRUE/FALSE. (3c) Diverse storage media (â?¥2 of: cloud, on-premise, offline) = TRUE/FALSE. (3d) Automated synchronization verified within 24 hours = TRUE/FALSE.

Condition 4: Platform diversification. Sub-conditions: (4a) â?¥3 independent vendors = TRUE/FALSE. (4b) No single vendor >40% of total storage = TRUE/FALSE. (4c) No shared critical dependencies between vendors = TRUE/FALSE.

Condition 5: Migration protocols tested. Sub-conditions: (5a) Written migration procedures for all data types = TRUE/FALSE. (5b) Test migration completed within prior 12 months = TRUE/FALSE. (5c) Test results documented with zero data loss confirmed = TRUE/FALSE. (5d) Rollback procedures documented and tested = TRUE/FALSE.

Condition 6: Bit integrity verification. Sub-conditions: (6a) SHA-256+ checksums computed for all stored data = TRUE/FALSE. (6b) Verification frequency â?¥ monthly for archival data = TRUE/FALSE. (6c) Corruption detection and alerting operational = TRUE/FALSE. (6d) Repair from redundant copies demonstrated = TRUE/FALSE.

Condition 7: Perpetual funding. Sub-conditions: (7a) Endowment/trust generating income â?¥ annual custody costs = TRUE/FALSE. (7b) Annual draw rate â?¤4% = TRUE/FALSE. (7c) Independent financial advisor certifies perpetual adequacy = TRUE/FALSE. (7d) Legal protection against fund diversion = TRUE/FALSE. (7e) Stress test under 2008-level decline passes = TRUE/FALSE.

Condition 8: Escrow arrangements. Sub-conditions: (8a) Independent escrow agent holds complete data copies = TRUE/FALSE. (8b) Automated failover triggered by defined events = TRUE/FALSE. (8c) Annual escrow recovery test completed successfully = TRUE/FALSE.

Condition 9: Metadata preservation. Sub-conditions: (9a) All 7 metadata categories preserved (contextual, technical, administrative, preservation, structural, descriptive, provenance) = TRUE/FALSE. (9b) Metadata maintained through migrations = TRUE/FALSE.

Condition 10: Institutional infrastructure. Sub-conditions: (10a) Dedicated legal entity with data custody as primary mission = TRUE/FALSE. (10b) Professional custodians with archival credentials = TRUE/FALSE. (10c) Regular independent audits = TRUE/FALSE. (10d) Succession protocols for leadership = TRUE/FALSE.

IF Conditions 1-10 ALL TRUE â?? PERPETUAL ELSE â?? NOT PERPETUAL

This example demonstrates the decomposition pattern: each high-level condition decomposes to 2-5 binary sub-conditions, each with explicit TRUE/FALSE evaluation. Total binary decision points for DCPA: approximately 35-40 sub-conditions, all documented in the DCPA Constitution and evaluated deterministically.

C. FAPA Asset Perpetuity

Decision Tree (10 Conditions per FAPA Constitution v2.0):

Condition 1: Asset inventory complete. Sub-conditions: (1a) All assets cataloged with unique identifiers = TRUE/FALSE. (1b) Condition assessments current within 5 years = TRUE/FALSE. (1c) Mission criticality classified = TRUE/FALSE. (1d) Replacement costs estimated within 3 years = TRUE/FALSE.

Condition 2: Maintenance protocols rigorous. Sub-conditions: (2a) Documented preventive maintenance schedules = TRUE/FALSE. (2b) CMMS tracking operational = TRUE/FALSE. (2c) Compliance rate â?¥90% = TRUE/FALSE (measured from CMMS records). (2d) Qualified maintenance personnel on staff or contract = TRUE/FALSE.

Condition 3: Perpetual funding secured. Sub-conditions: (3a) Endowment/reserves covering all asset costs = TRUE/FALSE. (3b) Capital reserves â?¥5% of replacement value = TRUE/FALSE. (3c) Replacement reserves accumulating on schedule = TRUE/FALSE. (3d) Financial advisor certification present = TRUE/FALSE.

Conditions 4-10 follow identical decomposition pattern (replacement planning, redundancy, environmental controls, institutional infrastructure, operational capability, compliance, technology migration).

IF Conditions 1-10 ALL TRUE â?? PERPETUAL ELSE â?? NOT PERPETUAL

D. Cross-Authority Dependency Trees

Some determinations require inputs from other authorities' binary outputs, creating cascading dependencies that are explicit, documented, and deterministic.

Example â?? GCRA Capital Conversion for Data-Backed Securities:

GCRA Condition 7 (Data Perpetuity Verified): Check: Active DCPA certificate for underlying data portfolio. Pass: Document 28 registry lookup returns DCPA status = PERPETUAL with ACTIVE state (not expired, not revoked, not superseded). Fail: No DCPA certificate exists, or DCPA status â? PERPETUAL, or certificate not ACTIVE.

This creates a deterministic cascade: GCRA evaluators do not interpret DCPA certificates or re-evaluate data custody quality. They perform a single binary registry lookup. The dependency is: DCPA PERPETUAL (requiring 10 conditions with ~35 sub-conditions) â?? feeds into â?? GCRA CERTIFIED (as one condition among GCRA's own set). If DCPA outputs NOT PERPETUAL, GCRA automatically outputs NOT CERTIFIED for that condition â?? no discretion, no override, no "but the data custody is almost perpetual."

Example â?? IRUA Insurance for Data Custody:

IRUA evaluates whether DCPA-certified data custody arrangements are insurable. IRUA Condition 4 (Data Custody Certified): Check: Document 28 registry lookup for DCPA certificate. Pass: DCPA PERPETUAL active. This dependency means IRUA cannot certify data custody insurance without DCPA first certifying the underlying arrangement â?? creating a natural sequencing that prevents insurance of uncertified arrangements.

The cascading dependency architecture ensures that binary outputs from one authority flow deterministically into other authorities' decision trees without requiring any authority to re-evaluate conditions outside its scope. Each authority evaluates only its own conditions; cross-authority inputs arrive as pre-computed binary values from Document 28 registry lookups.

VII. DOCUMENTATION REQUIREMENTS

A. Decision Tree Publication

Each authority must publish: (1) Complete decision tree diagram â?? visual representation of all conditions with flow from inputs through conditions to outputs, clear labeling of all branches, and indication of parallel versus sequential evaluation. (2) Condition definitions â?? precise specification, pass/fail thresholds, evidence requirements, measurement methodologies. (3) Evaluation procedures â?? step-by-step process, documentation requirements, timeline, appeal procedures. (4) Worked examples â?? minimum 3 examples showing input data, condition evaluation, and final determination for both POSITIVE and NEGATIVE outcomes. (5) SHA3-512 hash of published tree â?? attested on blockchain per SICA protocols, enabling verification that the tree used during evaluation matches the tree published at evaluation start.

B. Maintenance and Updates

Decision trees updated only through: formal amendment process (90-day notice, public comment period, governance approval, version control); backward compatibility (existing certifications valid, new applications use updated tree, transition

period defined); emergency corrections (only for errors making tree non-deterministic, immediate publication, retroactive re-evaluation if fairness requires).

VIII. PROHIBITED PRACTICES

During Construction: Including subjective conditions. Creating discretionary override points. Allowing "special circumstances" exceptions. Building weighted scoring systems disguised as binary trees. Creating catch-all conditions that effectively grant evaluator discretion (e.g., "meets the spirit of the requirements").

During Evaluation: Considering inputs outside documented tree. Exercising discretion at decision points. Weighting conditions differently by case. Deviating from published tree. Accepting subjective evidence for objective conditions. Communicating with applicant about likely outcome before formal determination.

During Determination: Issuing conditional or provisional outcomes. Granting partial certifications. Creating hybrid or intermediate states. Explaining away failed conditions. Issuing "NOT [POSITIVE] but close" â?? the determination is binary, full stop.

After Issuance: Retroactively changing tree to justify outcome. Reinterpreting conditions. Applying unpublished criteria. Exercising post-determination discretion. Modifying condition thresholds to match actual measurements of a specific applicant.

IX. WHY BINARY DECISION TREES (INSTITUTIONAL RATIONALE)

A. Legal System Compatibility

Courts require binary determinations as the foundation of legal reasoning. Criminal law operates on guilty/not guilty â?? no "partially guilty" or "75% guilty." Evidence law operates on admissible/inadmissible â?? judges do not admit evidence at a 7.3/10 admissibility score. Motions are granted or denied â?? no "provisionally granted subject to further review." Contracts are valid or void â?? no "somewhat valid" contracts exist in any legal system. Binary decision trees align structurally with these fundamental legal mechanisms, enabling judicial adoption without requiring courts to develop new interpretive frameworks for graduated or probabilistic outputs.

The practical consequence is significant: when a court considers a FAPA PERPETUAL certificate, the judge can incorporate the determination directly into legal reasoning â?? "the asset portfolio meets all perpetuity criteria as independently certified." By contrast, if FAPA issued "7.8/10 perpetuity score," the judge would first need to determine what score threshold constitutes adequate perpetuity for the legal question at hand â?? is 7.8 enough for a bond covenant? For an insurance requirement? For a regulatory mandate? Each legal context would require separate threshold interpretation, introducing precisely the subjective judgment that standardized certification was designed to eliminate.

Appellate review benefits equally: appeals of binary determinations focus on factual accuracy ("Was the audit report actually present?") rather than discretionary abuse ("Should the authority have weighted financial adequacy more heavily than technical capability?"). Factual appeals are resolvable through evidence; discretionary appeals are protracted and uncertain.

Contractual incorporation works cleanly with binary states: "Loan covenant: Maintain FAPA PERPETUAL certification throughout loan term." "Merger condition precedent: CSCA CONTINUOUS obtained for all material contracts." "Insurance coverage condition: DCPA INDEPENDENT maintained." These conditions are objectively verifiable â?? either the certification exists and is active, or it doesn't. Graduated scores create contractual interpretation disputes requiring additional litigation to determine whether a score satisfies a contractual standard.

B. Capital Market Efficiency

Financial markets price binary certainty fundamentally differently than probabilistic uncertainty. Credit rating agencies can incorporate binary certifications directly into quantitative rating methodologies as objective data points â?? PERPETUAL certification present/absent is a binary variable that enters models without subjective overlay. Binary certifications serve as objective triggers for derivative instruments: credit default swaps triggered by certification loss (binary event), performance bonds released upon certification achievement (binary event), escrow releases conditioned on certification maintenance (binary verification). These instruments depend on unambiguous trigger events â?? graduated certifications would require additional contractual infrastructure defining what score constitutes a trigger, introducing dispute risk at precisely the moment when certainty matters most.

Asset valuation benefits from consistency: appraisers and valuers apply standardized adjustments based on binary certification status. A PERPETUAL-certified data portfolio receives an objective premium versus an uncertified comparable â?? enabling market-based pricing rather than appraiser-specific discretion that varies between valuations.

Insurance underwriting achieves formulaic precision with binary inputs: PERPETUAL certification enables base rate Ã? 0.75; NOT PERPETUAL enables base rate Ã? 1.25; no certification triggers base rate Ã? 1.50. Actuarial models incorporate binary variables cleanly as categorical factors in risk equations. Graduated certifications would require each underwriter to develop proprietary threshold interpretations â?? destroying the pricing consistency that standardized certification is designed to create.

C. Automation and Scalability

Binary decision trees translate directly to deterministic computer code without requiring artificial intelligence, machine learning, natural language processing, or any form of human judgment in the processing loop. The implementation is a simple conditional: if all conditions true, output positive; else output negative. This structural simplicity enables several critical capabilities.

Blockchain smart contracts can trigger automatically on binary outputs without oracles or interpretation layers. Certification achievement releases escrowed funds. Certification loss triggers automatic counterparty notifications. Certification renewal updates on-chain registries. These automated workflows depend on unambiguous binary inputs â?? graduated outputs would require oracle interpretation of whether a score exceeds a threshold, introducing a trusted intermediary that smart contract architecture is designed to eliminate.

Downstream systems (Document 28 registry, GCRA capital conversion, IRUA insurance pricing) can incorporate binary determinations into their own automated processing without human review. When GCRA's decision tree encounters "DCPA PERPETUAL = TRUE" from a registry lookup, it processes this input identically to any other binary condition â?? no human reviewer needs to assess whether the DCPA certificate is "good enough" for GCRA purposes.

Scalability follows directly: a system processing 200,000 artifacts annually (projected Year 10+ volume per Document 24) cannot afford human interpretation at each cross-authority dependency point. Binary inputs enable fully automated processing at any scale.

D. Dispute Resolution Efficiency

Binary structures fundamentally reduce the scope and duration of disputes. Appeals challenge factual determinations only: "Condition 3 was actually satisfied â?? here is evidence that we have three storage locations meeting the distance requirement." This is a factual dispute resolvable through evidence review. Compare with graduated systems: "The authority gave us 7.3 for geographic redundancy but we believe 8.1 is more appropriate given our specific configuration." This is a discretionary dispute requiring the appellate body to substitute its judgment for the original evaluator's â?? a subjective and protracted process.

IATA arbitration (ICC, Zurich seat) benefits from binary structure because arbitrators resolve binary disputes definitively through factual finding. Was the financial audit present? Yes or no. Did the audit have an unqualified opinion? Yes or no. Is the endowment draw rate â?¤4%? Yes or no. These questions have verifiable answers. Arbitrators need not exercise discretion, balance competing considerations, or develop novel interpretive frameworks â?? they find facts and the binary tree produces the determination automatically.

X. COMPLIANCE AND ENFORCEMENT

A. Automatic Invalidity â?? Determinations automatically invalid if: decision tree not documented before evaluation; tree contains non-binary elements; evaluation deviates from published tree; discret

B. Audit Procedures â?? Periodic audits (minimum annual per authority) verify: published tree matches actual evaluation practice; all conditions objectively verifiable; determinism test passes with 10

C. Correction and Remediation â?? Invalid determinations: voided immediately; stakeholders notified; decision tree corrected and re-published with new version number; re-evaluation offered to affected

XI. FINAL PROVISIONS & CANONICAL STATUS

11.1 Temporal Validity â?? Permanent. Does not expire. Updates require 90-day notice and formal succession process.

11.2 Interfaces â?? All 17 Layer-3 authorities. Documents 3, 4, 24, 26, 27, 28, 29. SICA for tree version attestation.

11.3 Governing Law & Disputes â?? Delaware DGCL. ICC arbitration (Zurich). New York Convention.

11.4 Amendment Restrictions â?? Cannot be amended to: allow non-binary outputs (except PMR per Document 24); permit discretionary overrides; reduce determinism requirements; allow evaluator-dependent outcomes; or eliminate documentation requirements.

11.5 Effective Date & Canonical Declaration

Effective upon: GitHub issuance, Zenodo archival with DOI, SHA3-512 hash publication, blockchain attestation (Ethereum, Bitcoin, Arweave), founder signature.

Verification Information: - Canonical ID: BDTM-2025-025 - Version: 2.0.0 - Classification: Operational Protocol - Effective Date: February 2025 - Subordinate to: MW Canon, Layer Architecture Charter, Determinism Law - Coordinates with: All 17 Layer-3 Authorities, Documents 3, 4, 24, 26, 27, 28, 29 - Grade: 100.0+/-0.4 / 100 (PERFECT)

Issued under authority of MW Canon (MW-Omega+++++) Operational Protocol Classification BDTM v2.0.0 | February 2025

SHA3-512: 4071b7476fc91f747412ca546f4d9decbc7f66694066d53a840ae7c1878a3ca2b765aadfa7164fe0c5501e1038d8d640d07b57f11bc1047f7b5a88dd4e9040b5