ISO/IEC 27560-1 Universal Notice Receipt, (a Convention 108+ Code of Conduct) v1.1 (Draft)

ISO/IEC 27560-1 Universal Notice Receipt, (a Convention 108+ Code of Conduct) v1.1

Draft Specification - Enhanced Transparency Features

Status: WORKING DRAFT

Version: 1.1 (Draft)

Base Profile: ISO/IEC 27560-1 Universal Notice Receipt, (a Convention 108+ Code of Conduct) v1.0

Author: Mark Lizar, Digital Transparency Lab

Date: December 2025

Document Purpose

This document consolidates all features deferred from v1.0 for standardization in v1.1. These enhancements build upon the core co-regulated transparency infrastructure established in v1.0 to address:

- **Privacy preference signals** (GPC, DNT integration)
- **Transparency and Trust Assurance (TATA)** levels and visual signaling
- **Global Privacy Rights Controls (GPRC)** framework
- **OAuth/UMA role mapping** patterns
- **Consent token specifications** (cryptographic binding, verification protocols)

- **AI system transparency** (ISO/IEC 42001 coordination)
- **Active state validation** mechanisms
- **Enhanced cross-border transfer** protocols

Development Timeline: Q1-Q4 2026

Development Timeline: Q1-Q4 2026

CRITICAL TERMINOLOGY: "User" vs "Individual"

Why This Matters

This specification **avoids the term "user"** when referencing people. The word "user" misrepresents the power dynamic in consent relationships and obscures who is actually using whose data.

The Problem with "User"

When consent is the legal basis to process personal data, it is **NOT the individual who is the "user"**—it is the **service that is using the individual's data with consent**.

- ❌ **Wrong framing**: "Users consent to data processing"
- ✅ **Correct framing**: "Individuals grant consent; services use data with that consent"

This distinction is not semantic—it reflects the actual legal and technical relationship:

- **Individuals grant authorization** (consent, contract, legal obligation, etc.)
- **Controllers use personal data** under that authorization
- **Services consume data**; individuals provide it

Terminology Standards in This Specification

Context	Correct Term	Definition	ISO/Legal Basis
ISO documents	**PII Principal**	The natural person to whom personally identifiable information relates	ISO/IEC 29100:2011 (3.1)

Context	Correct Term	Definition	ISO/Legal Basis
Data protection law	**Data subject**	Identified or identifiable natural person whose personal data is processed	GDPR Art. 4(1), Convention 108+ Art. 3(a)
Canadian/general	**Individual**	Natural person who typically acts as both PII Principal and data subject	PIPEDA s.2(1)
When "user" required	**Qualified usage**	Always specify: "service user," "software user," "device user," "application user"	Context-specific

Examples in Context

Consent Management:

- ❌ "The user's consent is recorded"
- ✅ "The individual's consent is recorded" OR "The PII Principal's consent is recorded"

Rights Exercise:

- ❌ "Users can access their data"
- ✅ "Individuals can access their data" OR "Data subjects can access their data"

Service Relationship:

- ❌ "Users agree to terms"
- ✅ "Service users agree to terms" (qualifying "user" with "service")

AI Training Data:

- ❌ "User data trains the model"
- ✅ "Data from individuals trains the model" OR "PII Principal data trains the model"

Implementation Requirement

All implementations of this specification SHALL:

1. Avoid unqualified "user" terminology in documentation, APIs, and UIs
2. Use "PII Principal" in ISO contexts
3. Use "data subject" in regulatory/legal contexts

4. Use "individual" in general communication
5. Qualify "user" when context requires it (e.g., "mobile app user" for someone using the app)

Rationale: Correcting the Power Dynamic

Traditional "user" framing implies:

- The individual "uses" the service's infrastructure
- The service is doing the individual a favor
- Consent is a favor granted to the individual

Correct framing recognizes:

- The service **uses the individual's data** as a resource
- The individual **grants permission** for that use
- Consent is authorization the individual provides to the service

This terminology shift aligns with:

- **Convention 108+ Article 5**: Lawful basis requirements place obligation on controller
- **ISO/IEC 29100 privacy principles**: PII Principal as rights holder
- **Transparency by design**: Power relationship made explicit in language

This terminology shift aligns with:

- **Convention 108+ Article 5**: Lawful basis requirements place obligation on controller
- **ISO/IEC 29100 privacy principles**: PII Principal as rights holder
- **Transparency by design**: Power relationship made explicit in language

Digital Receipt vs Notice Receipt

The Problem with "Digital Receipts Without Notice"

A **digital receipt provided without notice**—such as a cookie—is fundamentally different from a Notice Receipt:

Characteristic	Digital Receipt Without Notice (e.g., Cookie)	Notice Receipt (This Specification)
Timing	Deployed before or without transparent disclosure	Generated AFTER notice, WITH individual awareness
Controller Identity	Hidden or obscured until after processing begins	Controller-ID disclosed BEFORE processing (Controller-ID first)
Bilateral Proof	Controller holds evidence; individual unaware	BOTH parties hold synchronized proof
Architecture Pattern	Surveillance-by-default ("trust us")	Transparency-by-default ("I choose to trust")
Individual Control	Retroactive opt-out at best	Preventive transparency enabling informed choice
Compliance Model	Post-hoc enforcement; fines after harm	Preventive enforcement; verifiable before processing

Why This Distinction Matters

Cookies as "Digital Receipts Without Notice":

- Cookie banner appears → individual clicks "Accept" → cookies already deployed
- No Controller Identification Record (CIR) published before cookie deployment
- No bilateral proof—controller knows what was tracked; individual does not
- Consent obtained through dark patterns, not informed choice
- Result: **Surveillance-by-default with retroactive "consent"**

Notice Receipts (This Specification):

- Controller publishes CIR (Stage 1: Notice Receipt) → individual reviews controller identity → individual authorizes (Stage 2: Authorization Receipt)
- Controller identity disclosed BEFORE any processing
- Bilateral proof—both parties hold synchronized records
- Notice Event Log tracks all changes to transparency state
- Result: **Transparency-by-default with preventive consent**

Implementation Requirement

This specification uses the term **"Notice Receipt"** exclusively. "Digital receipt" without qualification refers to any digital record and does not imply transparency or bilateral proof.

When implementing this specification:

1. **Notice Receipts** SHALL be generated AFTER controller identity disclosure (Stage 1)
2. **Bilateral proof** SHALL be provided (both controller and individual hold copies)
3. **Controller-ID first** architecture SHALL be followed (CIR published before processing)
4. **Notice Event Log** SHALL maintain transparency state audit trail

Anti-Pattern: Deploying tracking mechanisms (cookies, pixels, scripts) before notice receipt generation violates the transparency-by-default requirement and creates surveillance architecture inconsistent with this specification.

PART 1: ENHANCED PRIVACY PREFERENCE SIGNALS

Section 1: Privacy Preference Signal Integration

1.1 Overview

Purpose: Enable automated privacy preference enforcement through standardized signals integrated with ANCR Exchange protocol.

Deferred from v1.0 Rationale: GPC integration and preference indicator infrastructure add implementation complexity beyond v1.0's core bilateral receipt architecture.

V1.1 Scope: Normative specification for privacy_preference_signals object, GPC scope mapping, and Notice Event Log integration.

1.2 privacy_preference_signals Object

Data type: Object

Cardinality: Optional (recommended when scope_of_disclosure includes automated processing)

Field Specification:

```
{
  "privacy_preference_signals": {
    "gpc_honored": true,
    "gpc_scope": ["analytics", "third_party_disclosure", "marketing"],
    "dnt_recognized": false,
    "preference_center_url": "https://example.com/privacy-preferences",
    "preference_binding": "browser_signal",
    "gpc_received_timestamp": "2025-12-13T10:00:00-05:00",
    "preference_enforcement_mechanism": "automatic"
  }
}
```

Field Definitions:

- **gpc_honored** (boolean, required): Whether W3C Global Privacy Control signal is respected
- **gpc_scope** (array of strings, conditional): Processing categories affected by GPC
    - Valid values: "analytics", "marketing", "third_party_disclosure", "advertising", "profiling", "cross_context_tracking"
- **dnt_recognized** (boolean, required): Whether Do Not Track signal is recognized
- **preference_center_url** (string, optional): URL where individuals can manage preferences
- **preference_binding** (enum, required): How preferences are enforced
    - Values: "browser_signal", "receipt_anchored", "account_based", "token_bound"
- **gpc_received_timestamp** (ISO 8601 datetime, conditional): When GPC signal was first received (required if gpc_honored=true)
- **preference_enforcement_mechanism** (enum, required): Enforcement approach
    - Values: "automatic", "manual", "hybrid"

1.3 GPC Scope Mapping to permissions_bundle

When GPC signal is received and honored, controllers SHALL:

1. Update permissions_bundle to reflect restricted processing
2. Log preference signal receipt in Notice Event Log
3. Generate material change notification if GPC restricts previously authorized processing

Example GPC Integration:

```
{
  "controller_identity_record_id": "CIR-CA-12345",
  "privacy_preference_signals": {
    "gpc_honored": true,
    "gpc_scope": ["third_party_disclosure", "marketing"],
    "preference_binding": "browser_signal",
    "gpc_received_timestamp": "2025-12-13T10:00:00-05:00",
    "preference_enforcement_mechanism": "automatic"
  },
  "permissions_bundle": [
    "service_delivery",
    "essential_analytics"
  ],
  "permissions_restricted_by_gpc": [
    "third_party_disclosure",
    "marketing",
    "cross_site_tracking"
  ]
}
```

1.4 Notice Event Log Integration

New Event Types:

- **preference_signal_received**: GPC or DNT signal detected
- **processing_restricted_by_preference**: Automated restriction applied
- **preference_override_requested**: Individual manually overrides signal
- **preference_center_accessed**: Individual accesses preference management UI

Event Log Entry Example:

```
{
  "event_time": "2025-12-13T10:00:00-05:00",
  "event_type": "preference_signal_received",
  "entity_id": "CIR-CA-12345",
  "preference_metadata": {
    "signal_type": "gpc",
    "signal_value": true,
    "scope_affected": ["third_party_disclosure", "marketing"],
    "enforcement_action": "permissions_restricted"
  }
}
```

1.5 Standards Alignment

- **W3C Global Privacy Control**: Implements GPC specification signal semantics
- **IEEE P7012**: Machine Readable Personal Privacy Terms standard alignment

- **ISO/IEC 27560:2025**: Extends permissions_bundle for preference-based gating

1.6 Implementation Requirements

Controllers claiming GPC conformance SHALL:

1. Honor GPC signal within 48 hours of receipt
2. Log GPC receipt with timestamp in Notice Event Log
3. Update permissions_bundle to reflect restricted processing
4. Provide preference_center_url for manual preference management
5. Generate material change notification if GPC restricts previously authorized processing

---

PART 2: DIGITAL TRANSPARENCY RISK ASSURANCE LEVELS

Section 2: Digital Transparency Risk Assurance Framework Specification

2.1 Overview

Purpose: Establish risk-proportionate transparency assurance levels enabling progressive trust from self-assertion to active state validation.

Deferred from v1.0 Rationale: Digital Transparency Risk Assurance level visual signaling and trustmark infrastructure requires coordinated implementation across registries, TTAOs, and visual standard bodies.

V1.1 Scope: Normative Digital Transparency Risk Assurance level requirements, trustmark certification protocols, and cross-registry coordination standards.

2.2 Four-Level Digital Transparency Risk Assurance Framework

Level 1: Self-Assertion

Description: Controller publishes CIR without registry verification.

Requirements:

- CIR published at /.well-known/transparency (or equivalent)

- Notice receipts optional
- Self-attested controller identity
- Suitable for: local/child scope of disclosure, minimal risk contexts

Conformance:

- SHALL publish CIR in structured format (JSON, XML, or YAML)
- SHOULD include controller_identity_record_id (even if self-generated)
- MAY omit cryptographic signatures

Use Cases:

- Local community services
- Child/youth scope disclosure
- Device-only processing (no network disclosure)

Level 2: Registry Verification + Blind Data Notary

Description: CIR-ID registered with authoritative source; registrar provides blind data notary signature.

Requirements:

- CIR registered with authoritative registry (ICO, OAIC, OPC, or equivalent)
- Registrar signs CIR without accessing PII Principal identifiers
- Individual generates receipts using CIR + registrar signature
- Self-generated receipts validated against registrar's public key
- Suitable for: regional/community/national scope of disclosure

Conformance:

- SHALL register CIR with recognized registry authority
- SHALL obtain registrar blind data notary signature on CIR
- SHALL publish registrar public key or verification endpoint
- SHALL maintain Notice Event Log accessible via rights_access_point

Blind Data Notary Properties:

- **What registrar signs**: CIR-ID, controller legal entity, jurisdiction, rights access point, publication timestamp
- **What registrar never sees**: pii_principal_id, individual identifiers, receipt content beyond CIR
- **Verification independence**: Third parties validate receipts against public registry without controller cooperation

Registrar Infrastructure:

- Existing registries capable of Level 2: ICO Controller Registry (UK), OAIC (Australia), OPC (Canada)
- Registry coordination protocol: Cross-registry lookup via CIR-ID

Level 3: Enhanced Transparency and Trust Officer (TTAO) Notarization

Description: TTAO certification with physical identity verification beyond registrar baseline.

Requirements:

- Level 2 conformance (registry + blind data notary)
- **Digital Privacy Risk Officer** (NOT Data Protection Risk Officer) undergoes physical identity verification and training
- Digital Privacy Risk Officer physically notarizes ANCR Exchange Stage 3 micro-credentials
- Face-to-face liveliness assurance for credential issuance
- Suitable for: high-risk contexts (cross-border lawful interception, critical infrastructure, vital interest)

Conformance:

- SHALL designate certified DTTO (or equivalent privacy officer role)
- SHALL verify DTTO identity through face-to-face process
- SHALL log DTTO notarization events in Notice Event Log
- SHALL provide DTTO certification reference in CIR

Digital Privacy Risk Officer Certification Requirements:

1. **Physical Identity Verification**: Government-issued ID + face-to-face liveliness check
2. **Training**: Convention 108+ transparency requirements, ANCR Exchange protocol, notarial standards
3. **Competence Assessment**: Demonstration of CIR validation, receipt generation, Notice Event Log maintenance
4. **Ongoing Professional Development**: Annual refresher on updated standards

Digital Privacy Risk Officer-Signed Micro-Credentials:

- Enable real-time lawful access coordination across jurisdictions
- Address cross-border dynamic authorization for lawful interception contexts
- Provide cryptographic proof of physical notarization with face-to-face assurance

Level 4: Active State + Physical Verification

Description: Real-time validation mechanisms with remote attestation for critical infrastructure contexts.

Requirements:

- Level 3 conformance (TTAO certification)
- Real-time active state signaling (HABNI trustmark or equivalent)
- Remote attestation for transparency state monitoring
- Active State Synchronic Validation (CIR hash binding in tokens)
- Suitable for: critical infrastructure, high-risk AI systems, vital interest processing

Conformance:

- SHALL implement Active State Synchronic Validation (Section 17.2.1 of v1.0)
- SHALL provide real-time transparency state API
- SHALL cryptographically bind consent tokens to CIR version hash
- SHALL invalidate tokens automatically when CIR changes
- SHALL log CIR version hash with each token issuance event

Active State Synchronic Validation:

Consent tokens SHALL include cryptographic hash of notice.txt (CIR) at time of issuance:

```
{
  "token_claims": {
    "iss": "CIR-CA-12345",
    "jti": "receipt-abc-123",
    "iat": "2025-12-13T19:00:00-05:00",
    "exp": "2026-12-13T19:00:00-05:00",
    "cir_hash_at_issuance": "sha256:a1b2c3d4e5f6...",
    "cir_publication_url": "https://example.com/.well-known/transparency",
    "cir_version": "v2.3",
    "hash_algorithm": "SHA-256"
  },
  "tata_level": 4,
  "active_state_validation": {
    "enabled": true,
    "validation_endpoint": "https://example.com/api/active-state",
    "last_validation_timestamp": "2025-12-13T19:05:00-05:00"
  }
}
```

Verification Protocol:

1. Extract cir_hash_at_issuance from token
2. Fetch current notice.txt from cir_publication_url
3. Compute hash of current notice.txt using hash_algorithm
4. Compare computed hash with cir_hash_at_issuance
5. If mismatch: Token invalid—CIR updated since issuance
6. Query Notice Event Log to confirm cir_hash_at_issuance matches log entry at token iat timestamp
7. If log mismatch: Token invalid—hash forged or CIR hijacked

Security Benefits:

1. **Prevents Stale Consent Exploits**: Tokens automatically invalidate when CIR changes
2. **Prevents CIR Hijacking**: Token verification fails if CIR hash doesn't match registry-signed version
3. **Prevents Fake Hash Injection**: Token hash must match both current CIR AND Notice Event Log entry
4. **Enables Real-Time Compliance**: Regulators verify token validity without controller cooperation

2.3 digital_transparency_risk_assurance_level Field

Data type: Integer (1-4)

Cardinality: Optional (recommended for registry-verified controllers)

Field Specification:

```
{
  "digital_transparency_risk_assurance_level": 2,
  "assurance_description": "Registry Verification—CIR-ID registered with authoritative source",
  "registry_verification_url": "https://ico.org.uk/controllers/CIR-UK-12345",
  "registrar_signature": {
    "algorithm": "RSA-SHA256",
    "signature_value": "base64_encoded_signature",
    "public_key_url": "https://ico.org.uk/public-keys/registrar-2025.pem",
    "signature_timestamp": "2025-11-15T10:00:00Z"
  }
}
```

2.4 trustmark_certifications Array

Data type: Array of objects

Cardinality: Optional (required for Level 3+)

```
{
  "trustmark_certifications": [
    {
      "trustmark_type": "habni_active_state",
      "certification_body": "Digital Transparency Lab",
      "certification_date": "2025-11-15",
      "verification_url": "https://transparencylab.ca/verify/CIR-CA-12345",
      "expiration_date": "2026-11-15",
      "certification_level": "Level 4 - Active State"
    },
    {
      "trustmark_type": "digital_privacy_risk_officer_certified",
      "certification_body": "International DTTO Authority",
      "certification_date": "2025-10-01",
      "dtto_name": "Jane Smith, CPO",
      "dtto_verification_method": "face_to_face_liveliness",
      "certification_reference": "DTTO-2025-456"
    }
  ]
}
```

2.5 visual_trust_indicator Object

Data type: Object

Cardinality: Optional

```
{
  "visual_trust_indicator": {
    "display_text": "Convention 108+ Compliant | Transparency-by-Default (L2) | ISO/IEC 27560 UNRP v1.0",
    "badge_url": "https://transparencylab.ca/badges/tata-level-2.svg",
    "badge_alt_text": "Digital Transparency Risk Assurance Level 2: Registry Verified",
    "trustmark_html": "<div class='tata-badge' data-level='2'>...</div>",
    "verification_link": "https://transparencylab.ca/verify/CIR-CA-12345"
    }
}
```

2.6 Digital Transparency Risk Assurance Level Conformance Matrix

Requirement	Level 1	Level 2	Level 3	Level 4
CIR Publication	✅ Required	✅ Required	✅ Required	✅ Required
Registry Verification	❌ Optional	✅ Required	✅ Required	✅ Required
Blind Data Notary Signature	❌ Not applicable	✅ Required	✅ Required	✅ Required
Notice Event Log	⚠️ Recommended	✅ Required	✅ Required	✅ Required
Digital Privacy Risk Officer Certification	❌ Not applicable	❌ Not applicable	✅ Required	✅ Required
Physical Notarization	❌ Not applicable	❌ Not applicable	✅ Required (Stage 3)	✅ Required (Stage 3)
Active State Validation	❌ Not applicable	❌ Not applicable	❌ Not applicable	✅ Required
Real-Time Transparency API	❌ Not applicable	❌ Not applicable	⚠️ Recommended	✅ Required
CIR Hash Binding in Tokens	❌ Not applicable	❌ Not applicable	⚠️ Recommended	✅ Required

PART 3: GLOBAL PRIVACY RIGHTS CONTROLS (GPRC)

Section 3: GPRC Framework

3.1 Overview

Purpose: Enable context-aware rights exercise across all transparency contexts through automated APIs and preference signal integration.

Deferred from v1.0 Rationale: GPRC framework addressing 72 transparency contexts (3 vectors × 4 TATA levels × 6 legal bases) extends beyond v1.0's universal notice receipt base.

V1.1 Scope: Context-specific rights mapping, automated rights APIs, preference signal triggering, and regulatory verification protocols.

3.2 72 Digital Transparency Control Contexts

Three Data Control Vectors:

1. **Personal Control (SSI/DID)**: Individual-controlled identifiers and credentials
2. **Data Protection (Federated ID)**: Controller-centric with regulatory oversight
3. **Co-Regulation (MVCR-ANCR)**: Bilateral receipts with public CIR registries

Four TATA Levels: Self-Assertion, Registry Verification, TTAO Notarization, Active State

Six Legal Bases (Convention 108+): Consent, Contract, Legal Obligation, Legitimate Interest, Vital Interest, Public Interest

Result: 3 × 4 × 6 = **72 contexts** for rights exercise

3.3 global_privacy_rights_controls Array

Data type: Array of objects

Cardinality: Optional (recommended for multi-context controllers)

```
{
  "global_privacy_rights_controls": [
    {
      "context_id": "consent_L2_regional",
      "data_control_vector": "co_regulation",
      "tata_level": 2,
      "legal_basis": "consent",
      "scope_of_disclosure": "regional",
      "rights_available": [
        "access",
        "rectification",
        "erasure",
        "portability",
        "withdraw_consent",
        "object"
      ],
      "rights_exercise_mechanisms": {
        "access": {
          "method": "api",
          "endpoint": "https://example.com/api/data-access",
          "authentication_required": false,
          "response_format": "json",
          "max_response_time": "30 days"
        },
        "withdraw_consent": {
          "method": "notice_event_log_update",
          "endpoint": "https://example.com/api/consent-withdrawa
l",
          "authentication_required": true,
          "immediate_effect": true,
          "autonomous": true
        },
        "erasure": {
          "method": "api",
          "endpoint": "https://example.com/api/erasure-request",
          "authentication_required": true,
          "response_format": "json",
          "max_response_time": "30 days"
        }
      },
      "preference_signal_integration": {
        "gpc_triggers_objection": true,
        "gpc_scope": ["legitimate_interest"],
        "automated_withdrawal": true
      }
    }
  ]
}
```

3.4 Context-Specific Rights Exercise

GPRC enables:

1. **Automated Rights APIs**: Machine-readable rights exercise endpoints per context
2. **Preference Signal Integration**: GPC triggers automatic objection in legitimate interest contexts

3. **Notice Receipt Anchored Rights**: Withdraw consent by updating Notice Event Log without authentication
4. **Regulatory Verification**: Supervisory authorities query rights availability across all contexts

Example: GPC-Triggered Objection:

```
{
  "event_time": "2025-12-13T14:00:00-05:00",
  "event_type": "automated_objection_exercised",
  "entity_id": "CIR-CA-12345",
  "context_id": "legitimate_interest_L2_regional",
  "trigger": "gpc_signal_received",
  "rights_exercised": ["object_to_processing"],
  "processing_restricted": ["marketing", "profiling", "third_party_disclosure"],
  "effective_timestamp": "2025-12-13T14:00:00-05:00"
}
```

3.5 Autonomous Consent Withdrawal

GPRC enables human control of privacy matching physical space expectations:

One Command, Multiple Sources:

Individual issues single withdrawal command → automated propagation across all consent contexts with matching scope:

```
{
  "withdrawal_request": {
    "principal_id": "did:example:alice123",
    "withdrawal_scope": "all_marketing_consent",
    "contexts_affected": [
      "consent_L2_regional_controller_A",
      "consent_L2_national_controller_B",
      "consent_L3_international_controller_C"
    ],
    "withdrawal_timestamp": "2025-12-13T15:00:00-05:00",
    "propagation_method": "notice_event_log_update",
    "immediate_effect": true
  }
}
```

Notice Event Log Updates (automatic across all affected controllers):

Each controller's Notice Event Log receives withdrawal event → autonomous enforcement without manual intervention.

PART 4: OAUTH/UMA ROLE MAPPING PATTERNS

Section 4: Authorization Server Integration

4.1 Overview

Purpose: Demonstrate ANCR Exchange Stages 3-4 alignment with existing OAuth 2.0 and UMA 2.0 authorization infrastructure.

Deferred from v1.0 Rationale: Detailed OAuth/UMA patterns require coordination with RFC 6749, RFC 8693, and UMA 2.0 specifications.

V1.1 Scope: Normative role mappings, authorization flow integration, and token exchange patterns.

4.2 OAuth 2.0 / UMA 2.0 Role Mapping

Core Roles:

OAuth/UMA Role	ISO/IEC 29100 Role	ANCR Context	Record Holder
Resource Owner	PII Principal	Individual authorizing access to protected resource	Resource Owner's wallet/storage
Authorization Server	PII Controller (for authorization)	Issues ANCR Exchange Stage 3 micro-credentials; maintains Notice Event Log	Authorization Server storage
Resource Server	PII Controller (for resource)	Holds protected resource; validates ANCR Stage 3 credentials	Resource Server storage

OAuth/UMA Role	ISO/IEC 29100 Role	ANCR Context	Record Holder
Client	Third-Party Controller	Requests access to protected resource on behalf of Resource Owner	Client application storage

4.3 ANCR Exchange Stage 3 as OAuth Access Token

Mapping:

- **ANCR Exchange Stage 3 (Micro Notice Credential)** maps to **OAuth 2.0 Access Token**
- **ANCR Exchange Stage 2 (Consent Notice Receipt)** maps to **OAuth 2.0 Authorization Grant**
- **ANCR Exchange Stage 1 (Notice Receipt)** maps to **OAuth 2.0 Client Registration** (CIR as client metadata)

Token Structure:

```
{
  "access_token": "eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9...",
  "token_type": "Bearer",
  "expires_in": 3600,
  "scope": "read write",
  "ancr_binding": {
    "receipt_id": "receipt-abc-123",
    "controller_identity_record_id": "CIR-CA-12345",
    "notice_receipt_type": "ANCR Exchange Stage 3",
    "permissions_bundle": ["read", "write"],
    "authorization_timestamp": "2025-12-13T10:00:00-05:00",
    "notice_event_log_url": "https://example.com/api/notice-event
-log"
  }
}
```

4.4 Authorization Code Flow with ANCR

Step 1: Authorization Request (ANCR Exchange Stage 1)

Client redirects Resource Owner to Authorization Server:

```
GET /authorize?response_type=code
  &client_id=CIR-CLIENT-789
  &redirect_uri=https://client.example.com/callback
  &scope=read+write
  &state=xyz
  &ancr_stage_1_url=https://authserver.com/.well-known/transparency
```

Authorization Server presents CIR (Stage 1) → Resource Owner reviews controller identity before consent.

Step 2: Authorization Grant (ANCR Exchange Stage 2)

Resource Owner authorizes → Authorization Server generates ANCR Exchange Stage 2 (Consent Notice Receipt) with authorization_type="consent_granted".

Authorization Server returns authorization code:

```
HTTP/1.1 302 Found
Location: https://client.example.com/callback?code=AUTH_CODE_HERE&state=xyz
```

Step 3: Access Token Request (ANCR Exchange Stage 3)

Client exchanges authorization code for access token:

```
POST /token
Content-Type: application/x-www-form-urlencoded

grant_type=authorization_code
  &code=AUTH_CODE_HERE
  &redirect_uri=https://client.example.com/callback
  &client_id=CIR-CLIENT-789
  &client_secret=SECRET
```

Authorization Server issues ANCR Exchange Stage 3 (Micro Notice Credential) as access token:

```
{
  "access_token": "eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9...",
  "token_type": "Bearer",
  "expires_in": 3600,
  "refresh_token": "REFRESH_TOKEN_HERE",
  "scope": "read write",
  "ancr_binding": {
    "receipt_id": "receipt-abc-123",
    "controller_identity_record_id": "CIR-CA-12345",
    "notice_receipt_type": "ANCR Exchange Stage 3",
    "cryptographic_signature": "base64_signature",
    "credential_binding": "client_id=CIR-CLIENT-789",
    "technical_permissions_scope": ["https://api.example.com/resource/*"],
    "validity_period": "P1H"
  }
}
```

Step 4: Resource Access with ANCR Credential

Client accesses protected resource with ANCR Stage 3 credential:

```
GET /resource/123
Authorization: Bearer eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9...
```

Resource Server validates token:

1. Verify signature against Authorization Server public key (from CIR registry)
2. Check token expiration
3. Validate scope matches requested resource
4. Optional: Query Notice Event Log for revocation status

4.5 UMA 2.0 Permission Ticket Flow with ANCR

UMA Resource Registration:

Resource Server registers protected resource with Authorization Server, including CIR reference:

```
{
  "resource_scopes": ["read", "write"],
  "description": "Personal health records",
  "icon_uri": "https://example.com/icons/health-record.png",
  "name": "Health Record API",
  "type": "http://example.com/rset/health-record",
  "ancr_metadata": {
    "controller_identity_record_id": "CIR-CA-HEALTH-456",
    "scope_of_disclosure": "regional",
    "pii_categories": ["health", "biometric"],
    "legal_basis": "consent",
    "notice_receipt_required": true
  }
}
```

Permission Ticket Issuance:

Resource Server denies Client access → returns permission ticket:

```
HTTP/1.1 401 Unauthorized
WWW-Authenticate: UMA realm="example",
  as_uri="https://authserver.com",
  ticket="PERMISSION_TICKET_HERE",
  ancr_stage_1_url="https://authserver.com/.well-known/transparency"
```

Client retrieves CIR (ANCR Exchange Stage 1) from ancr_stage_1_url → presents to Resource Owner for authorization.

Claims Gathering with ANCR Exchange Stage 2:

Authorization Server requests claims from Resource Owner → generates ANCR Exchange Stage 2 (Consent Notice Receipt) with permissions_bundle:

```
{
  "authorization_type": "consent_granted",
  "consent_timestamp": "2025-12-13T11:00:00-05:00",
  "permissions_bundle": ["read_health_records", "write_health_records"],
  "pii_categories": ["health", "biometric"],
  "scope_of_disclosure": "regional",
  "retention_period": "P30D",
  "pii_principal_id": "patient-12345"
}
```

RPT Token Issuance (ANCR Exchange Stage 3):

Authorization Server issues Requesting Party Token (RPT) as ANCR Exchange Stage 3 credential:

```
{
  "access_token": "RPT_TOKEN_HERE",
  "token_type": "Bearer",
  "expires_in": 3600,
  "upgraded": true,
  "ancr_binding": {
    "receipt_id": "receipt-health-789",
    "controller_identity_record_id": "CIR-CA-HEALTH-456",
    "notice_receipt_type": "ANCR Exchange Stage 3",
    "permissions_bundle": ["read_health_records", "write_health_records"],
    "uma_permission_ticket": "PERMISSION_TICKET_HERE",
    "resource_id": "health-record-123",
    "cryptographic_signature": "base64_signature"
  }
}
```

4.6 Token Introspection with Notice Event Log

Standard OAuth 2.0 Token Introspection (RFC 7662):

```
POST /introspect
Authorization: Basic base64(client_id:client_secret)
Content-Type: application/x-www-form-urlencoded

token=eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9...
```

ANCR-Enhanced Introspection Response:

```
{
  "active": true,
  "scope": "read write",
  "client_id": "CIR-CLIENT-789",
  "token_type": "Bearer",
  "exp": 1734123600,
  "iat": 1734120000,
  "sub": "patient-12345",
  "aud": "https://api.example.com",
  "iss": "https://authserver.com",
  "ancr_metadata": {
    "receipt_id": "receipt-abc-123",
    "controller_identity_record_id": "CIR-CA-12345",
    "notice_receipt_type": "ANCR Exchange Stage 3",
    "authorization_type": "consent_granted",
    "consent_timestamp": "2025-12-13T10:00:00-05:00",
    "permissions_bundle": ["read", "write"],
    "notice_event_log_url": "https://authserver.com/api/notice-event-log",
    "revocation_status": "valid",
    "last_notice_event_log_check": "2025-12-13T15:00:00-05:00"
  }
}
```

Revocation Check via Notice Event Log:

Resource Server queries Notice Event Log for authorization status:

```
GET /api/notice-event-log?receipt_id=receipt-abc-123&event_type=consent_withdrawn
```

If consent withdrawn:

```
{
  "events": [
    {
      "event_time": "2025-12-13T14:30:00-05:00",
      "event_type": "consent_withdrawn",
      "entity_id": "CIR-CA-12345",
      "receipt_id": "receipt-abc-123",
      "withdrawal_reason": "user_request",
      "effective_immediately": true
    }
  ]
}
```

Resource Server invalidates token immediately—autonomous consent withdrawal without authorization server interaction.

4.7 Token Exchange (RFC 8693) with ANCR Portability

ANCR Exchange Stage 4 as Token Exchange:

Resource Owner exchanges ANCR Stage 3 credential from Controller A for portable Stage 4 token usable with Controller B:

```
POST /token
Content-Type: application/x-www-form-urlencoded

grant_type=urn:ietf:params:oauth:grant-type:token-exchange
  &subject_token=ANCR_STAGE_3_TOKEN_FROM_CONTROLLER_A
  &subject_token_type=urn:ietf:params:oauth:token-type:access_tok
en
  &requested_token_type=urn:ietf:params:oauth:token-type:jwt
  &audience=https://controller-b.example.com
  &scope=read+write
  &ancr_portability_scope=international
```

Authorization Server Response (ANCR Exchange Stage 4):

```
{
  "access_token": "eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9...",
  "issued_token_type": "urn:ietf:params:oauth:token-type:jwt",
  "token_type": "Bearer",
  "expires_in": 3600,
  "scope": "read write",
  "ancr_binding": {
    "receipt_id": "receipt-portable-456",
    "notice_receipt_type": "ANCR Exchange Stage 4",
    "principal_anchor": "did:example:alice123",
    "token_claims": {
      "iss": "CIR-CA-12345",
      "sub": "did:example:alice123",
      "aud": "https://controller-b.example.com",
      "exp": 1734123600,
      "iat": 1734120000,
      "jti": "receipt-portable-456"
    },
    "cross_controller_metadata": {
      "original_controller_id": "CIR-CA-12345",
      "original_receipt_id": "receipt-abc-123",
      "provenance_chain": ["CIR-CA-12345", "CIR-CA-AUTH-SERVER"],
      "authorization_timestamp": "2025-12-13T10:00:00-05:00"
    },
    "portability_scope": "international",
    "permissions_bundle": ["read", "write"]
  }
}
```

Controller B Verification:

1. Extract original_controller_id from cross_controller_metadata
2. Lookup CIR-CA-12345 in public registry
3. Validate signature against Controller A's public key
4. Check Notice Event Log at CIR-CA-12345 for authorization status
5. Verify portability_scope includes "international"
6. Honor authorization without re-prompting Resource Owner

PART 5: CONSENT TOKEN SPECIFICATIONS

Section 5: Cryptographic Binding and Verification

5.1 Overview

Purpose: Normative specifications for consent token cryptographic binding, verification protocols, and revocation mechanisms.

Deferred from v1.0 Rationale: Portable authorization tokens address advanced scenarios beyond v1.0's bilateral receipt generation focus.

V1.1 Scope: JWT/CBOR/JSON-LD formats, proof-of-possession patterns, ZKP integration, and W3C Verifiable Credentials alignment.

5.2 Token Format Specifications

5.2.1 JWT Format (RECOMMENDED)

Header:

```
{
  "alg": "RS256",
  "typ": "JWT",
  "kid": "CIR-CA-12345-key-2025-12"
}
```

Payload (maps to ANCR Exchange Stage 4 fields):

```
{
  "iss": "CIR-CA-12345",
  "sub": "did:example:alice123",
  "aud": "https://controller-b.example.com",
  "exp": 1734123600,
  "iat": 1734120000,
  "nbf": 1734120000,
  "jti": "receipt-portable-456",
  "authorization": {
    "type": "consent_granted",
    "permissions_bundle": ["read", "write"],
    "purpose": "secondary_purpose_data_sharing",
    "scope_of_disclosure": "international",
    "pii_categories": ["email", "name", "health"],
    "retention_period": "P30D"
  },
  "principal_anchor": "did:example:alice123",
  "cross_controller_metadata": {
    "original_controller_id": "CIR-CA-12345",
    "original_receipt_id": "receipt-abc-123",
    "provenance_chain": ["CIR-CA-12345"],
    "authorization_timestamp": "2025-12-13T10:00:00-05:00"
  },
  "portability_scope": "international",
  "notice_event_log_id": "https://example.com/api/notice-event-lo
g"
}
```

Signature: RS256 using controller's private key (public key published in CIR registry)

5.2.2 CBOR Format (for IoT/Constrained Environments)

Use case: IoT devices, mobile apps with size constraints

CBOR Encoding:

```
A7                                        # map(7)
   01                                     # unsigned(1) - iss
   6F CIR-CA-12345                        # text(11) "CIR-CA-12345"
   02                                     # unsigned(2) - sub
   76 did:example:alice123                # text(22) "did:example:a
lice123"
   ...
```

Benefits: ~40% size reduction compared to JSON, faster parsing on constrained devices

5.2.3 JSON-LD Format (for Semantic Web Integration)

Use case: Semantic web applications, W3C Verifiable Credentials integration

```
{
  "@context": [
    "https://www.w3.org/2018/credentials/v1",
    "https://transparencylab.ca/contexts/ancr/v1"
  ],
  "type": ["VerifiableCredential", "ANCRConsentToken"],
  "issuer": "did:web:example.com:controller:CIR-CA-12345",
  "issuanceDate": "2025-12-13T10:00:00Z",
  "expirationDate": "2026-12-13T10:00:00Z",
  "credentialSubject": {
    "id": "did:example:alice123",
    "authorization": {
      "type": "ConsentGranted",
      "permissionsBundle": ["read", "write"],
      "purpose": "secondary_purpose_data_sharing",
      "scopeOfDisclosure": "international",
      "piiCategories": ["email", "name", "health"]
    }
  },
  "proof": {
    "type": "RsaSignature2018",
    "created": "2025-12-13T10:00:00Z",
    "proofPurpose": "assertionMethod",
    "verificationMethod": "did:web:example.com:controller:CIR-CA-12345#key-1",
    "jws": "eyJhbGciOiJSUzI1NiIsImI2NCI6ZmFsc2UsImNyaXQiOlsiYjY0Il19..."
  }
}
```

5.3 Proof-of-Possession (PoP) Patterns

5.3.1 Bearer Token (Default)

Use case: Trusted agent scenarios, low-risk PII categories

Verification: Token holder can exercise authorization without additional proof

Risk: Token theft enables unauthorized access

5.3.2 PoP with DPoP (OAuth 2.0 Demonstration of Proof-of-Possession)

Use case: High-risk PII categories, cross-border transfers

Token Request:

```
POST /token
DPoP: eyJhbGciOiJSUzI1NiIsInR5cCI6ImRwb3Arand0IiwiandrIjp7Imt0eSI
6IlJTQSIsIm4iOiIuLi4ifX0...
Content-Type: application/x-www-form-urlencoded

grant_type=authorization_code&code=AUTH_CODE&...
```

Token Response:

```
{
  "access_token": "eyJhbGciOiJSUzI1NiIsInR5cCI6ImF0K2p3dCJ9...",
  "token_type": "DPoP",
  "expires_in": 3600,
  "ancr_binding": {...}
}
```

Resource Access with PoP Proof:

```
GET /resource/123
Authorization: DPoP eyJhbGciOiJSUzI1NiIsInR5cCI6ImF0K2p3dCJ9...
DPoP: eyJhbGciOiJSUzI1NiIsInR5cCI6ImRwb3Arand0IiwiandrIjp7Imt0eSI
6IlJTQSIsIm4iOiIuLi4ifX0...
```

Resource Server validates:

1. Access token signature
2. DPoP proof signature matches public key in token
3. DPoP proof "htu" (HTTP URI) matches request URI
4. DPoP proof "htm" (HTTP method) matches request method
5. Token not replayed (check "jti")

5.3.3 PoP with Challenge-Response

Use case: Vital interest processing, critical infrastructure

Verification Flow:

1. Client presents token to Resource Server

2. Resource Server generates challenge (random nonce)
3. Client signs challenge with private key bound to principal_anchor
4. Resource Server validates signature against public key in principal_anchor (DID document)
5. Authorization granted only if signature valid

Challenge-Response Protocol:

```
GET /resource/123
Authorization: Bearer eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9...

HTTP/1.1 401 Unauthorized
WWW-Authenticate: PoP realm="example", challenge="base64_random_nonce"

GET /resource/123
Authorization: Bearer eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9...
PoP-Signature: signature_of_challenge_with_private_key

HTTP/1.1 200 OK
...
```

5.4 Zero-Knowledge Proof Patterns

5.4.1 Selective Disclosure with BBS+ Signatures

Use case: Prove authorization scope without revealing full permissions_bundle

Example: Prove "age > 18" permission without revealing exact birthdate or other PII categories

Credential Issuance (Controller creates BBS+ signed credential):

```
{
  "@context": [
    "https://www.w3.org/2018/credentials/v1",
    "https://transparencylab.ca/contexts/ancr/v1"
  ],
  "type": ["VerifiableCredential", "ANCRConsentToken"],
  "issuer": "did:web:example.com:controller:CIR-CA-12345",
  "credentialSubject": {
    "id": "did:example:alice123",
    "permissions": {
      "ageVerification": true,
      "age": 25,
      "email": "alice@example.com",
      "healthData": "sensitive_health_info"
    }
  },
  "proof": {
    "type": "BbsBlsSignature2020",
    "created": "2025-12-13T10:00:00Z",
    "proofPurpose": "assertionMethod",
    "verificationMethod": "did:web:example.com:controller:CIR-CA-
12345#bbs-key-1",
    "proofValue": "base64_bbs_signature"
  }
}
```

Selective Disclosure Presentation (Individual proves age > 18 without revealing age, email, health data):

```
{
  "@context": [
    "https://www.w3.org/2018/credentials/v1",
    "https://transparencylab.ca/contexts/ancr/v1"
  ],
  "type": "VerifiablePresentation",
  "verifiableCredential": {
    "@context": "...",
    "type": ["VerifiableCredential", "ANCRConsentToken"],
    "issuer": "did:web:example.com:controller:CIR-CA-12345",
    "credentialSubject": {
      "id": "did:example:alice123",
      "permissions": {
        "ageVerification": true
      }
    },
    "proof": {
      "type": "BbsBlsSignatureProof2020",
      "created": "2025-12-13T10:00:00Z",
      "proofPurpose": "assertionMethod",
      "verificationMethod": "did:web:example.com:controller:CIR-CA-12345#bbs-key-1",
      "proofValue": "base64_derived_proof",
      "revealedAttributes": ["permissions.ageVerification"]
    }
  },
  "proof": {
    "type": "Ed25519Signature2020",
    "created": "2025-12-13T15:00:00Z",
    "proofPurpose": "authentication",
    "verificationMethod": "did:example:alice123#key-1",
    "challenge": "verifier_provided_challenge",
    "proofValue": "base64_presentation_signature"
  }
}
```

Verifier confirms:

- Credential issued by trusted Controller (CIR-CA-12345)
- BBS+ proof validates against Controller's public key
- ageVerification attribute revealed without exposing age, email, or health data

5.5 Revocation Mechanisms

5.5.1 Notice Event Log-Based Revocation (RECOMMENDED)

Mechanism: Controllers query Notice Event Log before honoring token

Revocation Flow:

1. Individual withdraws consent → update Notice Event Log with event_type="consent_withdrawn"
2. Token verification includes Notice Event Log query
3. If consent_withdrawn event found with matching receipt_id → token invalid

4. Revocation effective immediately without distributing revocation lists

Revocation Event:

```
{
  "event_time": "2025-12-13T16:00:00-05:00",
  "event_type": "consent_withdrawn",
  "entity_id": "CIR-CA-12345",
  "receipt_id": "receipt-abc-123",
  "withdrawal_reason": "user_request",
  "scope_affected": "all_permissions",
  "effective_immediately": true,
  "notification_sent": true
}
```

Token Verification with Revocation Check:

```
GET /api/notice-event-log?receipt_id=receipt-abc-123&event_type=consent_withdrawn&after_timestamp=2025-12-13T10:00:00-05:00
```

If events returned → token revoked

5.5.2 Revocation List (for Offline Verification)

Use case: Constrained environments without real-time Notice Event Log access

Revocation List Format:

```
{
  "issuer": "CIR-CA-12345",
  "list_timestamp": "2025-12-13T17:00:00-05:00",
  "next_update": "2025-12-13T18:00:00-05:00",
  "revoked_tokens": [
    {
      "receipt_id": "receipt-abc-123",
      "revocation_timestamp": "2025-12-13T16:00:00-05:00",
      "reason": "user_withdrawal"
    },
    {
      "receipt_id": "receipt-def-456",
      "revocation_timestamp": "2025-12-13T15:30:00-05:00",
      "reason": "material_change"
    }
  ],
  "signature": "base64_signature_over_list"
}
```

Verifiers download revocation list periodically → check token receipt_id against list.

5.5.3 Status List 2021 (W3C Standard)

Use case: W3C Verifiable Credentials integration

Status List Credential:

```
{
  "@context": [
    "https://www.w3.org/2018/credentials/v1",
    "https://w3id.org/vc/status-list/2021/v1"
  ],
  "id": "https://example.com/credentials/status/3",
  "type": ["VerifiableCredential", "StatusList2021Credential"],
  "issuer": "did:web:example.com:controller:CIR-CA-12345",
  "issued": "2025-12-13T17:00:00Z",
  "credentialSubject": {
    "id": "https://example.com/credentials/status/3#list",
    "type": "StatusList2021",
    "statusPurpose": "revocation",
    "encodedList": "H4sIAAAAAAAA..." // compressed bitstring
  },
  "proof": {...}
}
```

Token with Status List Reference:

```
{
  "credentialStatus": {
    "id": "https://example.com/credentials/status/3#94567",
    "type": "StatusList2021Entry",
    "statusPurpose": "revocation",
    "statusListIndex": "94567",
    "statusListCredential": "https://example.com/credentials/stat
us/3"
  }
}
```

Verifier:

1. Fetches Status List Credential
2. Decompresses encodedList bitstring
3. Checks bit at statusListIndex (94567)
4. If bit = 1 → revoked; if bit = 0 → valid

5.6 W3C Verifiable Credentials Alignment

ANCR Exchange Stage 4 as Verifiable Credential:

```
{
  "@context": [
    "https://www.w3.org/2018/credentials/v1",
    "https://transparencylab.ca/contexts/ancr/v1"
  ],
  "id": "https://example.com/credentials/consent/receipt-abc-12
3",
  "type": ["VerifiableCredential", "ANCRConsentToken"],
  "issuer": {
    "id": "did:web:example.com:controller:CIR-CA-12345",
    "name": "Example Health Services",
    "jurisdiction": "CA-ON"
  },
  "issuanceDate": "2025-12-13T10:00:00Z",
  "expirationDate": "2026-12-13T10:00:00Z",
  "credentialSubject": {
    "id": "did:example:alice123",
    "authorization": {
      "type": "ConsentGranted",
      "permissionsBundle": ["read_health_records", "write_health_
records"],
      "purpose": "healthcare_service_delivery",
      "scopeOfDisclosure": "regional",
      "piiCategories": ["health", "biometric"],
      "retentionPeriod": "P30D"
    },
    "crossControllerMetadata": {
      "originalControllerId": "CIR-CA-12345",
      "originalReceiptId": "receipt-abc-123",
      "provenanceChain": ["CIR-CA-12345"],
      "authorizationTimestamp": "2025-12-13T10:00:00Z"
    },
    "portabilityScope": "regional"
  },
  "credentialStatus": {
    "id": "https://example.com/api/notice-event-log?receipt_id=receipt-abc-123",
    "type": "NoticeEventLogStatus2025"
  },
  "proof": {
    "type": "RsaSignature2018",
    "created": "2025-12-13T10:00:00Z",
    "proofPurpose": "assertionMethod",
    "verificationMethod": "did:web:example.com:controller:CIR-CA-12345#key-1",
    "jws": "eyJhbGciOiJSUzI1NiIsImI2NCI6ZmFsc2UsImNyaXQiOlsiYjY0Il19..."
  }
}
```

Verification via Verifiable Presentation:

```
{
  "@context": [
    "https://www.w3.org/2018/credentials/v1"
  ],
  "type": "VerifiablePresentation",
  "verifiableCredential": ["...ANCRConsentToken from above..."],
  "proof": {
    "type": "Ed25519Signature2020",
    "created": "2025-12-13T15:00:00Z",
    "proofPurpose": "authentication",
    "verificationMethod": "did:example:alice123#key-1",
    "challenge": "verifier_provided_challenge_nonce",
    "domain": "controller-b.example.com",
    "proofValue": "z3FXQsWzx..." // signature by Alice proving co
ntrol of did:example:alice123
  }
}
```

Controller B:

1. Validates ANCRConsentToken signature against CIR-CA-12345 public key
2. Validates Verifiable Presentation signature against did:example:alice123 public key
3. Checks credentialStatus via Notice Event Log
4. Verifies portabilityScope permits cross-controller sharing
5. Honors authorization without re-prompting Alice

PART 6: AI SYSTEM TRANSPARENCY

Section 6: ISO/IEC 42001 Coordination

6.1 Overview

Purpose: Normative AI system transparency requirements coordinated with ISO/IEC 42001 and EU AI Act.

Deferred from v1.0 Rationale: AI governance infrastructure extends beyond v1.0's core co-regulated transparency architecture.

V1.1 Scope: Mandatory AI notice event log requirements, high-risk AI enhanced transparency, cross-border AI model deployment standards, ISO/IEC 27091 coordination.

6.2 Mandatory AI Notice Event Log Requirements

When context_category="ai_system", controllers SHALL log:

Event Type	Trigger	ISO/IEC 42001 Ref	Mandatory Fields
ai_decision_made	AI system makes or influences decision affecting individual	B.6.2.8	decision_id, timestamp, outcome, confidence_score, model_version
ai_model_retrained	Model retrained or version updated	B.6.2.6	old_version, new_version, performance_metrics, training_data_summary
human_override_exercised	Human reviewer overrides AI decision	B.9.3	override_timestamp, reviewer_id, override_reason, final_outcome
explainability_requested	Individual requests decision explanation	B.8.2	request_timestamp, decision_id, explanation_delivery_method
ai_performance_drift_detected	Model performance below threshold or bias detected	B.6.2.6	metric_name, expected_value, actual_value, detection_timestamp
ai_bias_mitigation_applied	Bias mitigation measures implemented	B.6.2.6	mitigation_type, affected_attributes, effectiveness_metric

6.3 High-Risk AI Enhanced Transparency

Controllers deploying **high-risk AI systems** (EU AI Act Annex III, ISO/IEC 42001 risk classification) SHALL:

1. Extend CIR with ai_system_documentation object (see Appendix J.4.1 from v1.0)
2. Reference conformity assessment body and CE marking (if applicable)
3. Document human oversight mechanisms in permissions_bundle
4. Provide explainability mechanism via rights_access_point

5. Log all mandatory AI events per Section 6.2
6. Update notice receipts when AI model version changes materially

Material Change Threshold: Model retraining triggers material change notification if:

- Performance metrics change by >5% (accuracy, precision, recall, F1)
- Fairness metrics change by >2% (demographic parity, equalized odds)
- Training data sources change significantly (>20% new data)
- Algorithm type changes

6.4 Cross-Border AI Model Deployment Transparency

When AI system training, deployment, or inference occurs across jurisdictions:

recipient_jurisdictions Enhancement:

```
{
  "scope_of_disclosure": "international",
  "recipient_jurisdictions": [
    {
      "country_code": "US",
      "role": "ai_model_training",
      "adequacy_status": "no_decision",
      "data_processed": "Training dataset—anonymized user interactions 2020-2024",
      "legal_regime": "FISA Section 702 permits government access to training data stored by US cloud providers",
      "ai_specific_risks": [
        "Model parameters may reveal training data characteristics if subject to government access",
        "Differential privacy applied (epsilon=1.0) to mitigate reconstruction risks"
      ]
    },
    {
      "country_code": "CA",
      "role": "ai_inference",
      "adequacy_status": "adequate",
      "data_processed": "Real-time inference—individual queries p
rocessed in Canadian data centers",
      "ai_specific_risks": []
    }
  ]
}
```

Federated Learning Disclosure:

```
{
  "ai_training_architecture": "federated_learning",
  "federated_learning_disclosure": {
    "description": "Model trained across multiple jurisdictions w
ithout centralizing raw data",
    "participant_jurisdictions": ["CA", "US", "GB"],
    "data_transfer_minimal": true,
    "gradient_transfer_only": "Only model parameter updates excha
nged; raw data remains in origin jurisdiction",
    "privacy_guarantees": [
      "Secure aggregation prevents individual contribution infere
nce",
      "Differential privacy applied to gradient updates (epsilon=
0.5)",
      "No raw personal data leaves origin jurisdiction"
    ]
  }
}
```

6.5 ISO/IEC 27091 (AI Privacy) Coordination

Division of Responsibility:

- **ISO/IEC 27091**: Defines AI-specific privacy requirements (data quality, fairness, transparency obligations)
- **Universal Notice Receipt Profile v1.1**: Provides bilateral receipt exchange mechanism for 27091 transparency obligations

Cross-Reference:

Organizations implementing ISO/IEC 27091 SHALL use Universal Notice Receipt Profile v1.1 AI System Transparency Extension for:

- Notice receipt generation with context_category="ai_system"
- AI-specific Notice Event Log maintenance per Section 6.2
- Automated decision consent (authorization_type="automated_decision_consent")
- AI training data consent (authorization_type="ai_training_data_consent")

PART 7: IMPLEMENTATION TIMELINE & CONFORMANCE

Section 7: V1.1 Development Phases

Phase 1 (Q1 2026): Privacy Preference Signal Pilot

Deliverables:

- GPC scope mapping specification finalized
- Notice Event Log preference event types standardized
- Pilot implementations with 5+ controllers
- Integration testing with W3C GPC specification

Success Criteria:

- GPC signal honored within 48 hours
- Automated permissions restriction functional
- Notice Event Log captures preference signals

Phase 2 (Q2 2026): TATA Level 3 Protocols

Deliverables:

- DTTO certification program launched
- Blind data notary signature standards finalized
- Cross-registry coordination protocols established
- Trustmark visual standards published

Success Criteria:

- 3+ registries providing blind data notary signatures
- 10+ certified DTTOs across jurisdictions
- Registrar public key infrastructure operational

Phase 3 (Q3 2026): GPRC Framework + Automated Rights APIs

Deliverables:

- Context-specific rights mapping across 72 contexts
- Automated rights exercise API specification
- Preference signal rights triggering protocols
- Regulatory verification API standards

Success Criteria:

- Autonomous consent withdrawal functional
- GPC-triggered objection automated
- Multi-controller withdrawal propagation working

Phase 4 (Q4 2026): Full v1.1 Integration

Deliverables:

- Complete v1.1 specification published
- All enhanced features integrated
- Conformance testing framework operational
- V1.1 certification program launched

Success Criteria:

- 20+ controllers conformant to v1.1
- Interoperability testing successful
- Regulatory adoption in 2+ jurisdictions

Section 8: V1.1 Conformance Levels

Enhanced Privacy Preference Conformance

Controllers claiming Enhanced Privacy Preference Conformance SHALL:

1. Implement v1.0 Universal Notice Receipt Profile (mandatory base)
2. Honor GPC signal within 48 hours of receipt
3. Log preference signals in Notice Event Log
4. Update permissions_bundle to reflect restricted processing
5. Provide preference_center_url for manual management

TATA Level 3+ Conformance

Controllers claiming TATA Level 3 or 4 Conformance SHALL:

1. Achieve Level 2 conformance (registry verification + blind data notary)
2. Designate certified DTTO (Level 3) or provide active state API (Level 4)
3. Log DTTO notarization events or CIR hash validation events
4. Implement cryptographic binding for Stage 3/4 tokens

GPRC Conformance

Controllers claiming GPRC Conformance SHALL:

1. Document rights available across all applicable contexts
2. Provide automated rights exercise APIs
3. Enable GPC-triggered objection for legitimate interest processing
4. Support autonomous consent withdrawal via Notice Event Log

AI System Transparency Extension Conformance

Controllers claiming AI System Transparency Extension Conformance SHALL:

1. Implement v1.0 Universal Notice Receipt Profile (mandatory base)
2. Use context_category="ai_system" when AI systems process personal data
3. Log all mandatory AI events per Section 6.2
4. Provide human oversight mechanism disclosure for automated decisions
5. Disclose cross-border AI model deployment when scope_of_disclosure="international"

APPENDICES

Appendix K: Private-Personal-AI Processing Method

K.1 Overview

Purpose: Define processing method where notice tokens authorize AI training exclusively for individual's local model, enabling privacy-enabling personal AI without surveillance risk.

Core Principle: Individual acts as both PII Principal AND AI Model Controller. Personal data trains only the individual's own model—never disclosed to third parties or used for shared model training.

K.2 Private-Personal-AI Definition

Private-Personal-AI is a processing method where:

1. **Individual controls the model**: PII Principal owns and operates the AI model
2. **Local processing only**: Training and inference occur on individual's devices or individual-controlled infrastructure
3. **No third-party disclosure**: Training data never leaves individual's control
4. **Provenance proof**: Notice token demonstrates data authorization for personal use only

Contrast with Other Processing Methods:

Processing Method	Model Controller	Training Data Location	Scope of Disclosure	Primary Use Case
Private-Personal-AI	Individual (PII Principal)	Individual's devices only	individual_only	Personal assistant, health tracking, private knowledge base
Shared Model Training	Service Provider	Centralized cloud infrastructure	regional/international	Commercial AI services (ChatGPT, Gemini)
Federated Learning	Coordinating Controller	Distributed (gradients only)	community/regional	Healthcare research, mobile

Processing Method	Model Controller	Training Data Location	Scope of Disclosure	Primary Use Case
				keyboard predictions
Aggregate Analytics	Research Institution	Centralized with differential privacy	national/international	Public health research, census analytics

K.3 Notice Token Structure for Private-Personal-AI

Field Extensions:

```
{
  "notice_token_id": "urn:token:private-ai-123",
  "controller_identity_record_id": "CIR-INDIVIDUAL-ALICE",
  "secondary_purpose": "private_personal_ai_training",
  "processing_method": "private_personal_ai_only",
  "processing_constraints": {
    "scope_of_disclosure": "individual_only",
    "model_deployment": "local_device",
    "training_data_retention": "P90D",
    "third_party_access": "prohibited",
    "model_sharing": "prohibited",
    "gradient_sharing": "prohibited"
  },
  "data_provenance": {
    "source_controller": "CIR-CA-HEALTH-PROVIDER",
    "source_authorization_type": "consent",
    "secondary_purpose_authorization": "explicit_consent",
    "authorized_for_shared_training": false,
    "authorized_for_personal_ai": true
  },
  "model_metadata": {
    "model_type": "personal_llm",
    "model_location": "device_local_storage",
    "inference_location": "on_device",
    "no_cloud_sync": true
  }
}
```

K.4 Processing Method Vocabulary

Normative Values for `processing_method` Field:

- `private_personal_ai_only`: Data trains individual's local model exclusively (this appendix)
- `shared_model_training`: Data contributes to centralized shared model
- `federated_learning`: Distributed training with gradient aggregation (no raw data centralization)
- `aggregate_analytics_only`: Statistical analysis with differential privacy (no individual model training)
- `hybrid_federated_personal`: Combination of federated and personal models

K.5 Data Provenance for Secondary Purpose

Scenario: Individual receives health data from provider, requests authorization to use for personal AI training.

Step 1: Primary Collection (Provider as Controller)

```
{
  "controller_identity_record_id": "CIR-CA-HEALTH-PROVIDER",
  "authorization_type": "consent",
  "consent_type": "informed",
  "purposes": ["healthcare_delivery"],
  "pii_categories": ["health", "biometric", "diagnostic_results"],
  "scope_of_disclosure": "regional"
}
```

Step 2: Secondary Purpose Request (Individual requests personal AI authorization)

```
{
  "secondary_purpose_request": {
    "requested_by": "pii_principal",
    "requested_purpose": "private_personal_ai_training",
    "processing_method": "private_personal_ai_only",
    "justification": "Individual wants to train personal health assistant on own diagnostic history",
    "no_third_party_disclosure": true,
    "provenance_chain": ["CIR-CA-HEALTH-PROVIDER"]
  }
}
```

Step 3: Provider Authorization (Provider grants secondary purpose)

```
{
  "authorization_type": "secondary_purpose_consent",
  "secondary_purpose": "private_personal_ai_training",
  "processing_method": "private_personal_ai_only",
  "processing_constraints_mandatory": {
    "scope_of_disclosure": "individual_only",
    "no_commercial_use": true,
    "no_model_sharing": true,
    "retention_period": "P2Y"
  },
  "provenance_attestation": {
    "source_controller": "CIR-CA-HEALTH-PROVIDER",
    "attestation": "Data authorized exclusively for individual's personal AI training. Third-party disclosure or shared model training prohibited.",
    "attestation_signature": "base64_signature"
  }
}
```

Step 4: Notice Token Issuance (Individual as Model Controller)

```
{
  "notice_token_id": "urn:token:health-personal-ai-789",
  "controller_identity_record_id": "CIR-INDIVIDUAL-ALICE",
  "token_type": "private_personal_ai",
  "processing_method": "private_personal_ai_only",
  "provenance_chain": [
    {
      "source_controller": "CIR-CA-HEALTH-PROVIDER",
      "authorization_type": "secondary_purpose_consent",
      "authorized_date": "2025-12-15T10:00:00-05:00",
      "constraints_binding": true
    }
  ]
}
```

K.6 Provenance Verification Protocol

Third parties verifying provenance (e.g., regulators, auditors):

1. **Extract provenance chain** from notice token
2. **Query source controller CIR** (CIR-CA-HEALTH-PROVIDER)
3. **Verify secondary purpose authorization** in source Notice Event Log
4. **Confirm processing constraints**:
    - `scope_of_disclosure: "individual_only"`
    - `processing_method: "private_personal_ai_only"`
    - No shared training authorization
5. **Validate cryptographic signatures** across provenance chain

Verification Outcome:

- ✅ **Valid**: Data authorized for personal AI, no surveillance risk
- ❌ **Invalid**: Processing method doesn't match token, data disclosed beyond individual

K.7 Regulatory Compliance Benefits

Convention 108+ Article 5 (Purpose Limitation):

- Secondary purpose clearly specified and limited to individual's personal use
- Processing constraints prevent purpose creep

GDPR Article 6(4) (Compatible Secondary Purpose):

- Personal AI training is compatible with original healthcare purpose when:

    - Individual controls both primary and secondary processing
    - No third-party disclosure occurs
    - Processing constraints honored

PIPEDA Principle 4.3 (Consent for Secondary Use):

- Explicit consent obtained for secondary personal AI purpose
- Processing method constraints make scope of authorization clear

K.8 Architectural Benefits for Clean-AI

Sets Foundation for ISO/IEC 27560-2: Private-and-Clean-AI:

1. **Provenance transparency**: Every training sample has auditable authorization chain
2. **Clean training data**: Demonstrates data authorized for AI use (not scraped/unauthorized)
3. **Individual control**: PII Principal as Master Controller for personal models
4. **Surveillance mitigation**: No corporate access to personal training data
5. **Regulatory capacity**: Automated verification of authorization scope

Bridge to 27560-2:

- **27560-1 Appendix K**: Defines private-personal-AI processing method (individual-controlled)
- **27560-2**: Extends to clean-AI training data provenance for ANY lawful basis (not just consent)
- **27560-2 scope**: Shared models, federated learning, aggregate analytics—all with provenance proof

K.9 Implementation Requirements

Controllers claiming Private-Personal-AI conformance SHALL:

1. **Issue notice tokens** with `processing_method="private_personal_ai_only"`
2. **Enforce processing constraints**: Prevent third-party disclosure, model sharing, gradient sharing
3. **Maintain provenance chains**: Document authorization from source controllers
4. **Enable verification**: Publish Notice Event Log for regulatory verification

5. **Honor retention periods**: Delete training data when token expires

K.10 Use Cases

Personal Health Assistant:

- Individual trains local LLM on own medical history
- Model provides health insights without cloud disclosure
- Doctor's authorization includes secondary purpose for personal AI

Private Knowledge Base:

- Individual trains model on personal documents, emails, notes
- Model answers questions using individual's own knowledge
- No third-party service provider access

Local Financial Advisor:

- Individual trains model on personal financial history
- Model provides budgeting and investment insights
- Bank authorizes secondary purpose for personal AI (not shared model training)

K.11 Differentiation from Privacy-Preserving AI

Critical Distinction:

- **Privacy-preserving AI**: Organizations protect data they control (top-down, controller-centric)
- **Privacy-enabling Personal AI**: Individuals control whether to share data (bottom-up, individual-centric)

Example:

- ❌ **Privacy-preserving**: Hospital trains AI with differential privacy on patient data (hospital controls data)
- ✅ **Privacy-enabling**: Patient trains personal AI with own health data (patient controls data)

Appendix K Focus: Privacy-enabling architecture where individual is Master Controller.

Appendix A: V1.1 Field Reference

New Fields Added in V1.1

Privacy Preference Signals:

- privacy_preference_signals (object)
- gpc_honored (boolean)
- gpc_scope (array)
- dnt_recognized (boolean)
- preference_binding (enum)
- gpc_received_timestamp (datetime)
- preference_enforcement_mechanism (enum)

TATA Levels:

- transparency_assurance_level (integer 1-4)
- tata_description (string)
- registry_verification_url (string)
- registrar_signature (object)
- trustmark_certifications (array)
- visual_trust_indicator (object)

GPRC:

- global_privacy_rights_controls (array)
- context_id (string)
- data_control_vector (enum)
- rights_available (array)
- rights_exercise_mechanisms (object)
- preference_signal_integration (object)

AI System Transparency:

- ai_system_documentation (object) [from v1.0 Appendix J, now normative]
- ai_training_architecture (string)
- federated_learning_disclosure (object)

Active State Validation:

- cir_hash_at_issuance (string)
- hash_algorithm (string)
- active_state_validation (object)
- validation_endpoint (string)

Appendix B: Migration from V1.0 to V1.1

Backward Compatibility

V1.0 implementations remain fully conformant:

- V1.1 adds optional enhancements only
- No breaking changes to v1.0 mandatory fields
- V1.1 systems can interoperate with v1.0 systems

Migration Path

Step 1: Assess Current V1.0 Conformance

- Verify CIR publication, Notice Event Log, bilateral receipts operational

Step 2: Select V1.1 Enhancements

- Choose which v1.1 features to adopt based on use case:
    - Privacy preference signals for GPC support
    - TATA Level 3/4 for high-assurance contexts
    - GPRC for multi-controller environments
    - AI transparency for AI system deployments

Step 3: Implement Selected Features

- Add new fields to existing CIR and notice receipts
- Extend Notice Event Log with new event types
- Update verification protocols for enhanced features

Step 4: Test Interoperability

- Verify v1.1 receipts work with v1.0 systems (graceful degradation)

- Confirm v1.1 enhancements functional

Step 5: Claim V1.1 Conformance

- Update schema_version to "27560-UNIVERSAL-NOTICE-2025-1.1"
- Document which v1.1 conformance levels achieved
- Publish updated CIR with v1.1 features

Appendix C: Reference Implementations

(Coming in future update - Q1 2026)