RM-S-PAY-001 Digital Payment Verification Standard

Standard ID: RM-S-PAY-001 **Version:** v1.0-F **Status:** Frozen **Publication Date:** 2026-07-23 **Effective Date:** 2026-07-23
Standard Family: RM-S — Canonical Standards Namespace **Normative Status:** Normative **Publisher:** RuleMark **Document Language:** English **Machine ID:** `rulemark:standard:rm-s-pay-001:v1.0-F`

Document Control

Field	Content
Standard ID	RM-S-PAY-001
Standard Name	Digital Payment Verification Standard
Version	v1.0-F
Status	Frozen
Publication Date	2026-07-23
Effective Date	2026-07-23
Standard Family	RM-S
Domain	Payment instruction, authorization, execution, settlement, finality, refund, dispute, evidence, machine verification
Normative Status	Normative
Target Subject	Digital payment records, batches, processes, agent payments, and verification implementations
Default Determination	PASS / CONDITIONAL / DEFICIENT / FAIL / NOT ASSESSED
Conformance Use	Authorized

Abstract

This standard establishes a unified, deterministic, and machine-executable set of rules for verifying that a digital payment record is authentic, complete, internally consistent, and re-computable.

This standard requires verification of payment identity, payer and payee identification, payment instruction content, authorization, idempotency and duplicate prevention, submission and execution, status transitions, settlement evidence, finality classification, amount conservation, fees and exchange rates, failures, refunds, reversals, chargebacks, disputes, record retention, evidence integrity, human–machine consistency, and the signed verification result.

This standard does not determine the lawfulness of the source of funds, does not certify KYC, AML, sanctions, or tax compliance, does not adjudicate dispute liability, and does not determine whether a payment is commercially advisable.

This standard answers only:

> At a specified verification time, under a specified standard version, for a specified conformity subject and a specified evidence set, does the digital payment satisfy the applicable requirements of RM-S-PAY-001?

1. Applicability

1.1 Conditions of Applicability

This standard applies to a digital payment that meets all of the following conditions:

1 an identifiable obligated party, payer, and payee exist, or a verifiable protected alias exists;

2 an explicit payment instruction exists, containing the amount, the currency or digital asset unit, the payer, the payee, and the creation time;

3 the payment is initiated, processed, confirmed, or settled through an electronic system;

4 at least one verifiable transaction identifier, ledger identifier, payment network identifier, or processor reference exists;

5 the payment status can be represented by a structured record that distinguishes at least `INITIATED`, `AUTHORIZED`, `SUBMITTED`, `SETTLED`, `FAILED`, and `REVERSED`;

6 payment-related evidence is lawfully or contractually accessible to the verifier;

7 the payment system can provide a determinate verification time and evidence generation time;

8 where the obligated party claims conformance, it can provide every mandatory item of evidence required by this standard;

9 the payment is a transfer of fiat currency, fiat-denominated electronic money, reserve-backed stablecoin, or other expressly denominated digital value;

10 the verification objective is to determine whether the payment record and its status are authentic, complete, consistent, and re-computable, and not to determine investment value.

1.2 Out of Scope

This standard does not apply to:

- cash payments, or payments completed only on paper with no electronic transaction record;
- quotations, valuations, invoices, or payment requests for which no payment instruction has yet been formed;
- records without a unique transaction identifier and for which no equivalent unique identifier can be established;
- purely internal accounting entries that do not represent a transfer of value between a payer and a payee;
- exchanges of value that are undenominated, unmeasurable, or whose amount unit cannot be determined;
- mining rewards, staking rewards, airdrops, or gifts, unless they are processed as payment transactions and satisfy all applicable conditions of this standard;
- issuance, valuation, suitability, or market-conduct review of securities transactions;
- product certification of payment instruments themselves, such as card chips, hardware wallets, or cryptographic modules;
- certification of blockchain consensus protocols, network throughput, or degree of node decentralization;
- complete KYC, AML, sanctions, or tax compliance certification of the payer or payee;
- final legal adjudication of dispute liability by a court, arbitral body, or payment network;
- monetary policy, issuance mechanism, or legal tender status of a central bank digital currency;
- certification of a payment network, bank, stablecoin, or wallet as a whole institution. What this standard certifies is the conformance of payment records, payment processes, or payment services within a declared scope.

1.3 Non-Goals

This standard:

- does not provide investment advice;
- does not certify legal compliance in any jurisdiction;
- does not replace a regulator, auditor, court, arbitral body, or legal counsel;
- does not constitute a credit rating;
- does not constitute a RuleMark guarantee, recommendation, or endorsement;
- does not determine whether the source of funds is lawful;
- does not determine whether the payer or payee ought to have entered into the transaction;
- does not prescribe the technical implementation of any payment network, bank, card scheme, blockchain, or wallet;

- does not determine whether payment rates, commercial pricing, or exchange-rate levels are fair;
- does not guarantee that a settled payment can never be revoked through legal, operational, or protocol mechanisms;
- does not replace PCI DSS, ISO 20022, EMV, SWIFT, card scheme rules, or any other external technical or operational specification.

1.4 Conforming Subject

A conformance claim MUST correspond to exactly one of the following conformity subject types:

1. a single digital payment record;
2. a batch of digital payments sharing common rules and evidence structure;
3. a payment service process that handles digital payments;
4. a machine or AI agent payment record executed on behalf of a payer;
5. a payment verification implementation declared by a payment service provider for a specified product, network, and version.

Unique identification MUST include at least:

- `conformity_subject_type`;
- `conformity_subject_id`;
- `obligated_party_id`;
- `payment_id` or `batch_id`;
- `payment_network`;
- `environment` (`PRODUCTION` or `SANDBOX`);
- `implementation_name`;
- `implementation_version`;
- `verification_time`;
- `standard_version`.

A conformance claim MUST NOT be made for an institution as a whole. The declared scope MUST identify the payment network, environment, implementation, and version to which the claim applies.

1.5 Participant Roles

This standard distinguishes the following participant roles. A single legal entity MAY assume multiple roles, but the responsibilities of each role MUST be separately identifiable.

- **Obligated Party**: the payment service provider, bank, wallet service, payment processor, merchant acquirer, stablecoin payment service, machine payment operator, or other party responsible for submitting the payment record. It maintains payment records, provides authentic, complete, and consistent evidence, declares the applicable scope, handles material status changes, and retains the records required for verification. It is the subject of the conformance claim.
- **Verifier**: an independent audit firm, compliance team, merchant, counterparty, regulatory observer, or authorized third party. It obtains evidence in accordance with the verification procedure, re-computes results, records conflicts, and signs the verification record. It MUST NOT extend the scope of its conclusion beyond what was verified.
- **Machine Validator**: software, a smart contract, an API service, or an AI tool that executes the schema and verification rules of this standard. It validates fields, enumerations, times, amounts, status transitions, evidence digests, signatures, and the overall result, and produces a deterministic outcome.
- **Evidence Provider**: a bank, payment network, blockchain node, card scheme, wallet, acquirer, clearing system, stablecoin issuer, dispute-handling system, or trusted timestamping service. It provides original or independent verifiable evidence and states its generation time, source, scope, and integrity protection method.

Where a commercial, fee, ownership, or control relationship exists between the Verifier and the Obligated Party, it MUST be disclosed under §19.3. An undisclosed material interest relationship renders the verification result invalid.

2. Normative Terminology

The normative keywords in this standard are used with the following meanings:

- **MUST / SHALL**: mandatory requirement;
- **MUST NOT / SHALL NOT**: express prohibition;
- **SHOULD**: recommended requirement; a documented reason MUST be recorded where not satisfied;
- **SHOULD NOT**: not recommended; a documented reason MUST be recorded where adopted;
- **MAY**: optional capability.

Only the above English keywords appearing in uppercase have normative effect. Informative explanations, examples, and clarifications shall not override normative requirements.

3. Definitions

Term	Definition
Payment	The electronic transfer of a determinate amount and denomination unit from a payer to a payee pursuant to a payment instruction.
Payment Record	The structured record describing a single payment from creation to terminal state, together with its associated evidence.
Payment Instruction	An instruction that determines the payer, payee, amount, denomination unit, and payment purpose before execution.
Payment Intent	A structured expression by the payer or an authorized agent of the intention to make a specific payment. An intent alone is neither an authorization nor a settlement.
Payer	The party that bears the payment amount and initiates or authorizes the payment.
Payee	The party designated to receive the payment amount.
Obligated Party	The party responsible for the conformance claim, record integrity, and provision of evidence.
Payment Service Provider	An organization or system that receives, routes, processes, executes, clears, settles, or confirms a digital payment.
Authorization	A verifiable approval of a specific payment instruction by the payer or a lawfully authorized agent.
Independent Authorization	An authorization that is bound to a distinct payment instruction digest and is issued by a separate authorization event, rather than being derived from, reused from, or inferred from a prior authorization.
Authentication	A process used to confirm the identity of a subject, device, key, or agent. Authentication is not payment authorization.
Submission	The state in which a payment instruction has been sent to an executing network, ledger, bank, or processor and a proof of receipt has been obtained.
Execution	The act by which a payment system performs the debit, transfer, posting, broadcast, or equivalent processing of an authorized payment instruction.
Clearing	The transmission, matching, confirmation, netting, or preparation of payment obligations prior to settlement.
Settlement	The discharge of the payment obligation between payer and payee through final or contractually agreed ledger posting, funds transfer, or digital asset transfer.
Settlement Evidence	Verifiable evidence generated by a settlement system, bank, payment network, blockchain, or equivalent authoritative source.

Term	Definition
Finality	The state in which, under the declared rules and at the verification time, a payment is no longer subject to unilateral cancellation through ordinary operational process.
Operational Finality	Completion under the ordinary operating rules of the payment system, after which no automatic reversal occurs except through refund, dispute, fraud recovery, court order, or protocol-level exception.
Legal Finality	A legal determination of the irrevocability of a transfer under applicable law. This standard does not certify legal finality unless an independent legal opinion is supplied as supplementary evidence.
Confirmation	A record by which a payment system indicates to a participant or verifier that a payment has reached a specified state.
Transaction Identifier	A string that uniquely identifies a single payment within the declared payment system and environment.
Network Reference	An external reference assigned by a payment network, bank, blockchain, processor, or clearing system.
Idempotency Key	A unique key used to ensure that repeated submission of the same payment instruction does not produce an unauthorized duplicate payment.
Duplicate Payment	A second or subsequent transfer of value arising from the same payment intent or the same idempotency key in the absence of an independent authorization.
Fee	An amount charged for processing a payment, recorded separately from the payment principal.
Exchange Rate	The numeric ratio used where a payment involves conversion of currency or unit, together with its source and time.
Gross Amount	The payment principal paid by the payer, exclusive of fees or inclusive of fees only where the record states so expressly.
Net Amount	The amount actually received by the payee.
Refund	A reverse transfer of value initiated by the payee or the payment service provider after the original payment.
Refundable Principal	The gross amount of the original payment, exclusive of fees. Fees borne by either party are not refundable principal and MUST be recorded separately under §11.3. Where the payment involved conversion, the refundable principal is expressed in the source currency of the original payment instruction.
Reversal	The cancellation, correction, or roll-back of an original payment before final settlement or pursuant to system rules.
Chargeback	A forced recovery of funds initiated by the payer's institution under payment network or legal dispute process.
Dispute	A formal record of an objection concerning payment authorization, amount, delivery, fraud, or liability.
Material Event	An event that changes the payment status, the reliability of evidence, the settlement outcome, or the conformance conclusion.
Evidence	A verifiable record, document, log, ledger datum, signature, or external confirmation that supports or refutes a requirement.
Evidence Digest	A 64-character lowercase hexadecimal SHA-256 digest computed over canonicalized evidence content.
Verification Time	The UTC time on which the verifier bases the conformance determination.
Evidence Age	The verification time minus the evidence generation time.
Human Disclosure	Payment information and conformance statements presented for human reading.
Machine Record	The structured JSON verification record produced under the schema of this standard.
PASS	All applicable MUST requirements are satisfied, no MUST NOT is violated, all mandatory evidence is valid, and the verification record is signed.
CONDITIONAL	All MUST and MUST NOT requirements are satisfied and one or more SHOULD requirements are unsatisfied.

Term	Definition
DEFICIENT	No requirement is violated on the available evidence, but one or more mandatory items cannot be verified because evidence is missing, incomplete, or expired, and the gap is curable.
FAIL	At least one applicable MUST is unsatisfied, one MUST NOT is violated, mandatory evidence is missing or invalid, or the result is not re-computable.
NOT_ASSESSED	Verification is not complete and no conformance conclusion is constituted.

3.1 Restriction on Ambiguous Terms

The following words MUST NOT be used alone in normative clauses: reasonable, appropriate, sufficient, adequate, timely, regular, high-quality, secure, transparent, real-time, independent, material, low-risk.

This standard additionally prohibits the standalone use of: instant, immediate, final, irreversible, confirmed, completed, negligible, acceptable delay, normal processing time, trusted provider, industry standard security, commercially reasonable.

Where any such word is used, the clause MUST also define: a measurable value; a time range; test conditions; the responsible party; and the determination method. The terms defined in §3 satisfy this requirement for the meanings given there.

4. Core Principles

4.1 Evidence First

Any conformance conclusion MUST be supported by verifiable evidence. A statement by the obligated party alone MUST NOT constitute independent evidence.

4.2 Identifiable Responsibility

Every function, claim, and item of evidence MUST be traceable to a responsible party.

4.3 Same-Point Reconciliation

Payment status, evidence validity, and the overall determination MUST use the same verification time.

4.4 Non-Discretionary Determination

The same inputs, the same standard version, and the same verification time MUST produce the same result.

4.5 Machine Readability

Core payment data and verification results MUST have a structured representation.

4.6 Minimum Necessary Claim

A conformance conclusion MUST NOT exceed the verified subject, network, version, and time range.

4.7 Freshness

Evidence exceeding the maximum age specified in §14.4 MUST NOT be used to prove current state.

4.8 Traceability

Every mandatory requirement MUST be traceable to evidence, a verification step, and a failure code.

4.9 Separation of Status and Funds

A status displayed by a payment system MUST NOT substitute for independent settlement evidence.

4.10 Separation of Authorization and Authentication

An authenticated subject MUST NOT be treated as having authorized a specific payment.

4.11 One Payment, One Canonical Identity

Every payment MUST have a unique payment identifier within the declared environment, and the same idempotency key MUST NOT produce an unauthorized duplicate payment.

4.12 Amount Conservation

The payment principal, fees, exchange rate, net amount, and refunds MUST be re-computable through a determinate formula.

4.13 Terminal States Cannot Be Simulated

States such as `SETTLED`, `FAILED`, and `REVERSED` MUST have corresponding evidence and MUST NOT be determined from interface display alone.

4.14 Layered Finality

Operational finality and legal finality MUST be distinguished. Where legal finality has not been verified, legal irrevocability MUST NOT be asserted.

4.15 Full Lifecycle Record

Authorization, submission, execution, settlement, refund, reversal, and dispute states MUST retain a traceable history.

4.16 Failure Closure

Where mandatory evidence is missing, evidence conflicts, or the result is not re-computable, the result MUST be FAIL. A default pass MUST NOT be applied.

5. Canonical Identity and Responsible Parties

5.1 Conformity Subject Canonical Identity

Every conformity subject submitted for verification MUST have a canonical identity containing at least:

```
{
  "conformity_subject_type": "",
  "conformity_subject_id": "",
  "obligated_party_id": "",
  "payment_id": "",
  "batch_id": "",
  "payment_network": "",
```

```
    "environment": "",
    "implementation_name": "",
    "implementation_version": "",
    "verification_time": "",
    "standard_version": ""
  }
```

Exactly one of `payment_id` or `batch_id` MUST be present, according to the declared `conformity_subject_type`.

5.2 Obligated Party Registration

The obligated party MUST disclose its unique identifier, legal name, jurisdiction of registration, a responsibility statement locator, and an identity evidence reference.

5.3 Payer, Payee, and Agent Identification

The payment record MUST identify the obligated party, the payer, and the payee. Where a protected alias is used, the record MUST permit an authorized verifier to map the alias to the underlying subject.

5.4 Change of Responsibility

Where the obligated party for a conformity subject changes, the record MUST capture the party before and after the change, the effective time, and the evidence supporting the change, and the affected verification result MUST be re-computed.

6. Payment Identity and Instruction

6.1 Conformity Subject Identifier

Every conformity subject MUST have a unique, stable, and non-duplicated `conformity_subject_id`. Where the identifier is incomplete or cannot be shown to be unique, the result MUST be FAIL using `RMFPAY-ID-001`.

6.2 Payment Identifier

Every payment MUST have a `payment_id` that is unique within the declared payment network and environment. Where the payment identifier is duplicated or uniqueness cannot be demonstrated, the result MUST be FAIL using `RMFPAY-ID-003`.

6.3 Party Identification

The payment record MUST identify the obligated party, the payer, and the payee. Where a protected alias is used, the record MUST be capable of being mapped to the underlying subject by an authorized verifier. Where any of these parties or an agent cannot be confirmed, the result MUST be FAIL using `RMFPAY-ID-002`.

6.4 Payment Instruction Content

The payment instruction MUST record the principal amount, the denomination unit, the payer, the payee, the creation time, and the payment purpose or purpose code. Where a field is missing or incomplete, the result MUST be FAIL using `RMFPAY-INS-001`.

6.5 Amount Representation

Amounts MUST be represented as decimal strings with the number of decimals recorded. Binary floating-point values MUST NOT be used as the canonical amount. Where the format, unit, or decimals do not conform, the result MUST be FAIL using `RMFPAY-AMT-001`.

7. Authorization

7.1 Existence and Ordering of Authorization

Every payment MUST have authorization evidence bound to that payment instruction, and the authorization time MUST NOT be later than the first execution time. Where authorization evidence is missing or the authorization time is later than execution, the result MUST be FAIL using `RMFPAY-AUT-001`.

7.2 Authorization Content

Authorization evidence MUST contain the authorizing subject, the authorization method, the authorization time, the authorization scope, and the canonical digest of the authorized payment instruction. Where the evidence is not bound to a specific payment instruction or the authorization scope is incomplete, the result MUST be FAIL using `RMFPAY-AUT-002`.

7.3 Authentication Is Not Authorization

The obligated party MUST NOT treat successful authentication alone as successful payment authorization. Where authentication is treated as authorization, the result MUST be FAIL using `RMFPAY-AUT-003`.

7.4 Machine and AI Agent Authorization

For a payment initiated by a machine or AI agent, the record MUST identify the agent, the delegating principal, the authorization scope, the maximum single-payment amount, the permitted payees, and the authorization expiry time. Where these fields are incomplete, the result MUST be FAIL using `RMFPAY-AGT-001`.

7.5 Agent Limits

A machine or AI agent MUST NOT generate a conforming payment after its authorization has expired, above its maximum single-payment amount, or outside its permitted payee range. Where such a payment is generated, the result MUST be FAIL using `RMFPAY-AGT-002`.

8. Idempotency and Duplicate Prevention

8.1 Idempotency Key

A payment system MUST assign an idempotency key to any payment instruction that a client may retry. Within the same obligated party and the same environment, an idempotency key MUST NOT be reused for a different payment intent for at least 35 days. Where the key is missing, reused, or the retention period is insufficient, the result MUST be FAIL using `RMFPAY-DUP-001`.

8.2 Duplicate Payments

The same payment intent or the same idempotency key MUST NOT produce two or more successful transfers of value in the absence of an independent authorization as defined in §3. Where such a duplicate is found, the result MUST be FAIL using `RMFPAY-DUP-002`.

9. Submission, Execution, and Status

9.1 Proof of Receipt

After submission, the obligated party MUST retain the proof of receipt or external reference returned by the payment network, processor, bank, or ledger. Where this is missing, the result MUST be FAIL using `RMFPAY-EXE-001`.

9.2 Status Transitions

Payment status transitions MUST conform to the permitted transition table in §18.4, and each transition MUST record the UTC time, the actor, and a reason code. Where the status history is missing, times are not monotonic, or a transition is not permitted, the result MUST be FAIL using `RMFPAY-STA-001`.

9.3 Prohibited Terminal Transitions

A payment MUST NOT transition directly from `FAILED`, `REVERSED`, or `REFUNDED_FULL` to `SETTLED`. Where a payment is to be attempted again, a new `payment_id` and a new authorization MUST be created. Where a prohibited transition occurs, the result MUST be FAIL using `RMFPAY-STA-002`.

9.4 Status Query Interface

The payment service SHOULD provide a verifiable payment status query interface that allows an authorized verifier to obtain the current status and evidence digests by `payment_id`.

9.5 Terminal Record Generation

The payment service SHOULD generate a machine-readable terminal-state record within 60 seconds after the payment reaches a terminal state.

9.6 Unknown External State

Where the state is unknown because of an external network interruption, the payment service SHOULD use `PENDING_EXTERNAL_CONFIRMATION` and SHOULD NOT mark the payment directly as `FAILED` or `SETTLED`.

10. Settlement and Finality

10.1 Settlement Evidence

Where a payment is declared `SETTLED`, settlement evidence MUST exist from a settlement system, bank, payment network, blockchain, or equivalent authoritative source. Where such evidence is missing or invalid, the result MUST be FAIL using `RMFPAY-SET-001`.

10.2 Settlement Evidence Content

Settlement evidence MUST contain the external reference, the settlement time, the settled amount, the denomination unit, the settlement status, and the evidence source. Where any of these is absent, the result MUST be FAIL using `RMFPAY-SET-002`.

The settled amount MUST correspond to `net_amount` as computed under §11.1, expressed in the settlement currency, unless the record expressly declares a different basis and states the reconciling items. Where the settled amount cannot be reconciled to `net_amount` to zero minor units, the result MUST be FAIL using `RMFPAY-AMT-002`.

10.3 Blockchain Finality

For payments requiring block confirmation, the obligated party MUST declare the target network, the block height or ledger position, the transaction hash, and the confirmation rule required to reach operational finality. Where these parameters or proofs are missing, the result MUST be FAIL using `RMFPAY-FIN-001`.

10.4 Non-Blockchain Finality

For non-blockchain payments, the obligated party MUST declare the operational final state defined by the payment network and the corresponding evidence. Where the rule or evidence is missing, the result MUST be FAIL using `RMFPAY-FIN-002`.

10.5 Legal Finality

The obligated party MUST NOT describe operational finality as legal irrevocability or legal finality unless an independent legal opinion has been obtained and supplied as evidence. Where legal irrevocability is asserted without such verification, the result MUST be FAIL using `RMFPAY-FIN-003`.

> Note: the legal wording of this clause MUST be confirmed by a payments legal expert before formal freezing.

10.6 Finality State Classification

Where finality is recorded, the obligated party MUST classify the finality state of the payment at the verification time as exactly one of:

State	Meaning
`PENDING`	The declared finality rule has not yet been satisfied.
`CONDITIONAL`	The declared finality rule is satisfied, but a declared condition may still reverse the payment without a new payment instruction.
`REVERSIBLE`	Operational finality is reached, and reversal remains available through ordinary process such as refund, dispute, or chargeback.
`IRREVOCABLE`	Operational finality is reached and no ordinary process available to the obligated party can reverse the payment. Legal irrevocability is not asserted unless §10.5 is satisfied.

The classification MUST be consistent with the settlement evidence and the status history. Where the classification is missing, or contradicts the evidence or the status history, the result MUST be FAIL using `RMFPAY-FIN-004`.

The classification MUST be determined from the declared finality rule and the elapsed refund and dispute windows of the payment network, not from the generic transition table in §18.4. The transition table states which transitions are structurally permitted; it does not state whether a reversal remains available for this payment at the verification time.

`IRREVOCABLE` MUST be recorded only where all of the following hold at the verification time:

1. operational finality under §10.3 or §10.4 has been reached;
2. every refund, dispute, and chargeback window declared by the payment network for this payment type has expired, and the expiry MUST be evidenced;
1. the obligated party retains no ordinary process by which it can reverse the payment.

Where any of these does not hold, `REVERSIBLE` or `CONDITIONAL` MUST be recorded instead. Where `IRREVOCABLE` is recorded and any adjustment under §12.2 subsequently occurs, the classification was incorrect and the result MUST be FAIL using `RMFPAY-FIN-004`.

11. Amounts, Fees, and Exchange Rates

11.1 Amount Conservation

The payment principal, the total amount debited from the payer, the fees, the exchange rate, and the net amount received by the payee MUST satisfy the declared amount formula and MUST be re-computable. Where the recomputation difference is not zero minor units, the result MUST be FAIL using `RMFPAY-AMT-002`.

The declared amount formula is:

```
total_debited  = gross_amount + fees borne by the payer
net_amount     = converted(gross_amount) − fees borne by the payee
converted(x)   = x × exchange_rate, rounded per §11.2
refundable_principal = gross_amount of the original payment (fees excluded)
cumulative_refund ≤ refundable_principal
```

11.2 Exchange Rates

Where a payment involves conversion of currency or unit, the record MUST capture the source unit, the target unit, the exchange rate, the rate source, the rate observation time, and the rounding mode. Where any of these is missing, the result MUST be FAIL using `RMFPAY-FX-001`.

Intermediate computation MUST retain at least 18 decimal places, or the full precision supported by the payment unit, whichever is greater. The default rounding mode is round-half-even, and the final result MUST be rounded to the decimals declared for the target unit.

11.3 Fees

Payment fees MUST be recorded separately from the principal and MUST identify the fee-bearing party, the fee recipient, the fee amount, and the denomination unit. Where fees are not separated or the bearing party is not identified, the result MUST be FAIL using `RMFPAY-FEE-001`.

12. Failures, Refunds, Reversals, and Disputes

12.1 Failure Records

Where a payment fails, the record MUST contain the failure time, the failure stage, the failure code, the failure source, and whether any movement of funds occurred. Where any of these is absent, the result MUST be FAIL using `RMFPAY-ERR-001`.

12.2 Refunds, Reversals, and Chargebacks

A refund, reversal, or chargeback MUST reference the original `payment_id` and MUST record an independent event identifier, the amount, the time, the reason, and the external reference. Where the original payment is not referenced or event fields are missing, the result MUST be FAIL using `RMFPAY-REV-001`.

12.3 Refund Limits

After a partial refund, the cumulative refunded amount MUST NOT exceed the refundable principal of the original payment. After a full refund, the status MUST be `REFUNDED_FULL`. Where the cumulative refund exceeds the limit or the full-refund status is incorrect, the result MUST be FAIL using `RMFPAY-REV-002`.

12.4 Disputes

Every dispute MUST have a unique `dispute_id` and MUST record the opening time, the disputed amount, the reason, the current status, and the final disposition. Where the dispute record is missing, not unique, or the disposition is incomplete, the result MUST be FAIL using `RMFPAY-DSP-001`.

13. Material Events

13.1 Types of Material Event

A material event includes at least: a change of a verified payment from `SETTLED` to `REVERSED`, `CHARGEBACK`, or `REFUNDED_FULL`; evidence that becomes unverifiable; withdrawal or invalidation of settlement evidence; compromise of a signing key used for authorization or verification; and any other event that changes the conformance conclusion.

13.2 Notice Deadline

The obligated party MUST record and publish an updated status within 24 hours after discovering a material event. Where this deadline is not met, the result MUST be FAIL using `RMFPAY-EVT-001`.

13.3 Notice Content

```
{
  "event_id": "",
  "event_type": "",
  "affected_payment_id": "",
  "detected_at": "",
  "published_at": "",
  "previous_status": "",
  "current_status": "",
  "description": "",
  "evidence_id": "",
  "requires_reverification": true
}
```

13.4 Verification Status Handling

Where a material event may affect conformance, the existing verification result MUST be treated as expired from the time of the event, and re-verification MUST be performed before the result is relied upon again.

14. Evidence Requirements

14.1 Required Evidence

Evidence ID	Evidence Type	Provided By	Required
EVD-001	Conformity subject and obligated party identity record	Obligated party; company registry or trusted identity system	Mandatory
EVD-002	Original payment instruction	Obligated party	Mandatory
EVD-003	Payer, payee, and agent authorization relationship record	Obligated party; identity or authorization system	Mandatory
EVD-004	Payment authorization evidence	Obligated party; payer system; signing system	Mandatory
EVD-005	Submission and external proof of receipt	Payment network, bank, ledger, processor	Mandatory
EVD-006	Payment status history and idempotency record	Obligated party	Mandatory
EVD-007	Settlement evidence	Bank, clearing system, payment network, blockchain, or equivalent	Mandatory where `SETTLED`
EVD-008	Finality rules and finality evidence	Obligated party, payment network, ledger, or legal counsel	Mandatory where finality is asserted
EVD-009	Fee and exchange rate evidence	Obligated party; liquidity or rate provider	Mandatory where fees or conversion apply
EVD-010	Failure, refund, reversal, and chargeback events	Obligated party, payment network, bank	Mandatory where such events occur
EVD-011	Dispute record	Obligated party; dispute-handling system	Mandatory where a dispute occurs
EVD-012	Record retention and deletion policy proof	Obligated party	Mandatory

Evidence ID	Evidence Type	Provided By	Required
EVD-013	Machine verification record and computation detail	Machine validator; verifier	Mandatory
EVD-014	Human disclosure version	Obligated party	Mandatory
EVD-015	Verifier digital signature and key information	Verifier	Mandatory
EVD-016	Material event notice record	Obligated party	Mandatory where such an event occurs
EVD-017	Status query interface response and availability proof	Obligated party	Optional (SHOULD)

14.2 Evidence Metadata

All mandatory evidence MUST contain the source, the generation time, the acquisition time, the content digest, and access location information. Where any of these is absent, the result MUST be FAIL using `RMFPAY-EVD-001`.

```
{
  "evidence_id": "",
  "evidence_type": "",
  "provider_id": "",
  "generated_at": "",
  "acquired_at": "",
  "uri": "",
  "sha256": "",
  "signature_present": false
}
```

14.3 Evidence Digests

All evidence files and machine records MUST use SHA-256 digests, and a signed record MUST reference the canonical digest of the signed content. Where a digest does not match or the digest format is incorrect, the result MUST be FAIL using `RMFPAY-EVD-002`.

The digest MUST be a 64-character lowercase hexadecimal string computed over the canonicalized content.

14.4 Evidence Freshness

Evidence of the current payment status MUST have an age of no more than 24 hours at the verification time. Terminal settlement evidence does not expire through the passage of time, but its integrity MUST remain verifiable. Where the current-status evidence exceeds the maximum age, the result MUST be FAIL using `RMFPAY-EVD-003`.

14.5 Record Retention

Payment records, status history, and mandatory evidence MUST be retained for at least 7 years from the terminal state, unless applicable law requires a longer period. Where the retention period is insufficient or cannot be demonstrated, the result MUST be FAIL using `RMFPAY-RET-001`.

14.6 Missing Evidence

Where any mandatory evidence is missing, the overall result MUST be `FAIL`, and `CONDITIONAL` MUST NOT be used in its place.

14.7 Conflicting Evidence

Where evidence conflicts, the verifier MUST record both sides of the conflict, their source and time, stop the PASS determination for the affected requirements, and determine the overall result as FAIL until the conflict is resolved.

14.8 Evidence Authority Ordering

Evidence sources are ordered by authority as follows, from highest to lowest:

1. distributed ledger or blockchain record confirmed under the declared finality rule;
2. settlement confirmation issued by a bank, clearing system, or payment network;
3. payment gateway or processor record;
4. application log or internal system record of the obligated party.

This ordering MUST be recorded for each item of conflicting evidence and MUST be used when reporting a conflict. It MUST NOT be used to resolve a conflict automatically: §14.7 continues to apply, and a conflict between sources of any authority level still produces FAIL until it is resolved.

An application log or internal record of the obligated party MUST NOT be the sole evidence for settlement, finality, or authorization.

15. Verification Procedure

15.1 Verification Sequence

The verifier MUST execute in the following order:

1. confirm the standard identifier, version, and machine identifier in use;
2. confirm the conformity subject type and the uniqueness of `conformity_subject_id` and `payment_id` or `batch_id`;
3. confirm the obligated party and its responsibility for the declared scope;
4. confirm the verification time in UTC, declare the authoritative time source under §18.8, and freeze the evidence set used for this verification;
5. verify that `payment_id` is unique within the declared network and environment;
6. verify the identity or protected alias mapping of the payer, payee, agent, and delegating principal;
7. verify that the original payment instruction contains the amount, unit, both parties, creation time, and payment purpose;
8. verify that amounts use decimal strings with explicit decimals and that no floating-point canonical amount is used;
9. verify that authorization evidence is bound to the payment instruction digest and that the authorization time is not later than the first execution time, distinguishing authentication from authorization;
10. verify idempotency key uniqueness and search for duplicate successful payments arising from the same payment intent or key;
11. verify submission and the external proof of receipt;
12. replay the status history and confirm that all transitions are permitted, times are monotonic, and terminal transitions conform to the rules;
13. for `SETTLED` payments, verify the settlement evidence source, digest, amount, time, and external reference;
14. verify the operational finality rules and the finality state classification under §10.6; where legal finality is asserted, verify the independent legal opinion;
15. re-compute the principal, total debited, fees, net amount, and cumulative refunds, requiring a difference of zero minor units;
16. for conversion and fees, verify the exchange rate, source, time, rounding mode, and fee-bearing party;
17. for failed payments, verify the failure stage, code, source, and whether funds moved;
18. for refunds, reversals, and chargebacks, verify the original payment reference, independent event identifier, amount limit, and final status;
19. for disputes and material events, verify the unique identifier, status history, final disposition, and the 24-hour notice requirement;

20 verify that the record retention policy is at least 7 years from the terminal state;

21 verify the source, generation time, acquisition time, age, digest, and access location of all mandatory evidence;

22 verify that key fields are consistent between the machine data and the human disclosure;

23 re-compute all SHA-256 digests and verify evidence chain integrity;

24 verify the digital signature, signature algorithm, key identifier, and public key or certificate location;

25 compute each requirement result and record the evidence used, the verification step, and any failure code;

26 compute the overall result under §16 and re-compute the key amount and status results using a second implementation;

27 sign the final verification record and record the signing time.

15.2 Determinism Requirement

The verification procedure MUST use fixed rules. Requirements MUST NOT be lowered based on the obligated party's size, brand, market share, or reputation.

15.3 Individual Result

Each requirement MUST record an applicability of `APPLICABLE` or `NOT_APPLICABLE`, and each applicable requirement MUST produce one of `PASS`, `FAIL`, or `NOT_ASSESSED`. Where the result is `FAIL`, a failure code from §17 MUST be recorded.

15.4 NOT_APPLICABLE

A requirement may be marked `NOT_APPLICABLE` only where this standard expressly permits and the verifier records the reason. It MUST NOT be used to bypass a generally applicable MUST requirement.

15.5 Re-computability

The verification record MUST contain, for every applicable requirement, the result, the evidence used, the verification step, and any failure code, such that the overall result can be re-computed by a second conforming implementation. Where the result cannot be re-computed, the result MUST be FAIL using `RMFPAY-VER-001`.

15.6 Verification Signature

The verification record MUST be signed by the verifier using a verifiable digital signature, and MUST include the signing time, the signature algorithm, the key identifier, and the certificate or public key location. Where the signature is missing, invalid, or cannot be verified, the result MUST be FAIL using `RMFPAY-SIG-001`.

15.7 Payment Lifecycle Overview

The following diagram is informative and does not create requirements. It shows where each normative section applies across the payment lifecycle.

```
Payment Instruction   (§6)
        |
    Authorization     (§7)      <- idempotency assigned (§8)
        |
     Submission       (§9.1)
        |
      Execution       (§9.2)
        |
      Settlement      (§10.1-10.2)
        |
      Finality        (§10.3-10.6)
        |
    Verification      (§15)  -> Determination (§16) -> Signature (§15.6)

At any point after Settlement:
    Refund / Reversal / Chargeback (§12.2-12.3)
```

```
Dispute                      (§12.4)
Material Event               (§13)
```

16. Conformance Determination

16.1 PASS

The overall result may be PASS only when all of the following are simultaneously satisfied:

1. all applicable MUST requirements are PASS;
2. no MUST NOT is violated;
3. all mandatory evidence is present, complete, and within its stated validity;
4. key fields are consistent between the human disclosure and the machine record;
5. there are no unresolved evidence conflicts;
6. the payment amounts, fees, exchange rate, net amount, and refunds are re-computable with a difference of zero minor units;
7. the status transitions are permitted, and settlement evidence exists where the status is `SETTLED`;
8. no duplicate successful payment exists without an independent authorization;
9. the result can be re-computed by a second conforming implementation to the same overall result;
10. the verification record is signed by the verifier with a valid digital signature.

16.2 CONDITIONAL

The overall result may be CONDITIONAL only where all MUST and MUST NOT requirements are satisfied, all mandatory evidence is valid, and one or more SHOULD requirements are unsatisfied.

16.3 FAIL

The overall result MUST be FAIL where any of the following occurs, except where §16.6 applies:

- any applicable MUST is unsatisfied;
- any MUST NOT is violated;
- mandatory evidence is missing, expired, digest-mismatched, or its signature cannot be verified;
- `payment_id` is not unique, or the same payment intent produced a duplicate successful payment without independent authorization;
- the payment is declared `SETTLED` but no valid settlement evidence exists;
- a status transition does not conform to the permitted transition table;
- authorization occurred after first execution, or authorization is not bound to a specific payment instruction;
- the amount recomputation difference is not zero minor units;
- the cumulative refund exceeds the refundable principal of the original payment;
- key fields are inconsistent between the human disclosure and the machine record;
- a second verification implementation cannot re-compute the same result;
- the verification record is unsigned or the signature is invalid.

16.4 NOT ASSESSED

NOT ASSESSED is used where verification is not formally complete, the evidence set has not been frozen, or the verifier has not signed. It does not constitute a conformance conclusion.

16.5 Result Validity Period

Where the payment is not in a terminal state, the verification result MUST carry an expiry time no later than 24 hours after the verification time.

Where the payment is in a terminal state, the result has no fixed expiry, but it MUST expire immediately upon a material event under §13.

16.6 DEFICIENT

DEFICIENT is used where **no requirement is affirmatively violated on the evidence available, but one or more mandatory items cannot be verified** because the required evidence is missing, incomplete, or expired, and the gap is curable by supplying that evidence.

DEFICIENT MUST be used, in preference to FAIL, where all of the following hold:

1 every requirement that could be evaluated on the available evidence is satisfied;

2 one or more mandatory items could not be evaluated because evidence was absent,

incomplete, or beyond the freshness limits of §14.4;

1 no MUST NOT is violated and no evidence conflict under §14.7 is present.

Where any requirement is affirmatively violated on the available evidence, the result MUST be FAIL, not DEFICIENT. Where both a violation and a gap are present, FAIL takes precedence.

A DEFICIENT result MUST list every item that could not be verified and state, for each, the evidence that would resolve it.

Assessment from public information only. Where an assessment is not based on an evidence set submitted by the obligated party, but on publicly available information alone, the result MUST NOT be reported as FAIL on the ground that evidence is absent. Items that cannot be verified from the available information MUST be reported as DEFICIENT and identified individually. Such an assessment MUST state that the obligated party did not participate.

DEFICIENT states only that specified items could not be verified within the scope assessed. It is not a statement about the quality, soundness, or good faith of the obligated party.

17. Failure Codes

Code	Meaning
RMFPAY-ID-001	Conformity subject identifier incomplete or not unique
RMFPAY-ID-002	Obligated party, payer, payee, or agent identity cannot be confirmed
RMFPAY-ID-003	Payment identifier duplicated or uniqueness cannot be demonstrated
RMFPAY-INS-001	Payment instruction fields missing or incomplete
RMFPAY-AMT-001	Amount format, unit, or decimals non-conforming
RMFPAY-AMT-002	Principal, fees, net amount, or refunds cannot be conserved on recomputation
RMFPAY-AUT-001	Authorization evidence missing or authorization later than execution
RMFPAY-AUT-002	Authorization not bound to a specific payment instruction, or scope incomplete
RMFPAY-AUT-003	Authentication incorrectly treated as payment authorization
RMFPAY-DUP-001	Idempotency key missing, reused, or retention period insufficient
RMFPAY-DUP-002	Duplicate successful payment without independent authorization

Code	Meaning
RMFPAY-EXE-001	Submission or external proof of receipt missing
RMFPAY-STA-001	Status history missing, times discontinuous, or transition not permitted
RMFPAY-STA-002	Prohibited transition from a terminal state to settled
RMFPAY-SET-001	Settlement declared but settlement evidence missing or invalid
RMFPAY-SET-002	Settlement evidence lacks amount, time, status, source, or external reference
RMFPAY-FIN-001	Blockchain finality parameters or proof missing
RMFPAY-FIN-002	Operational finality rule or evidence missing for a non-blockchain payment
RMFPAY-FIN-003	Legal irrevocability asserted without legal verification
RMFPAY-FIN-004	Finality state classification missing or inconsistent with evidence
RMFPAY-FX-001	Exchange rate, source, time, or rounding mode missing
RMFPAY-FEE-001	Fees not separated from principal, or bearing party not identified
RMFPAY-ERR-001	Failure record lacks stage, code, source, or fund-movement information
RMFPAY-REV-001	Refund, reversal, or chargeback does not reference the original payment, or event fields missing
RMFPAY-REV-002	Cumulative refund exceeds the limit, or full-refund status incorrect
RMFPAY-DSP-001	Dispute record missing, not unique, or disposition incomplete
RMFPAY-RET-001	Record retention period insufficient or cannot be demonstrated
RMFPAY-DAT-001	Key fields inconsistent between human disclosure and machine record
RMFPAY-EVD-001	Mandatory evidence missing
RMFPAY-EVD-002	Evidence digest mismatch or incorrect digest format
RMFPAY-EVD-003	Current-status evidence exceeds the maximum age
RMFPAY-SIG-001	Digital signature missing, invalid, or unverifiable
RMFPAY-VER-001	Verification result cannot be re-computed by a second implementation
RMFPAY-EVT-001	Material event not recorded and updated within 24 hours
RMFPAY-TIM-001	Authoritative time source undeclared, unpermitted, or discrepancy beyond tolerance
RMFPAY-AGT-001	Machine or AI agent authorization fields incomplete
RMFPAY-AGT-002	Machine or AI agent payment over limit, out of scope, or after expiry

18. Machine Interface

18.1 Verification Record

```
{
  "record_id": "",
  "schema_version": "1.0",
  "verification_time": "",
  "result_expires_at": "",
  "standard": {},
  "conformity_subject": {},
  "obligated_party": {},
  "parties": {},
  "payment_instruction": {},
  "authorization": {},
  "execution": {},
  "status_history": [],
  "settlement": {},
  "finality": {},
  "fees": [],
  "exchange_rate": {},
  "adjustments": [],
  "disputes": [],
  "material_events": [],
```

```
  "evidence": [],
  "requirement_results": [],
  "human_disclosure": {},
  "overall_result": {},
  "verifier": {},
  "signature": {}
}
```

`settlement` MUST be present where the current status is `SETTLED`. `finality` MUST be present where finality is asserted. `exchange_rate` MUST be present where a conversion occurs. `agent` within `parties` MUST be present where the conformity subject type is `AGENT_PAYMENT`.

18.2 Numeric Format

All amounts MUST use decimal string representation and MUST declare the denomination unit and the number of decimals. Binary floating-point values MUST NOT be used as the canonical amount.

```
{
  "gross_amount": "100.00",
  "currency": "USD",
  "decimals": 2
}
```

18.3 Time Format

All times MUST use ISO 8601 UTC format with a precision of at least one second.

18.4 Permitted Status Transitions

Current Status	Permitted Next Status
INITIATED	AUTHORIZED / FAILED
AUTHORIZED	SUBMITTED / FAILED
SUBMITTED	PENDING_EXTERNAL_CONFIRMATION / EXECUTING / SETTLED / FAILED
PENDING_EXTERNAL_CONFIRMATION	EXECUTING / SETTLED / FAILED
EXECUTING	SETTLED / FAILED / REVERSED
SETTLED	REFUNDED_PARTIAL / REFUNDED_FULL / CHARGEBACK
REFUNDED_PARTIAL	REFUNDED_PARTIAL / REFUNDED_FULL / CHARGEBACK
FAILED	none; a new `payment_id` MUST be created to attempt payment again
REVERSED	none; a new `payment_id` MUST be created to attempt payment again
REFUNDED_FULL	none
CHARGEBACK	none, unless the dispute disposition is expressed as an independent adjustment event

18.5 Human–Machine Consistency

The payment identifier, amount, currency, status, settlement time, and overall result MUST be consistent between the human disclosure and the machine record. Where they conflict, the result MUST be FAIL using `RMFPAY-DAT-001`.

18.6 Canonicalization and Digest

Before signing, a machine record MUST use a deterministic canonicalization procedure and generate a SHA-256 digest.

18.7 Digest Algorithm Identifier

The machine record MUST record the digest algorithm identifier. The default value is `sha-256`. A verifier MUST NOT use an unregistered digest algorithm to produce a formal verification record.

18.8 Authoritative Time Source

Different systems in a payment may report different times for the same event. To make the verification result reproducible, the following rules apply.

Declaration. The verification record MUST declare the authoritative time source used for each of: the verification time, the settlement time, and the finality time.

Permitted sources. Each declared source MUST be one of:

Source	Meaning
`UTC_SYSTEM`	The clock of the recording system, expressed in UTC.
`BLOCK_TIMESTAMP`	The timestamp of the block or ledger entry containing the transaction.
`SETTLEMENT_TIMESTAMP`	The time asserted by the settlement system, bank, clearing system, or payment network.
`TRUSTED_TIMESTAMP`	A timestamp issued by a trusted timestamping authority.

Precedence. Where sources disagree for the same event, the following order MUST be applied, from highest to lowest: `TRUSTED_TIMESTAMP`, `BLOCK_TIMESTAMP`, `SETTLEMENT_TIMESTAMP`, `UTC_SYSTEM`. The time actually used MUST be recorded together with its source.

Tolerance. Where two sources for the same event differ by more than 300 seconds, the verifier MUST record the discrepancy, and MUST NOT treat the event time as established.

Verification time. The verification time MUST use `UTC_SYSTEM` or `TRUSTED_TIMESTAMP`, and MUST be a single value for the whole verification, in accordance with §4.3.

Where the authoritative time source is not declared, an unpermitted source is used, or a discrepancy exceeds the tolerance without being recorded, the result MUST be FAIL using `RMFPAY-TIM-001`.

19. Security Considerations

19.1 Principal Threats

This standard considers at least: forged settlement evidence; replay of an authorization against a second payment; duplicate payment through idempotency key reuse; back-dating of authorization; substitution of internal status display for settlement evidence; misrepresentation of operational finality as legal finality; silent overwriting of status history; digest or signature forgery; exchange-rate or fee manipulation; refund amounts exceeding the principal; agent authorization escalation beyond delegated limits; selective omission of failure, dispute, or chargeback records; and verifier conflict of interest.

19.2 Security Controls

The obligated party and the verifier MUST verify evidence digests and signatures, retain append-only status history, record provenance, prevent silent overwriting, bind authorization to the payment instruction digest, enforce idempotency key uniqueness, retain failure results, and sign the verification record.

19.3 Verifier Independence

The verifier MUST disclose commercial, fee, ownership, control, and other interest relationships with the obligated party that may affect independence. An undisclosed material interest relationship renders the verification result invalid.

19.4 Residual Risk

Even where the result is PASS, there may still be subsequent reversal, chargeback, or court-ordered recovery; undetected fraud; insolvency of a bank, processor, or counterparty; compromise of a key after the verification time; a different legal interpretation of finality; or concealment of facts by an evidence provider.

19.5 Threat Mapping Matrix

Threat	Control Requirement	Evidence	Failure Code
Forged settlement evidence	§10.1, §14.3	EVD-007, EVD-013	RMFPAY-SET-001, RMFPAY-EVD-002
Authorization replay	§7.2	EVD-004	RMFPAY-AUT-002
Duplicate payment via key reuse	§8.1, §8.2	EVD-002, EVD-006	RMFPAY-DUP-001, RMFPAY-DUP-002
Back-dated authorization	§7.1	EVD-004, EVD-006	RMFPAY-AUT-001
Internal status shown as settlement	§10.1, §4.9	EVD-007	RMFPAY-SET-001
Operational finality claimed as legal	§10.5	EVD-008, EVD-015	RMFPAY-FIN-003
Silent overwriting of status history	§9.2, §19.2	EVD-006	RMFPAY-STA-001
Digest or signature forgery	§14.3, §15.6	EVD-013, EVD-015	RMFPAY-EVD-002, RMFPAY-SIG-001
Exchange rate or fee manipulation	§11.2, §11.3	EVD-009	RMFPAY-FX-001, RMFPAY-FEE-001
Refund exceeding principal	§12.3, §3	EVD-010	RMFPAY-REV-002
Finality overstated as irrevocable	§10.6	EVD-007, EVD-008	RMFPAY-FIN-004
Agent authorization escalation	§7.4, §7.5	EVD-003, EVD-004	RMFPAY-AGT-001, RMFPAY-AGT-002
Omission of dispute or chargeback	§12.4, §13.2	EVD-011, EVD-016	RMFPAY-DSP-001, RMFPAY-EVT-001
Verifier conflict of interest	§1.5, §19.3	—	verification result invalid
Time source manipulation	§18.8	EVD-005, EVD-007	RMFPAY-TIM-001
Unverifiable result	§15.5	EVD-013	RMFPAY-VER-001

20. Conformance Claim

20.1 Minimum Fields

A conformance claim MUST include at least the standard identifier, the standard version, the conformity subject type and identifier, the obligated party, the payment network, the environment, the implementation and version, the verification time, the result, the evidence set digest, the verifier identity, the result expiry where applicable, the signature, and any failure codes.

20.2 Example

```
{
  "standard_id": "RM-S-PAY-001",
  "standard_version": "v1.0-F",
  "conformity_subject_type": "SINGLE_PAYMENT",
  "conformity_subject_id": "payment:example:0001",
  "obligated_party_id": "psp:example",
  "payment_network": "SEPA_INST",
  "environment": "PRODUCTION",
  "verification_time": "2026-01-01T00:00:00Z",
  "result_expires_at": "2026-01-02T00:00:00Z",
  "verification_result": "PASS",
  "evidence_set_sha256": "",
  "verifier_id": "",
```

```
  "signature": ""
}
```

20.3 Prohibited Statements

A conformance claim MUST NOT use expressions such as "completely safe", "zero risk", "government approved", "RuleMark guaranteed", "permanently valid", "legally irrevocable" without verified legal finality, "funds source verified", or "AML compliant".

21. Test Cases

21.1 PASS Case

A production payment has a unique `payment_id` and idempotency key; the payer authorized the payment before execution with the authorization bound to the instruction digest; the payment network returned a proof of receipt; the settlement system confirmed a principal of 100.00 USD as settled; the payer paid an additional 1.00 USD processing fee and the payee received a net 100.00 USD; the amount recomputation difference is zero; there are no refunds, reversals, chargebacks, or disputes; all evidence digests and the verifier signature are valid; a second verification implementation obtains the same result.

Expected result: `PASS`

21.2 Settlement Evidence Missing

The machine record marks the payment `SETTLED`, but supplies only an internal database status, with no settlement evidence from a bank, payment network, blockchain, or equivalent source.

Expected result: `DEFICIENT`, `RMFPAY-SET-001`

Settlement evidence is absent rather than contradicted, so §16.6 applies.

21.3 Authorization Evidence Missing

The payment has a settlement receipt, but no authorization evidence bound to the specific payment instruction, and no authorization evidence item exists in the evidence index.

Expected result: `DEFICIENT`, `RMFPAY-AUT-001`, `RMFPAY-EVD-001`

Authorization evidence is absent rather than contradicted, so §16.6 applies.

21.4 Data Conflict

The machine record shows a settled amount of 250.00 EUR while the human disclosure shows 205.00 EUR; all other fields agree.

Expected result: `FAIL`, `RMFPAY-DAT-001`

21.5 SHOULD Not Satisfied

All MUST and MUST NOT requirements are satisfied, and settlement, authorization, status, amounts, evidence, and signature are all valid; the payment service does not provide a public or authorized API to query current status by `payment_id`, so §9.4 is unsatisfied.

Expected result: `CONDITIONAL`

21.6 Duplicate Payment

The same idempotency key produced two successful settlements within 10 seconds, and the second has no independent authorization.

Expected result: `FAIL`, `RMFPAY-DUP-002`

21.7 Agent Exceeding Limit

An AI agent has a maximum single-payment authorization of 500.00 USD but initiated and settled 750.00 USD, with no additional authorization.

Expected result: `FAIL`, `RMFPAY-AGT-002`

21.8 Prohibited Status Transition

A payment first entered `FAILED`, and the same `payment_id` was then changed directly to `SETTLED`, with no new payment and no new authorization.

Expected result: `FAIL`, `RMFPAY-STA-002`

22. Requirement Traceability Matrix

Requirement Area	Clause	Core Evidence	Verification Step	Failure Code
Conformity subject identity	§6.1	EVD-001	2	RMFPAY-ID-001
Payment identifier uniqueness	§6.2	EVD-002	5	RMFPAY-ID-003
Party identification	§6.3	EVD-001, EVD-003	3, 6	RMFPAY-ID-002
Payment instruction content	§6.4	EVD-002	7	RMFPAY-INS-001
Amount representation	§6.5	EVD-002, EVD-013	8	RMFPAY-AMT-001
Authorization existence and ordering	§7.1	EVD-004	9	RMFPAY-AUT-001
Authorization content	§7.2	EVD-004	9	RMFPAY-AUT-002
Authentication is not authorization	§7.3	EVD-004, EVD-005	9	RMFPAY-AUT-003
Agent authorization fields	§7.4	EVD-003, EVD-004	6, 9	RMFPAY-AGT-001
Agent limits	§7.5	EVD-003, EVD-004, EVD-006	9, 12	RMFPAY-AGT-002
Idempotency key	§8.1	EVD-002, EVD-006	10	RMFPAY-DUP-001
Duplicate payment	§8.2	EVD-006, EVD-007	10, 15	RMFPAY-DUP-002
Proof of receipt	§9.1	EVD-005	11	RMFPAY-EXE-001
Status transitions	§9.2	EVD-006	12	RMFPAY-STA-001
Prohibited terminal transitions	§9.3	EVD-006	12	RMFPAY-STA-002
Settlement evidence	§10.1	EVD-007	13	RMFPAY-SET-001
Settlement evidence content	§10.2	EVD-007	13	RMFPAY-SET-002
Settled amount reconciliation	§10.2, §11.1	EVD-007, EVD-009	13, 15	RMFPAY-AMT-002
Blockchain finality	§10.3	EVD-007, EVD-008	14	RMFPAY-FIN-001
Non-blockchain finality	§10.4	EVD-007, EVD-008	14	RMFPAY-FIN-002

Requirement Area	Clause	Core Evidence	Verification Step	Failure Code
Legal finality	§10.5	EVD-008, EVD-015	14	RMFPAY-FIN-003
Finality state classification	§10.6	EVD-007, EVD-008	14	RMFPAY-FIN-004
Amount conservation	§11.1	EVD-002, EVD-007, EVD-009, EVD-013	15	RMFPAY-AMT-002
Exchange rates	§11.2	EVD-009	16	RMFPAY-FX-001
Fees	§11.3	EVD-009	16	RMFPAY-FEE-001
Failure records	§12.1	EVD-006, EVD-010	17	RMFPAY-ERR-001
Refunds and reversals	§12.2	EVD-010	18	RMFPAY-REV-001
Refund limits	§12.3	EVD-010, EVD-013	18	RMFPAY-REV-002
Disputes	§12.4	EVD-011	19	RMFPAY-DSP-001
Material events	§13.2	EVD-010, EVD-011, EVD-016	19, 21	RMFPAY-EVT-001
Evidence metadata	§14.2	EVD-001 to EVD-015	21	RMFPAY-EVD-001
Evidence digests	§14.3	EVD-013, EVD-015	23	RMFPAY-EVD-002
Evidence freshness	§14.4	EVD-006, EVD-007	21	RMFPAY-EVD-003
Record retention	§14.5	EVD-012	20	RMFPAY-RET-001
Re-computability	§15.5	EVD-013, EVD-015	25, 26	RMFPAY-VER-001
Verification signature	§15.6	EVD-015	24, 27	RMFPAY-SIG-001
Human–machine consistency	§18.5	EVD-014	22	RMFPAY-DAT-001
Authoritative time source	§18.8	EVD-005, EVD-007, EVD-013	4, 13	RMFPAY-TIM-001
Status query interface (SHOULD)	§9.4	EVD-017	21	—
Terminal record generation (SHOULD)	§9.5	EVD-006, EVD-017	21	—
Unknown external state (SHOULD)	§9.6	EVD-006	12	—

23. Lifecycle

23.1 Draft

In Draft status, this standard MUST NOT serve as a basis for formal RuleMark conformance. It may be used for public consultation and test verification, and verification results MUST be marked experimental.

23.2 Frozen

Once formally published, this standard may be referenced and implemented.

23.3 Frozen

A version may be frozen only after all of the following are satisfied:

- the automated consistency check reports zero errors;
- each MUST is mapped to evidence, a verification step, and a failure code;
- the machine schema accepts a conforming record and rejects non-conforming records;
- failure codes are stable;

- test cases pass;
- the human and machine versions are consistent;
- all normative references are confirmed to exist. This standard is self-contained; normative references are limited to the public technical specifications listed in §24.1;
- a SHA-256 digest has been generated for the normative files and the digest manifest has been signed by RuleMark;
- RuleMark sovereign approval has been recorded, stating the approver, the date, the approved version, and the approved digest.

Independent review by an external expert is recommended and, where performed, MUST be recorded with the reviewer's identity, the reviewed version, and the reviewed digest. It is not a condition of freezing.

23.4 Superseded

Where a new version replaces this version, it MUST record the new version number, effective date, compatibility changes, migration rules, and the last valid date of the old version.

23.5 Withdrawn

After withdrawal, this standard MUST NOT be used for new conformance claims. Historical verification records MUST continue to be retained.

24. References and Dependencies

24.1 Normative Technical References

- ISO 8601 date and time representation;
- ISO 4217 currency codes;
- ISO 3166-1 alpha-2 country codes;
- RFC 3339 Internet timestamps;
- RFC 3986 URI;
- RFC 8785 JSON Canonicalization Scheme;
- JSON Schema Draft 2020-12;
- SHA-256;
- JWS, COSE, or an equivalent digital signature container;
- BCP 47 language tags.

24.2 RuleMark Dependencies

This standard is self-contained and does not depend on any external RuleMark standard. Canonical identity, evidence, verification and determination, digital signature, and lifecycle rules are defined internally at §5, §14, §15, §16, §15.6, and §23 respectively.

Where RuleMark later issues corresponding core-layer standards, a subsequent version of this standard MAY align these internal clauses with normative references to those standards. Until then, this standard MUST NOT cite an identifier that has not been issued.

24.3 Informative References

ISO 20022 payment message semantics may be used as a mapping reference. PCI DSS, EMV, SWIFT, and card scheme rules may be consulted as implementation references. Informative references do not automatically become normative requirements of

this standard.

25. Implementation Requirements

An implementation conforming to this standard MUST be able to:

1. uniquely identify the conformity subject and the payment;
2. record and validate the payment instruction and its canonical digest;
3. bind authorization evidence to a specific payment instruction and verify its ordering against execution;
4. enforce idempotency key uniqueness and detect duplicate successful payments;
5. retain an append-only status history and validate transitions against §18.4;
6. verify settlement evidence from an authoritative external source;
7. classify operational and legal finality separately, and record the finality state under §10.6;
8. re-compute the principal, fees, exchange rate, net amount, and cumulative refunds to zero minor units of difference;
9. record failures, refunds, reversals, chargebacks, and disputes with their required fields;
10. record material events and enforce the 24-hour notice deadline;
11. verify evidence metadata, age, and digests, and apply the authoritative time source rules of §18.8;
12. verify consistency between the machine record and the human disclosure;
13. produce a deterministic conformance result with per-requirement traceability;
14. sign and retain the verification record;
15. permit a second conforming implementation to re-compute the same overall result.

26. Final Non-Claim

A PASS result under RM-S-PAY-001 indicates only:

> At the specified verification time, under the specified standard version, for the specified conformity subject and the specified evidence set, all applicable mandatory requirements of RM-S-PAY-001 are satisfied.

PASS does not indicate future continued conformance, the lawfulness of the source of funds, KYC, AML, sanctions, or tax compliance, the absence of fraud, the outcome of any dispute, legal irrevocability of the transfer, the solvency of any party, regulatory approval, an investment recommendation, or a RuleMark guarantee.

Document Status Declaration

```
Standard ID: RM-S-PAY-001
Version: v1.0-F
Status: FROZEN
Publication Date: 2026-07-23
Effective Date: 2026-07-23
Normative Status: NORMATIVE
Conformance Use: AUTHORIZED
Machine ID: rulemark:standard:rm-s-pay-001:v1.0-F
```

This standard MUST NOT be marked `FROZEN` before the automated consistency check reports zero errors, the machine schema has been tested against conforming and non-conforming records, the digest manifest has been signed by RuleMark, and RuleMark sovereign approval has been recorded.