



Data and Information Management Domain Document

New York State Department of Health (DOH)
Office of Health Insurance Programs (OHIP)

October 23, 2026

Contents

1.0	Introduction	1
1.1	Purpose.....	1
1.2	Objectives	1
1.3	Authority	2
1.4	Scope	2
1.5	How This Document Is Organized.....	3
1.6	Companion Standards	3
1.7	Compliance Tier Framework	4
1.8	Compliance and Enforcement.....	6
2.0	Data Governance Standard.....	9
2.1	Introduction	9
2.2	Objectives	9
2.3	General Data Standards	10
2.4	Other Standards.....	10
2.5	Data Creation and Maintenance.....	11
2.6	Data Corrections	12
2.7	Data Management Plan Requirements	13
2.8	Master and Reference Data	13
2.9	Master Data and Reference Data Management Guidelines	14
2.10	Metadata Management	15
2.11	Data Models	18
2.12	Data Integration and Controls	19
2.13	Data Quality	19
2.14	Data History	22
2.15	Defining and Capturing Data Requirements.....	22
2.16	Test Data.....	23
3.0	Data Architecture and Naming Standard.....	25
3.1	Introduction	25

3.2	Objectives	26
3.3	Applications and Exclusions.....	27
3.4	General Data Objects Naming Principles and Standards.....	28
3.5	Conceptual and Logical Naming Standards	32
3.6	Physical Naming Standards	34
3.7	General Modeling Standards.....	37
3.8	Data Architecture Artifacts	40
3.9	Maintaining the Data Architecture and Naming Standard.....	59
3.10	Related Data Maintenance Topics and Best Practices	59
4.0	Database Design Standard	60
4.1	Introduction	60
4.2	Objectives	60
4.3	Relationship to Other OHIP Standards.....	61
4.4	Database Implementation Standards	61
4.5	Database Lifecycle Management	65
4.6	Database Security.....	71
5.0	Data Conversion Standard	72
5.1	Introduction	72
5.2	Objectives	73
5.3	Companion Standards	73
5.4	Guiding Principles	73
5.5	Pre-Conversion Standards	74
5.6	Conversion Execution	77
5.7	Testing and Validation	84
5.8	Cutover, Rollback, and Post-Go-Live	85
5.9	Audit, Reporting, and Transparency.....	88
5.10	Security and Privacy	89
6.0	Data Lakehouse Architecture Standard.....	90
6.1	Introduction	90
6.2	Objectives	90
6.3	Functional Responsibilities.....	91
6.4	Layered Data Architecture.....	92
6.5	Data Ingestion Standards.....	95
6.6	Data Quality Gates.....	98

6.7	Data Remediation and Quarantine	105
6.8	Data Lineage	107
6.9	Data Versioning	109
6.10	Data Catalog and Metadata Management	110
6.11	Schema Evolution and Change Control	113
6.12	Data Retention and Lifecycle Management.....	114
6.13	Business Continuity and Disaster Recovery.....	116
6.14	Data Quality Testing Standards	119
6.15	Platform-Specific Implementation Guidance	121
6.16	Performance Monitoring and Key Performance Indicators (KPIs)	128
6.17	Compliance and Enforcement	129
7.0	Content and Document Management Standard	130
7.1	Introduction	130
7.2	Objectives	130
7.3	Scope and Applicability	130
7.4	Content Lifecycle Management.....	131
7.5	Metadata and Catalog Integration	133
7.6	E-Discovery Search and Production.....	133
7.7	Content Quality and Information Governance	134
7.8	Tools and Standards.....	134
7.9	Compliance and Enforcement	134
8.0	Contact Information	134
9.0	Revision History	134
	Appendices	136
	Appendix A: Deliverables and Artifacts.....	137
	Appendix B: Master Requirements Summary	145
	Appendix C: MITA Service Definition Packages	209
	Appendix D: Acronyms	212

Table of Tables

Table 1 – How This Document is Organized	3
Table 2 – Companion Standards.....	3
Table 3 – Compliance Tier Framework	4
Table 4 – Domain Requirements Summary	5
Table 6 – Retention Requirements by Data Classification	13
Table 7 – Categories of Metadata	15
Table 8 – Data Quality Dimensions	19
Table 9 – Test Data Specifications.....	23
Table 10 – Naming Levels.....	25
Table 11 – General Naming Class Words	30
Table 12 – Application of Abbreviation Rules.....	32
Table 13 – Surrogate Key Naming Class Words	36
Table 14 – Model Quality Validation Criteria	39
Table 15 – Conceptual Data Model	41
Table 16 – Logical Data Model.....	42
Table 17 – Physical Data Model.....	44
Table 18 – Pre-Packaged Database Data Elements	45
Table 19 – Sections and Necessary Properties of the Physical Data Model	46
Table 20 – Canonical Data Model	52
Table 21 – Metadata Data Model	56
Table 22 – Metadata Type, Purpose, Component, and Requirement	57
Table 23 – Referential Integrity and Constraint Standards.....	63
Table 24 – Audit Column Standards	64
Table 25 – Database Maintenance and Cleanup	66
Table 26 – Database Change Control	67
Table 27 – Data Conversion Guiding Principles.....	73
Table 28 – Pre-Conversion Phase Activities	78
Table 29 – Conversion Phase Activities	79
Table 30 – Post-Conversion Phase Activities.....	79

Table 31 – Data Acceptance Criteria.....	81
Table 33 – Data Lakehouse Functional Responsibilities	91
Table 34 – Consumer Access Tiers	92
Table 35 – Mandatory Metadata Fields	96
Table 36 – Quality Gate 1: Data Validation Requirements and Thresholds	99
Table 37 – Quality Gate 2: Data Validation Requirements and Thresholds	100
Table 38 – Data Quality Thresholds by Criticality	101
Table 39 – Remediation Timeframes	106
Table 40 – Mandatory Data Catalog Attributes	111
Table 41 – Schema Change Classifications.....	113
Table 42 – Recovery Objectives	116
Table 43 – Data Delivery Mechanisms.....	126
Table 44 – Database Naming.....	127
Table 45 – Schema Naming.....	127
Table 46 – Warehouse Naming.....	127
Table 47 – Other Object Naming.....	128
Table 48 – Performance Monitoring KPIs.....	128
Table 49 – Mandatory Content Metadata.....	133
Table 50 – Data Governance Standard Deliverables and Artifacts	137
Table 51 – Data Architecture and Naming Standard Deliverables and Artifacts	139
Table 52 – Database Design Standard Deliverables and Artifacts	139
Table 53 – Data Conversion Standard Deliverables and Artifacts.....	140
Table 54 – Data Lakehouse Architecture Standard Deliverables and Artifacts	143
Table 55 – Content and Document Management Standard Deliverables and Artifacts	143
Table 56 – Master Requirements Summary Table	145

Table of Figures

Figure 1 - Sample Conceptual Data Model	41
---	----

Figure 2 - Sample Logical Data Model	43
Figure 3 - Sample Physical Data Model	45
Figure 4 - Sample Canonical Data Model.....	52
Figure 5 - Sample Canonical Data Model in Swagger.....	53
Figure 6 - Sample Metadata Architecture Model	57
Figure 7 - Database Change Control Workflow	67
Figure 8 - Data Fix Decision Tree	70
Figure 9 - Data Conversion Phases and Activities	78
Figure 10 - Cutover Go/No-Go Decision Flow.....	86
Figure 13 - OHIP Data Lakehouse Architecture	93
Figure 14 - Data Quality Gate Process Flow	99
Figure 15 - Data Remediation Workflow.....	105

Issued by:	Finalized:
Office of Health Insurance Programs	10/23/26
Data and Information Management Domain Document	Owner: Division of Operations and Systems (DOS) and Chief Technology Office (CTO)

1.0 Introduction

1.1 Purpose

The Office of Health Insurance Programs (OHIP) Data and Information Management Domain Document establishes technical, operational, and quality standards governing how OHIP Medicaid Enterprise Systems (MES) solutions are designed, developed, tested, and delivered. This domain standard ensures all contractors implementing or enhancing OHIP systems do so with consistency, auditability, and in compliance with federal certification requirements.

The OHIP standards establish the mandatory technical, operational, and security requirements for New York State to remain compliant with federal regulations under 42 CFR § 433.112, qualifying the state for enhanced Federal Financial Participation (FFP). Deliverables and artifacts associated with this standard can be found in [Appendix A](#).

1.2 Objectives

The objectives of this consolidated standard are to:

- Align data strategies and architecture with the Medicaid Information Technology Architecture (MITA) Initiative and the Centers for Medicare and Medicaid Services (CMS) Conditions for Enhanced Funding (CEF).
- Ensure data is managed as a strategic asset in compliance with all applicable federal and state regulations, including HIPAA, 42 CFR Part 431, IRS Publication 1075, and NYS Public Health Law.
- Improve enterprise-wide data quality, consistency, and reliability by defining standards for data creation, maintenance, quality control, and life cycle management across all data domains.
- Enable interoperability for data exchange using standardized formats including HIPAA EDI X12, HL7v2, FHIR, and USCDI v4+ appropriate to each transaction type.
- Define mandatory architecture modeling practices for data architecture artifacts. Business, application, technology, and security architecture artifact requirements are defined in the OHIP Solution Delivery Domain Standard, Business and Technical Architecture Standard. Data architecture artifacts must align with MITA 3.0, TOGAF, and ArchiMate 3.1, with completeness, quality, and traceability requirements for all models.
- Establish consistent naming conventions for all data elements, entities, attributes, relationships, and physical database objects across all OHIP systems.

- Enforce physical database design, referential integrity enforcement at the database layer, lifecycle management, and change governance standards.
- Define rigorous requirements for data conversion and migration including source profiling, transformation, validation, cutover, and post-conversion retention.
- Standardize data analysis methodologies, statistical rigor, reproducibility, federal and state reporting, and data visualization standards.
- Establish the OHIP Data Lakehouse architecture: layered design (Raw, Processed, Curated), quality gates, lineage, versioning, metadata cataloging, schema evolution, retention, and disaster recovery.
- Provide enforceable requirements that establish clear expectations for Contractors implementing and maintaining data systems on behalf of OHIP.

 **Note:** This Domain Document references knowledge areas from the Data Management Body of Knowledge (DMBOK) published by the Data Management Association (DAMA) International for best practices, standards, and guidelines to support the management of data within an organization.

1.3 Authority

The New York State Department of Health (DOH) is the primary authority overseeing the health, safety, and well-being of New Yorkers since 1901. The Department delegates direct operational control and administrative responsibility for the Medicaid Program to the OHIP. Title 11 of the New York State Social Services Law enables OHIP to establish and enforce the standards necessary to administer the State's Medicaid Program. This foundational authority, further supported by the broad regulatory powers vested in the Commissioner of Health by the NYS Public Health Law, enables OHIP to create comprehensive policies and procedures required for the effective, secure, and compliant operation of the MES.

For health information standards, compliance with HIPAA regulations is mandatory, with specific New York enforcement through the NYS Public Health Law. Article II of the NYS Technology Law mandates security policies that comply with both State and federal requirements. The NYS Information Security Breach and Notification Act ensures entities managing sensitive data adhere to privacy and security protocols.

OHIP applies these standards across both hybrid and non-hybrid entities in the Department, ensuring robust protection for all sensitive information including Protected Health Information (PHI), Personally Identifiable Information (PII), Medicaid Confidential Data (MCD), IRS Federal Tax Information (FTI), and other regulated data, aligning with federal and State mandates for data integrity, security, and privacy.

1.4 Scope

This Standard applies to specific contractually negotiated activities between OHIP and existing contracts and to new procurements/contracts identified as part of the evolving MES Roadmap.

The Contractor **MUST** demonstrate how the requirements of this standard will be met within their specific system context. Where system-specific constraints exist, the Contractor **MUST** document those constraints and propose alternative approaches for OHIP approval that achieve an equivalent level of compliance and rigor. Proposed alternatives do not relieve the Contractor of any obligation under this standard unless OHIP provides written approval through the Standards Deviation Request (SDR) process defined in [Section 1.7.1](#).

1.4.1 Multi-System Applicability

OHIP operates a diverse portfolio of systems that vary in architecture, technology stack, and operational maturity. This standard is platform-agnostic in its requirements while acknowledging that implementation approaches can differ across OHIP's portfolio of systems.

1.5 How This Document Is Organized

This document encompasses multiple related OHIP standards. Each component standard is presented as a dedicated section (Sections 2 through 7). An Appendix and Master Requirements Summary Table appears at the end of the document.

Table 1 – How This Document is Organized

Section	Source Standard
Section 2	Data Governance Standard
Section 3	Data Architecture and Naming Standard
Section 4	Database Design Standard
Section 5	Data Conversion Standard
Section 6	Data Lakehouse Architecture Standard
Section 7	Content and Document Management Standard

1.6 Companion Standards

The following external OHIP standards directly govern, extend, or bound activities described in this domain standard. Standards previously cited as companions that have been consolidated within this document are now cross-referenced by section number. All references to individual component standards within Sections 2–7 **SHOULD** be read as references to that section of this document.

 **Note:** Internal cross-references between sections of this consolidated document (Sections 2 through 7) are provided within each section's introduction. The table below lists only external companion standards that are maintained as separate documents.

Table 2 – Companion Standards

OHIP Standard	Domain	Scope Relationship
OHIP Security and Privacy Compliance Standard	Security, Privacy and Compliance	Governs encryption, access control, de-identification, BAA execution, and PHI/PII classification. All data standards in this document defer to this standard for security matters.
OHIP Usability and Accessibility Standard	Solution Delivery	Governs Section 508 compliance, WCAG 2.2 Level AA, and accessibility requirements for dashboards and user-facing applications.
OHIP ITSM Standard	Infrastructure and Operations	Governs general change management, incident management, capacity planning, and escalation procedures. Data-specific requirements herein operate within that framework.
OHIP SDLC Standard	Solution Delivery	Governs development lifecycle, testing, and release management. Data requirements herein supplement SDLC testing requirements.

OHIP Business and Technical Architecture Standard	Solution Delivery	Governs business architecture artifacts (capability models, process models, use case models), application architecture artifacts (integration architecture, component models, solution architecture), technology architecture artifacts (deployment models, network topology), and security architecture artifacts.
OHIP Test Management Standard	Solution Delivery	Governs testing methodology, test environment management, UAT, and defect tracking. Lakehouse-specific testing requirements herein are supplements.
OHIP Preferred Operational Tools Standard	Infrastructure and Operations	Governs approved analytical, modeling, data quality, and project tracking tools. All tool selections MUST align with the approved list.
OHIP Log Management Standard	Infrastructure and Operations	Governs retention requirements for pipeline logs, system logs, audit logs, and operational logs.

1.7 Compliance Tier Framework

This standard uses three compliance tiers applied consistently throughout all sections and in the Master Requirements Summary in [Appendix B](#). Every requirement has been individually vetted against enforceability, technical precision, and practicality before receiving its tier assignment. Tier designations appear throughout the document and in the Tier column of [Appendix B](#).

Table 3 – Compliance Tier Framework

Tier	Type	Definition
MUST	Mandatory	MUST requirements are mandatory. Violation constitutes a material breach. Any deviation requires submission of a formal SDR and written approval from the OHIP IT Standards Team prior to implementation. MUST requirements are binding on all procurement activities and contracts and are subject to audit as part of CMS SMC certification reviews.
SHOULD	Strong Recommendation	SHOULD requirements are strongly recommended. Exceptions MAY be permitted under specific documented circumstances. Any deviations require written justification to the OHIP Project Manager for approval and MUST be documented in the project deviation log. SHOULD requirements are cited as expected practice in procurement activities and contracts. An SDR is not required; however, all deviations MUST be affirmatively justified in writing.
MAY	Permitted Option	MAY requirements denote a permitted option or permissible alternative at the discretion of the Contractor or OHIP. When a MAY option includes stated conditions, those conditions MUST be satisfied before the option MAY be exercised. MAY statements are informational within procurement activities and contracts and do not require prior approval or deviation documentation.

1.7.1 Standards Deviation Request (SDR) Process

A SDR is the formal mechanism by which a Contractor seeks approval to deviate from a **MUST** requirement prior to implementation. An SDR does not apply to **SHOULD** or **MAY** requirements. The SDR

process is prospective: it **MUST** be completed and approved prior to the deviating approach is implemented. Retroactive SDRs will not be accepted and do not cure a breach.

The Contractor **MUST** submit a completed SDR to the OHIP IT Standards Team for any proposed deviation from a **MUST** requirement. An SDR is a prospective request: it **MUST** be submitted and approved by OHIP before the Contractor implements any non-compliant approach. The SDR package **MUST** include:

- **Requirement ID:** the specific requirement identifier from Appendix B
- **Proposed Alternative:** a complete description of the approach to be used instead of the **MUST** requirement
- **Technical Justification:** documented rationale explaining why the **MUST** requirement cannot be met
- **Risk Assessment:** identification of risks introduced by the deviation and their likelihood and impact
- **Mitigation Controls:** specific measures that will reduce or compensate for identified risks
- **Proposed Timeline:** target date for achieving full compliance, if the deviation is temporary

OHIP will review the SDR and issue a written approval or denial within ten (10) business days of receipt of a complete submission. Incomplete submissions will be returned without review. Approved SDRs are logged in the OHIP Deviation Register with the effective date, any conditions attached to the approval, and the anticipated compliance date where applicable. Conditional approvals **MUST** be re-reviewed at the specified review date.

 **Note:** OHIP reserves the right to revoke an approved SDR if circumstances change, if conditions are not met, or if the deviation creates unforeseen risk to program integrity, federal compliance, or data security. Revocation of an SDR is subject to reasonable notice except where urgent risk requires immediate action.

1.7.2 Domain Requirements Summary

The following table summarizes requirement counts across the 6 component standards. A complete master listing of all requirements with individual tier assignments appears in [Appendix B](#).

Table 4 – Domain Requirements Summary

Standard	Section	ID Range	MUST	SHOULD	MAY	Total
Data Governance Standard	Sec. 2	DATA-0001 – DATA-0118	88	24	6	118
Data Architecture and Naming Standard	Sec. 3	DATA-0119 – DATA-0230	97	7	8	112
Database Design Standard	Sec. 4	DATA-0231 – DATA-0326	86	2	8	96
Data Conversion Standard	Sec. 5	DATA-0327 – DATA-0430	86	11	7	104
Data Lakehouse Architecture Standard	Sec. 6	DATA-0431 – DATA-0593	141	9	13	163

Content and Document Management Standard	Sec. 7	DATA-0594 – DATA-0622	26	3	0	29
Total			524	56	42	622

Table key: **MUST** = mandatory. **SHOULD** = strongly recommended. **MAY** = permitted option.

1.8 Compliance and Enforcement

This section establishes the obligations, rights, and remedies that govern compliance with this standard throughout the contract term. This standard is incorporated into the Contract, and OHIP reserves the right to update it during the term; the version in effect at the time of performance applies to the requirements it sets. The compliance and enforcement provisions in this section supplement, and do not replace, the remedies and enforcement terms of the governing Contract.

1.8.1 Applicability and Effective Date

The Contractor **MUST** comply with this standard and all applicable laws, regulations, and funding requirements. Compliance is required from the effective date of the Contract or Contract Amendment.

1.8.2 Updates to the Standard

OHIP reserves the right to amend this standard at any time. OHIP will use reasonable efforts to provide at least thirty (30) days' notice before enforcement of substantive changes. The Contractor **MUST** implement changes by the designated effective date.

For changes required by law, funding conditions, or urgent security risks, OHIP reserves the right to designate an earlier effective date. The Contractor **MUST** comply by that date regardless of other project commitments.

Within ten (10) business days of receiving notice of a substantive change, the Contractor **MUST** submit an impact assessment to OHIP and, if applicable, a proposed implementation plan including any requested cost or schedule adjustments. Acceptance of adjustments is at OHIP's sole discretion.

1.8.3 Monitoring, Audit, and Records Retention

OHIP and its designees reserve the right to audit the Contractor's and subcontractors' compliance with this standard upon ten (10) business days' notice, or less for cause or urgent security issues. The Contractor **MUST** provide reasonable access to relevant facilities, systems, records, and personnel upon request.

The Contractor **MUST** retain records evidencing compliance with this standard for at least seven (7) years from creation, or as otherwise required by law or funding requirements, whichever is longer.

1.8.4 Notification of Non-Compliance and Security Events

The Contractor **MUST** notify OHIP in writing within three (3) business days upon identification of any non-compliance with this standard.

Where non-compliance materially impacts the confidentiality, integrity, or availability of OHIP systems or data, or could result in legal or funding exposure, notification **MUST** be made immediately and no later than twenty-four (24) hours after discovery.

All notifications **MUST** describe: the nature and scope of the issue, root cause if known, immediate containment steps taken, and a proposed Corrective Action Plan (CAP) with timeline.

1.8.5 Corrective Action Plan (CAP)

Within ten (10) business days of notice, whether by the Contractor or OHIP of any non-compliance with this standard, the Contractor **MUST** provide a CAP acceptable to OHIP. The CAP **MUST** include defined milestones, responsible parties, and due dates for each remediation step.

OHIP reserves the right to impose or modify a CAP where the Contractor's proposed plan is insufficient.

1.8.6 Remedies and Enforcement

In addition to any other rights and remedies available under the Contract or applicable law, OHIP reserves the right to exercise any or all of the following remedies upon Contractor non-compliance:

- Withhold payments due under the Contract
- Assess service credits or liquidated damages as set forth in the Contract
- Suspend or limit the Contractor's system access
- Require removal of personnel or sub-contractors
- Require re-performance of non-compliant work at no additional cost to OHIP

Remedies are cumulative and can be exercised concurrently. OHIP can also refer matters to oversight bodies or regulators as required by law or policy.

1.8.7 Exceptions and Risk Acceptance

An exception is distinct from a SDR (see [Section 1.7.1](#)). An SDR is a prospective mechanism, it is submitted before a non-compliant approach is adopted. An exception, by contrast, applies where strict compliance is not technically feasible after implementation has begun, or where an existing system or constraint prevents full compliance by the effective date. OHIP will evaluate each exception request on its individual merits and will not grant exceptions that would violate applicable law, federal funding requirements, or CMS certification conditions.

To request an exception, the Contractor **MUST** submit a written request to OHIP that includes: a risk assessment identifying the nature and severity of non-compliance risk; proposed mitigation measures that reduce that risk to an acceptable level; and a proposed timeline either for achieving full compliance or for implementing a permanent alternative solution.

Exceptions are subject to the following conditions:

- Exceptions are time-bound—they expire at the date specified in the approval unless renewed by OHIP in writing.
- Exceptions are non-transferable—they apply only to the specific Contractor, system, and circumstance described in the request.
- Exceptions require OHIP's prior written approval before the non-compliant approach is adopted or continued.
- No exception will be granted that would violate applicable law, federal funding requirements, or OHIP policy.

- OHIP can revoke an approved exception at any time upon material changes in risk, regulatory requirements, or program needs.

 **Note:** *Approved exceptions are tracked in the OHIP Deviation Register alongside approved SDRs. Both represent formal acknowledgment by OHIP of a documented departure from standard requirements. Neither constitutes a waiver of OHIP's rights under the Contract.*

2.0 Data Governance Standard

2.1 Introduction

This Data Governance Standard establishes the foundational policies, principles, and mandatory requirements for managing data as a strategic asset across the OHIP MES. As the governing standard within the Data and Information Management domain, this section sets the baseline expectations that all other domain standards build upon. Deliverables and artifacts associated with this standard can be found in [Appendix A](#).

This standard governs how OHIP data is created, classified, owned, maintained, protected, and retired across the full data lifecycle. It addresses:

- **Federal and Interoperability Compliance:** Ensuring all data adheres to CMS MITA, HIPAA, ONC interoperability rules, and applicable federal data standards.
- **System of Record (SOR) Framework:** Defining how authoritative data sources are designated, maintained, and communicated across the MES ecosystem.
- **Master and Reference Data Management:** Setting standards for the creation, modification, and governance of shared master data domains and enterprise reference data.
- **Metadata Management:** Requiring that business, technical, operational, and data quality metadata be defined, documented, and maintained for all enterprise data assets.
- **Data Quality, History, and Integration Controls:** Establishing requirements for data quality classification, data history preservation, and data integration controls.

This standard does not govern the physical implementation of databases, conversion and migration processes, analytical methodologies, or Lakehouse platform architecture. Those requirements are defined in Sections [4.0](#), [5.0](#), [6.0](#), and [7.0](#) of this Domain Document, respectively.

2.2 Objectives

The objectives of this Standard are to:

- Align data strategies and architecture with the MITA Initiative.¹
- Ensure data is managed as a strategic asset in compliance with all federal and state regulations.
- Improve enterprise-wide data quality, consistency, and reliability by defining standards for data creation, maintenance, and quality control.
- Enhance data interoperability and sharing by standardizing master data, reference data, and metadata management practices.
- Protect data assets and prevent data loss by establishing clear policies for data security, retention, and archiving.

¹ For more information, refer to [MITA 3.0 Part II: Information Architecture](#).

2.3 General Data Standards

2.3.1 Federal Compliance

The Contractor **MUST** ensure that all data adheres to relevant federal standards to enable interstate consistency and to comply with funding requirements. This standard generally applies across states and aligns with the CMS MITA framework and the CMS Streamlined Modular Certification (SMC) Requirements, detailed below.

The Contractor **MUST**:

- Adhere to federal interoperability standards identified by Health and Human Services (HHS) and the Office of the National Coordinator (ONC) for health information exchange.
- Adopt the United States Core Data for Interoperability (USCDI) v4+ as the baseline for certified health IT systems, in compliance with the Health Technology and Interoperability (HTI) HTI-1 and HTI-2 Rules.

The Contractor **SHOULD**:

- Align with the ONC Interoperability Standards, such as the USCDI for core data elements and data classes, where applicable, to ensure data content complies with national definitions.
- Employ other federally recognized standards for vocabulary (e.g., SNOMED CT, LOINC, and RxNorm) and transport where appropriate.

2.4 Other Standards

2.4.1 Data Ownership and Portability

Data ownership remains with the State regardless of hosting model. The Contractor **MUST**:

- Ensure that provisions for regular data extracts or backups in a usable format are available for transfer to a State-controlled environment for any Software as a Service (SaaS)/Commercial Off-The-Shelf (COTS) solution, so that data can be converted without loss during solution replacement or upgrades.
- Adhere to [Section 5.0](#), which governs data export processes, mapping, and transformation rules.
- Submit a SaaS Data Conversion Plan for OHIP approval before any conversion activities begin. The plan **MUST** describe the end-of-life data export process (e.g., full database export or Application Programming Interface [API] retrieval of all records) to ensure continuity into the successor solution.

2.4.2 Configuration Over Customization

When configuring data exchange mappings, the Contractor **SHOULD** use tools or standard middleware rather than hard-coding logic within the application code. Modern COTS solutions typically provide configurable mapping fields for external data formats. The Contractor **SHOULD** utilize these features to keep conversion logic maintainable and avoid custom code that complicates upgrades.

2.4.3 SaaS Updates and Versioning

Cloud-based solutions are subject to frequent updates. The Contractor **MUST**:

- Establish a process with OHIP to evaluate how updates may impact data interfaces.
- Enforce versioned APIs or backward compatibility for data exchange endpoints so that an update does not break the State's data flows.²
- Conduct regression testing for data exchanges before major software upgrades. Versioning and regression testing requirements for data interfaces are further governed by [Section 5.0](#) and the OHIP Solution Delivery Domain Document, Test Management Standard.

2.5 Data Creation and Maintenance

This section establishes the mandatory rules for managing the SOR across all data types, including derived data. Data **MUST** be strictly controlled when entering the OHIP Enterprise, and a SOR **MUST** be identified for each data element. The Contractor **MUST** make changes to data elements in the SOR and communicate changes to relevant stakeholders.

2.5.1 Standard Definitions and Guidelines for the System of Record (SOR)

The Contractor **MUST** comply with the following SOR requirements:

- Data ownership and accountability **MUST** be assigned and executed per OHIP direction.
- If current data needs to be corrected after initial setup, the correction **MUST** be made in the SOR, and the solution **MUST** enforce the business rules used during setup, including data validation, default values, and referential integrity rules.
- The SOR **MUST** preserve the identity of the staff member or system that changed the data and the date and time the change occurred for every create, update, or delete action.
- The SOR **MUST** be defined for each individual data element.
- The SOR **MUST** include functionality to correct data through a system interface rather than through manual or scripted changes to the database that might bypass security and data validation rules.
- If a downstream solution or user requires derived data, the derived data elements **MUST** be preserved. The solution that initially creates the derived data element **MUST** become the SOR for that element, and users **MUST** reference that element from the SOR. Systems other than the SOR **MUST NOT** recreate intermediate results that the SOR did not preserve.
- Business data that supports a business event or decision **MUST** be preserved so that the reference data values effective at the time of the event can be retrieved and associated with it.
- Data parameters that impact values preserved by the SOR **MUST** be preserved so that parameter values in effect at a specific time can be determined. Parameter data **MAY** include effective dates or factors used to calculate projections.

² Refer to the OHIP Infrastructure and Operations Domain Document, Application Integration Standard for more information.

- Additive data, typically summed to create a total billing amount for each service, **MUST** be changed using a cancel-and-correct approach where the previous amount is negated and the new amount is added, ensuring a proper audit trail.
- Data **MUST** be created or modified only within authoritative sources. An authoritative source is the SOR or an OHIP-approved alternative data source designated to provide specific information to downstream solutions or users.
- Data security and sensitivity classification of data sources **MUST** be published and easily accessible to users.
- Data users, business projects, solutions, and processes **MUST** use business information only from approved authoritative sources.

The Contractor **SHOULD**:

- Communicate data corrections to all affected end-users when a data value is corrected after initial setup.

Data within the OHIP MES is classified into the following categories: Master Data (shared across business areas and used as key dimensions for aggregation), Reference Data (codes, abbreviations, and definitions that are seldom changed), Metadata (structured descriptions of data characteristics), Transaction Data (data from business events such as encounters, claims, or enrollments), and Derived Data (data resulting from transformation or aggregation of other data). Detailed definitions are provided in [Section 2.10](#) and [Section 2.12](#).

The Contractor **MUST** support OHIP's annual review of data sources by providing automated reports on security classification, sensitivity classification, access classification, and compliance status.

2.6 Data Corrections

The Contractor **MUST**:

- Ensure solutions include the ability for authorized users to make changes to data in a controlled manner. Where deletion of business data in a production environment is necessary, the changes **MUST** follow a defined change control process with documented notification.³
- Use a tracking tool to log, track, and report all data correction requests, changes, and approvals.
- Ensure corrections to data are made at the SOR and propagated to all downstream solutions. After data is corrected in the SOR, downstream solutions **MUST** reflect the corrected data.

2.6.1 Data Purge, Archive, Retention

Data retirement involves the disconnection of interfaces or ongoing access to the designated data and the archival or purge of that data. The Contractor **MUST** ensure that:

- A process for archiving and purging data is in place in compliance with all applicable federal, State, and OHIP retention policies, regulatory requirements, and non-destruct policies.⁴

³ Refer to the OHIP Program and Project Management Domain Document, Technical Change Control Standard for more information.

⁴ Refer to the OHIP Infrastructure and Operations Domain Document, Log Management Standard for more information.

- Data retirement, purge, and archiving rules are fully documented, traced, and logged.
- Archive locations are registered, dated, and maintained by the respective business units or records custodians.
- OHIP review and approval is obtained before any data retirement, purging, archival, or significant data changes occur.

The following archival approach **MAY** be used:

- Data archival based on a business unit of work tied to a business event (e.g., a business unit of work may be defined as creating coverage plans, and a business event may be members enrolled within those plans).

Table 5 – Retention Requirements by Data Classification

Data Classification	Minimum Retention	Regulatory Basis
PHI	7 years	HIPAA requirements; NYS Public Health Law; Federal Medicaid retention rules
FTI	7 years	IRS Publication 1075; Internal Revenue Code (IRC) Section 6103
MCD	7 years	42 CFR Part 431; CMS retention requirements
PII	7 years	NYS Personal Privacy Protection Law; agency retention schedules
INTERNAL	3 years	Operational requirements; audit support
PUBLIC	3 years	Operational requirements; MAY be extended for historical analysis

For datasets containing multiple classifications, the longest applicable retention period **MUST** be applied. The Contractor **MUST** document the retention period for each dataset in the data catalog and implement automated enforcement mechanisms.

2.7 Data Management Plan Requirements

The Contractor **MUST** develop and maintain a Data Management Plan that provides a structured approach to handling data throughout its lifecycle, encompassing collection, storage, analysis, sharing, and preservation. The Contractor **MUST** submit the Data Management Plan for Department review and approval prior to database design activities.

2.8 Master and Reference Data

Master data is data that is shared and required across business areas, processes, and solutions. It serves as a key dimension for aggregating data in the data warehouse or analytic store. Examples of master data in the OHIP MES include:

- **Provider:** provider ID, name, National Provider Identifier (NPI), address, type, specialty, and location(s).
- **Member:** member ID, name, date of birth, correspondence address, billing address, and phone number.

Reference data organizes, categorizes, or lists permissible data values. It usually consists of codes, abbreviations, and definitions and is constant or seldom changed. Examples include State codes, Diagnostic Codes (ICD-9/10), Current Procedural Terminology (CPT) codes, Member Status Codes (e.g., A-Active and I-Inactive), and Member Relationship Codes (e.g., S-Self, DS-Dependent Spouse, and DC-Dependent Child).

2.9 Master Data and Reference Data Management Guidelines

The Contractor **MUST** categorize all data domains and data elements and comply with the following standards:

- All master data domains and elements **MUST** have an assigned SOR as the Authoritative Source.
- Data categorized as master data **MUST** be consistently defined, and the BEDG **MUST** approve any changes to definitions.
- All downstream solution consumers or data users **MUST** source and use master data from the SOR.⁵
- All downstream business processes and data users **MUST** access master data from the SOR to reduce duplication and ensure consistency.
- Data categorized as master data **MUST** have specific access rights, data quality rules, and data quality thresholds documented, published, and maintained.
- Data **MUST** be created, updated, purged, and maintained in the master SOR.
- Metadata **MUST** be supplied for each master data element as defined in [Section 2.12](#).
- Master Data domains and associated Reference Data **MUST** be audited quarterly for data quality and OHIP Data Governance compliance. When exceptions are found, a CAP⁶ **MUST** be developed at the direction of the responsible Data Owner and OHIP.
- Hierarchies **MUST** be standardized, and the component information **MUST** be controlled and standardized. Hierarchies **MUST**:
 - o Be named consistently (including nodes) and comply with existing standards.
 - o Be utilized for regulatory reporting.
 - o Have a consistent rollup structure (i.e., the parent at any level **MUST** be at the same level for each named hierarchy, and child levels **MUST** be at the same level for each named hierarchy).
 - o Be structured to have at least one parent and one child to qualify as a hierarchy. Single nodes **MUST NOT** be used or classified as a hierarchy.
 - o Identify any alternate hierarchies, including the required exception from the State and the purpose for reporting.

The Contractor **MUST** adopt and maintain enterprise-standard code sets as designated by OHIP. Standard code sets include, but are not limited to, International Classification of Diseases (ICD) codes,

⁵ The SOR may be an MDM data hub.

⁶ For more information, refer to the OHIP Security, Privacy, and Compliance Domain Document, Security and Privacy Compliance Standard.

Current Procedural Terminology (CPT) codes, National Drug Codes (NDC), and any other code sets mandated by federal or State authorities. The Contractor **MUST** use enterprise-standard code sets where they exist and **MUST NOT** create custom code sets. All code set mappings and crosswalks **MUST** be documented, submitted to BEDG for review, and maintained in the Enterprise Data Catalog. Data exchange formats **MUST** comply with applicable federal and national standards including Health Level 7 (HL7), HIPAA EDI X12, and other standards as specified by OHIP.⁷

2.10 Metadata Management

Metadata is a structured description of data characteristics reflecting a data object's meaning, content, structure, usage, and purpose. Metadata categories include business, data quality, technical, and operational.

The Contractor **MUST** ensure that metadata is collected and standardized for all data domains and elements. As standardization across the ecosystem evolves, comparisons and trending metadata **MUST** provide insight into the health and progress of the environment towards achieving OHIP's Data Governance and data quality objectives. The categories of metadata identified in the subsections below **MUST** be defined and managed in an OHIP-approved solution or tool.

2.10.1 Categories of Metadata

[Section 3.0](#) defines an extended metadata framework of seven types (Structural, Syntactic, Semantic, Integration, Business, Operational, and Sensitivity) applicable to the architecture and data modeling context. The four categories defined in this section (Business, Data Quality, Technical, and Operational) represent the enterprise governance taxonomy for operational data management. These two frameworks are complementary and **MUST** be read together.

Table 7 provides the authoritative reconciliation mapping. Where a data element requires metadata that spans both frameworks, the Contractor **MUST** satisfy the requirements of this section and [Section 3.0](#). In any apparent conflict, the BEDG **MUST** issue a binding determination.

Table 6 – Categories of Metadata

Category in Section 2.0 of this Document	Category in Section 3.0 of this Document	Reconciliation Notes
Business Metadata	Business Metadata	Direct equivalents. Section 2.0 governs operational business metadata; Section 3.0 governs business metadata within architecture models.
Data Quality Metadata	No direct equivalent	Section 2.0 is the sole authoritative source for data quality metadata (allowable values, rules, targets, triggers, scores).
Technical Metadata	Structural, Syntactic, or Integration Metadata	Section 2.0 governs the operational layer (database names, schemas, tables, lineage, security, volumetrics). Section 3.0 governs the architecture and model layer. Both MUST be satisfied.

⁷ Refer to [Section 5.0](#) for more information.

Operational Metadata	Operational Metadata	Direct equivalents. Section 2.0 governs operational metadata for data pipelines and jobs; Section 3.0 governs operational metadata within architecture artifacts.
<i>Not addressed in this document</i>	Semantic Metadata	Governed by Section 3.0 . Covers meaning, context, and interpretation of data elements within architecture models.

2.10.1.1 Business Metadata

The Contractor **SHOULD** ensure that the following business metadata are defined, documented, monitored, and reported, as appropriate:⁸

- Business name
- Business definition
- Authoritative source(s)
- Data ownership roles, owners, and accountability
- Data ownership organization(s)
- Data category
- Data security classification

2.10.1.2 Data Quality Metadata

The Contractor **SHOULD** ensure that the following data quality metadata are defined, documented, monitored, and reported as appropriate:

- Allowable values
- Data quality rules
- Data quality targets/objectives
- Data quality triggers
- Data quality scores

2.10.1.3 Technical Metadata

The Contractor **SHOULD** ensure that the following technical metadata are defined, documented, monitored, and reported as appropriate:

- Database (DB) name, Internet Protocol/Uniform Resource Locator (IP/URL), or file name and path
- Schemas, schema descriptions, and versions

⁸ Additional metadata may be defined within specific procurement-related documents.

- Tables, table/field relationships, table descriptions, columns/fields, data types, index identification, type of key identification, field-level constraints, field descriptions, and field types (direct or derived)
- Data transformation and data lineage from SOR to identified field
- Derivations and/or calculations
- Security at database, schema, table, and column level
- Database or table partitions and archiving settings
- Alias information
- Volumetrics (table/database growth rates, current number of table rows, table usage characteristics, and indexing structures)
- User access and/or query patterns to specific tables or data elements; this **MAY** include information on user roles, frequency of access, and any transformations applied during data lineage tracing
- Aging/purge criteria

2.10.1.4 Operational Metadata

The following operational metadata **SHOULD** be defined, documented, monitored, and reported as appropriate:

- Data availability and refresh rates
- Data generation and update job schedule start date and time
- Expected completion date and time for data generation and update jobs
- List of jobs, including run sequences and dependencies
- Location of error logs

2.10.2 Enterprise Glossary

The Enterprise Glossary establishes and maintains the authoritative set of business terms, metrics, and concepts used across OHIP data assets.

The Enterprise Glossary and the metadata-management capability (the Data Catalog/metadata repository) are complementary rather than duplicative. The Enterprise Glossary is the authoritative, business-facing layer of terms, metrics, and concept definitions: it is written for non-technical users, owned by the BEDG and data stewards, and concerned with what a business concept means and who owns its definition. The Metadata Manager (Data Catalog) is the broader technical repository that registers all data assets and captures the full range of metadata—business, data-quality, technical, operational, and lineage—to support discovery, governance, and lifecycle management. In short, the Enterprise Glossary defines meaning while the Metadata Manager catalogs and governs the assets and their metadata; the glossary is maintained within the catalog as a governed component of it, and its terms link to physical columns.

Ownership and Governance

The Enterprise Glossary is maintained by the BEDG in collaboration with DOS and OHIP data stewards. The Contractor **MUST** ensure the glossary is kept current and accessible, with a defined process for approving, adding, updating, or deleting terms in a timely manner.

Catalog Integration

The Contractor **MUST**:

- Maintain the enterprise glossary within the Data Catalog, providing standardized definitions that enable consistent interpretation of data across different teams and use cases.
- Include a business definition for each glossary term that is written for non-technical users, includes any relevant regulatory or compliance context, maps to physical data elements in the catalog, and designates the term owner responsible for definition accuracy.
- Integrate the enterprise glossary with catalog search capabilities so that users searching for business concepts can discover relevant datasets.

Link glossary terms to column-level metadata where applicable, enabling users to understand the business meaning of specific data elements.

Data Definition Standards

In accordance with quality software guidelines, OHIP enforces detailed inline documentation and comprehensive description fields in data models, database fields, XML schemas, and other information definition and storage elements.

When defining an entity, the Contractor **MUST**:

- Begin the definition with the entity name in Title Case followed by "is" or "is a" (e.g., "Provider is an individual or organization..."). Do not begin with "The" or "A."
- Define as an instance (use "a/an") rather than a class (use "the").
- Define the entity as a thing, not a process.
- Avoid circular definitions. The definition **MUST NOT** restate the entity name as its own explanation.
- Where the entity name contains an acronym, spell out the acronym in full on first use within the definition.

When defining an attribute, the Contractor **MUST**:

- Begin the definition with the attribute name in Title Case followed by "is the" (e.g., "Service Date is the date on which...").
- Provide a singular, complete, and self-contained definition.
- Define as a fact, not a process.
- Avoid circular definitions.
- Include permissible values or value sets in the definition where applicable, particularly for coded attributes.

2.11 Data Models

Data models illustrate data concepts, entities, structures, and relationships at various abstractions, including conceptual, logical, and physical. The Contractor **SHOULD** adhere to OHIP's standardized

naming conventions⁹. The Contractor **MUST** create, update, and maintain data models per [Section 3.0](#), which defines the authoritative requirements for Conceptual, Logical, Physical, and Canonical Data Models, including notation, Data Dictionary properties, normalization, and traceability rules. Physical design requirements (including storage parameters, partitioning, encryption at the database layer, and Role-Based Access Control) are governed by [Section 4.0](#).

2.12 Data Integration and Controls

Data integration addresses how data is transmitted to and received by solutions and repositories. The Contractor **MUST** ensure that:

- Mapping and business rules, transformation scripts, source code, and technical logs for data movement between solutions are documented and published in a central and accessible OHIP-managed repository.
- Controls include, at a minimum, the detection, notification, and management of exceptions when moving information from source to target.
- PHI and PII data are encrypted at rest and in transit from source to destination. Encryption specifications, including applicable FIPS 140 series requirements and approved protocols (HTTPS, SFTP, SSL, VPN), are governed by the OHIP Security, Privacy, and Compliance Domain Document, Security and Privacy Compliance Standard.
- Proposed exceptions are escalated to OHIP for final determination.

The Contractor **SHOULD** ensure that:

- Data movement from one IT-supported data store to another is automated and verified through completeness and accuracy controls that compare data in the source to data in the target.

2.13 Data Quality

Data quality¹⁰ evaluates how well data complies with its intended purpose and requirements. Data quality **MUST** be assessed and reported against the following eight standard dimensions. The Contractor **MUST** apply these dimensions when defining data quality rules, thresholds, and metrics for all enterprise data elements. Minimum quality thresholds by data criticality for Lakehouse quality gate enforcement are specified in [Section 7.6.4](#).

Table 7 – Data Quality Dimensions

Data Quality Dimension	Definition
Accessibility	Data is available to authorized users in a usable form, without requiring interpretation that varies between users. Accessibility failures include data locked in inaccessible

⁹ Refer to [Section 3.0](#) of this document for more information.

¹⁰ The requirements governing how data quality must be measured, validated, and reported in analytical and reporting contexts are defined in [Section 6.0](#) of this document.

	formats, inconsistent representations of the same element, or data that requires undocumented tribal knowledge to use correctly.
Accuracy	Data correctly reflects the real-world entity or event it represents. Accuracy is verified through comparison to an authoritative source or SOR.
Completeness	All required data elements are present and populated. Completeness rules MUST define which fields are mandatory, conditionally required, or optional for each data domain.
Consistency	Data values are uniform and non-conflicting across systems, sources, and time periods. Where the same data element appears in multiple systems, values MUST NOT conflict with the authoritative SOR.
Timeliness	Data reflects the current state of the entity and is available when needed by downstream processes and business users. Both currency and delivery timeliness MUST be defined and measured for each data domain.
Uniqueness	Data is free of unintended duplicates. Duplicate records MUST be detectable, reportable, and resolvable through defined stewardship processes.
Validity	Data values conform to defined formats, reference sets, allowable value ranges, or domain constraints. Data MAY be structurally conformant but still flagged as invalid by a business rule; both conditions MUST be captured and tracked separately.
Integrity	Referential integrity is maintained across related data elements and entities. All required parent-child relationships and foreign key constraints MUST be intact.

2.13.1 Data Quality Standards

The Contractor **MUST** ensure that:

- Solutions **MUST** include both preventative and detective data quality controls. Preventative controls **MUST** block invalid data from being recorded in a SOR, and detective controls **MUST** identify errors and support the correction of those errors.
- Data quality issues are monitored, logged, and tracked using the standard tools designated by OHIP. The Contractor **MUST** escalate issues to OHIP as needed for resolution.
- Data quality metrics, standards, targets, and acceptable thresholds are defined for all enterprise data elements.

The Contractor **SHOULD** ensure that:

- Data quality rules are collected, validated, reconciled, maintained, and reviewed quarterly with OHIP.

Data quality controls **MUST** be implemented across three types, applied at appropriate points in the data lifecycle. The Contractor **MUST** implement controls spanning all three types for all enterprise data domains:

- **Preventive Controls:** Block invalid, incomplete, or non-conforming data from entering a SOR at the point of origin or ingestion. Examples include field-level validation rules in application interfaces, ETL rejection logic, and schema-enforced constraints that prevent non-compliant records from being written.

- **Detective Controls:** Identify data quality issues that exist within a data store after ingestion. Examples include data profiling, automated threshold monitoring, exception reporting, and reconciliation processes that surface anomalies for steward review.
- **Corrective Controls:** Address and remediate identified data quality issues. Examples include structured change request processes, batch correction jobs, steward-initiated data updates, and documented reprocessing workflows. All corrective actions **MUST** be logged and traceable to the originating detective or preventive control finding.

The Contractor **SHOULD** ensure that:

- Data quality performance is reported monthly or according to approved OHIP processes.
- The Data Owner requests a data quality review with OHIP to ensure the integrity of core data elements is maintained throughout the data lineage within the OHIP Enterprise.

2.13.2 Root Cause Analysis (RCA) and Continuous Improvement

Treating data quality symptoms without addressing their origins leads to recurring issues and erodes confidence in the data over time. The Contractor **MUST** ensure that:

- All data quality issues identified at or above the High criticality threshold defined in [Section 7.13.1](#) are subject to a documented RCA within ten (10) business days of identification. Where a data quality issue is also a logged incident, RCA is performed within the OHIP ITSM Standard's incident and problem-management process.
- Root cause findings are recorded in the data quality tracking tool designated by OHIP and include: a description of the issue, the system or process where the defect originated, the contributing factor (technical, process, or human), and the corrective or preventive action taken.
- Where RCA identifies a systemic defect in a business process or technical interface, the Contractor **MUST** submit a remediation plan to OHIP within fifteen (15) business days proposing process or system corrections to prevent recurrence. This data-governance remediation-plan commitment operates within, but is not duplicated by, the OHIP ITSM Standard's problem-management framework.

The Contractor **SHOULD** apply the Plan-Do-Check-Act (PDCA) improvement cycle when managing recurring data quality issues: planning remediation based on root cause assessment, implementing corrections, monitoring outcomes against defined thresholds, and initiating a new cycle when data quality falls below acceptable levels or when new data sets or business rules introduce fresh quality requirements. The Contractor **SHOULD** ensure data quality improvement actions are reviewed quarterly alongside the data quality rule review required under [Section 2.13](#) to assess whether implemented corrections have reduced defect rates to within acceptable thresholds.

2.13.3 Continuous Monitoring Frequency

The frequency at which data quality monitoring is performed **MUST** be defined for each monitored data domain and documented in the Data Management Plan required under [Section 2.9](#). The Contractor **MUST** determine monitoring frequency based on all four of the following factors:

- **Utilization frequency:** How often the data is accessed or consumed by downstream processes, users, or systems. Data used in daily operational processing requires more frequent monitoring than data used only in end-of-period reporting.

- **Operational importance:** The criticality of the business process or federal reporting obligation the data supports. Data underpinning T-MSIS submissions, CMS-64 reporting, PERM audits, or eligibility determinations **MUST** be monitored no less than weekly. Data supporting routine internal operations **SHOULD** be monitored no less than monthly.
- **Data volatility:** How frequently data values are expected to change. Highly volatile data, such as member enrollment status or claim adjudication results, require more frequent monitoring than stable reference data.
- **Cost and operational impact of monitoring:** The labor and compute cost of the monitoring process and any performance impact on production systems. Where monitoring costs are significant, the Contractor **MUST** document the trade-off analysis and proposed monitoring approach for OHIP approval.

Monitoring frequency decisions **MUST** be reviewed annually and updated whenever a material change occurs in data utilization patterns, system architecture, or federal reporting requirements.

For data managed within the OHIP Data Lakehouse, [Section 7.6.6.2](#) provides specific assessment frequency requirements by criticality classification that implement the monitoring principles defined in this section.

2.14 Data History

Data history governs the point-in-time preservation of business data values and is distinct from data logging¹¹ and data lineage.

The Contractor **MUST** adhere to the requirements below when data history is required:

- Capture original values as well as changes.
- Use insertion only: history records cannot be updated or deleted. Any changes or corrections **MUST** create a new history record that is date- and time-stamped. If the history **MUST** document multiple changes within a day, each version **MUST** record the time the values were preserved and the time they were replaced. The latest update date and time stamp indicates the most current record.
- When data is deleted in the SOR, versions in data history **MUST NOT** be physically deleted; instead, the data in the history **MAY** be logically deleted (e.g., a delete flag indicator **MAY** be set to designate that the record is logically deleted).
- Ensure data history is traceable to the SOR(s) regardless of the location of the SOR.

2.15 Defining and Capturing Data Requirements

The Contractor **MUST** ensure that data requirements for new and modified business capabilities are defined precisely enough to verify that solutions, interfaces, and databases satisfy OHIP's Data Standards and requirements. Business data requirements for new projects **MUST** adhere to the governance deliverables specified in the applicable lifecycle methodology (e.g., Waterfall, Agile, or Hybrid) and **MUST** include:

¹¹ Refer to the OHIP Infrastructure and Operations Domain Document, Log Management Standard for more information.

- **Detailed data requirements** (e.g., plan name, provider name, plan number, provider ID, member/client ID, MMIS ID) describing the information that the business needs. This includes data acquired, consumed, produced, or published within a business event, along with sufficient context to clarify what is being referenced. For example, "planning flag" is an insufficient requirement, whereas "planning flag for an organizational change" provides the necessary context.

The Contractor **SHOULD** ensure that:

- During the project requirements phase, data owners, data managers, data suppliers, and data consumers are identified to assist in defining and reviewing data requirements for master data, reference data, metadata, transaction data, and derived data.
- OHIP approval be obtained for business requirements that are traceable to physical implementation, including every physical database and data feed created or used by a project.
- Process data requirements describing how each process uses the information provided in the project's detailed business data requirements, including information governance, data currency needs, whether the process creates or updates data, and how long data **SHOULD** be retained to comply with business, regulatory, and legal requirements. Business events **SHOULD** include an owner, the producing and/or consuming events, and the data associated with each event.
- Data requirements **SHOULD** include master and reference data, metadata, and data quality requirements described in previous sections.

2.16 Test Data

The Contractor **MUST** ensure that:

- A Test Management Plan is submitted to OHIP prior to initiating any testing activities.
- Approval from OHIP's Bureau of Security and Bureau of Privacy and Compliance is obtained before any sensitive or production data is sent to the Contractor.¹²
- All test data activities comply with the specifications and expectations outlined in Table 9.

Table 8 – Test Data Specifications

Test Data Specifications	Contractor Requirements
Test Data Approval	MUST obtain OHIP's approval of the Test Management Plan, addressing all data specifications in this table, before extracting any production or sensitive data for testing.
Test Data Elements	MUST identify the databases, tables, and fields to be extracted, and MUST specify selection criteria and any required referential integrity between tables.
Test Data Extraction	MUST collaborate with database administrators (DBAs) to extract, transform, and load data into the appropriate test environments.

¹² For more information, refer to the OHIP Security, Privacy, and Compliance Domain Document, Security and Privacy Compliance Standard.

Test Data Transformation	SHOULD coordinate with OHIP or other Contractors whether test data transformation (cleansing, de-identifying, or other modifications) is required and MUST perform these activities as assigned.
Test Data Cleansing	SHOULD evaluate whether cleansed data is required for the tests. When realistic data quality issues are needed to validate error handling, the Contractor SHOULD ensure test data appropriately reflects those conditions.
Test Data De-Identification ¹³	MUST de-identify PII, PHI, FTI, and other sensitive data unless operating in a production environment. The Contractor MUST specify the fields requiring de-identification by documenting the methodology and rules in the Test Management Plan, use realistic substitution algorithms, and ensure methods support integration testing across solutions and Contractors.
Test Data Access	MUST designate authorized personnel who MAY handle sensitive data during test data preparation, test execution, or test result review.
Test Data Security	MUST apply full security controls to any test environments, databases, and personnel involved in handling sensitive data.
Test Dataset Tracking	MUST use an OHIP-approved tool to track all test datasets, including versions, updates, refreshes, and datasets created by other tests.
Test Data Identification in Results	MUST ensure the test results identify the specific test dataset version to be used. This enables effective defect management and supports oversight efforts related to data “fixes” required to achieve test success.

¹³ Refer to the Security, Privacy, and Compliance Domain Document, Security and Privacy Compliance Standard and the OHIP Solution Delivery Domain Document, Test Management Standard.

3.0 Data Architecture and Naming Standard

3.1 Introduction

This OHIP Data Architecture and Naming Standard establishes mandatory requirements for the architecture, design, modeling, and naming of data elements, entities, attributes, relationships, and physical database objects across OHIP MES solutions. It consolidates the data naming conventions and architecture modeling practices into a unified standard that governs how data is structured, named, and modeled from conceptual design through physical implementation.

Consistent, meaningful naming is foundational to data integration, interoperability, and governance: it ensures that data assets can be reliably understood, shared, and reused across teams, systems, and solution boundaries without ambiguity or rework. Architecture modeling provides the structural framework within which naming decisions are made, ensuring that models at every level (conceptual, logical, physical) are complete, traceable, and aligned with enterprise standards.

This standard governs naming at three distinct levels. Each level serves a different audience and carries different formatting constraints. The Contractor **MUST** ensure that all three levels are aligned for each named element and that naming conventions are enforced at every modeling stage.

Deliverables and artifacts associated with this standard can be found in [Appendix A](#).

Table 9 – Naming Levels

Level	Purpose	Audience	Format	Example
Business Name	Common term used by non-technical staff. No formal syntax required.	Business users, stakeholders	Title Case, full words, spaces as delimiters, no abbreviations	Client Street Address
Logical Name	Structured representation of the data concept, aligned to the enterprise data model.	Data architects, analysts	Title Case, full words or recognized abbreviations, spaces as delimiters	Address Line 1
Physical Name	Technical implementation in a database, constrained by DBMS limitations.	Database developers, DBAs	Uppercase or mixed case, underscores as delimiters, abbreviations where required	ADDR_LN_1

This standard applies during the design, development, integration, (DDI) and operations stages of OHIP's Solution Development Life Cycle (SDLC). It addresses:

- **General Data Architecture Principles:** Core rules governing how models are structured, assessed, and maintained across the enterprise.
- **General Object Naming Principles:** Core rules governing how names are constructed to be precise, self-descriptive, unambiguous, and aligned with business terminology.
- **Abbreviations and Acronyms:** Standards for when and how abbreviations are permitted.

- **Conceptual and Logical Naming:** Requirements for naming subject areas, entities, attributes, and relationships in enterprise data models.
- **Physical Naming:** Requirements for database schemas, tables, views, columns, and foreign keys.
- **Data Architecture Artifacts:** Conceptual, logical, physical, and canonical data models with completeness and quality requirements.

 **Note:** Business architecture artifacts (capability models, process models, use case models), application architecture artifacts (integration architecture, component models, solution architecture, technology architecture artifacts (deployment models, network topology), and security architecture artifacts have been relocated to the OHIP Solution Delivery Domain Standard, Business and Technical Architecture Standard.

 **Cross-Reference:** Requirements governing the physical database structures that implement these models are found in [Section 4.0](#). Naming conventions specific to Snowflake Lakehouse objects are addressed in [Section 7.15](#).

3.2 Objectives

The objectives of this Standard¹⁴ are to:

- Establish a consistent framework for naming data elements, reducing ambiguity and promoting clarity across all systems.
- Define mandatory architecture modeling practices that ensure data models are complete, traceable, and aligned with CMS MITA standards. Business, application, technology, and security architecture modeling practices are governed by the OHIP Solution Delivery Domain Standard.
- Improve data comprehension and usability for both technical and business stakeholders by ensuring names are meaningful, precise, and self-descriptive.
- Enhance data integration and reusability by standardizing terminology and modeling practices across the OHIP MES ecosystem.
- Provide requirements that guide stakeholders when developing data models and database designs.
- Establish quality standards for data architecture artifacts including completeness, consistency, and traceability requirements.
- Ensure all models align with federal reference architectures (MITA, FHIM, NIEM, FHIR) and support CMS certification.

¹⁴ This OHIP Data Architecture and Naming Standard is supported by the Data Management Book of Knowledge. Supporting guidance can be found in sections related to data modeling and best practices.

3.3 Applications and Exclusions

3.3.1 Applications of the Standard

This Standard governs data elements throughout all stages of solution development.¹⁵ It applies during the DDI stages, where OHIP maintains a reasonable level of influence and control over data design.

Examples of application areas include, but are not limited to:

- Data warehouses, data lakes, and other data stores (for example; OLTP).
- Databases designed and maintained by OHIP or created under the guidance or direction of OHIP by third parties.
- Data interfaces under the control of OHIP. If the interface is with a solution outside of OHIP oversight and management, the components from any OHIP middleware boundary (and inside OHIP) are expected to align with this Standard.
- Corporate data models, whether they have yet to be fully implemented or not.
- Report definitions, dashboards, curated datasets, and objects.

3.3.2 Exclusions for the Standard

This Standard is not intended to be applied to applications and data that are either outside OHIP's scope of control, are unrealistic to enforce based on OHIP's architectural review, or when legacy solutions have been implemented before the initial publish date of this standard.

Examples of exclusions include:

- COTS or SaaS Contractor products with proprietary data models. However, COTS and SaaS Contractor products are expected to adhere to reasonable naming standards.
- Contractor-based or "black box" solutions with internal data models not exposed to outside solutions, except through published APIs (e.g., software packages such as MS Office, where only the elements exposed by the published API are visible).
- Legacy solutions implemented before the effective date of the Data and Information Management Domain Document unless the effort involved in retrofitting their data model to align with this standard justifies the benefits to be achieved.

In most of the cases listed above, mappings for interfaces and migrations **MUST** adhere to the naming standards for respective operations. OHIP will evaluate Contractor products with inconsistent naming standards to weigh the benefits against the associated risks. Any instances where the standards cannot be followed **MUST** be treated as exceptions and handled accordingly. OHIP anticipates that variances where names cannot be standardized will be handled via translations. OHIP reserves the right to approve all data naming and any variances from this standard.

 **Note:** For non-relational and platform-specialized data technologies (e.g., document, graph, key-value, vector, or time-series databases; event streaming platforms and message schemas; and semi-structured formats such as JSON, Parquet, and Avro within the Data Lakehouse), the naming,

¹⁵ For more information, refer to the OHIP Solution Delivery Domain Document, SDLC Standard.

metadata, and schema governance standards defined in [Sections 3.5](#) and [3.6](#) apply to the extent supported by the target platform's architecture and object model (e.g., collections, node labels, property names, topic names). Where a platform uses fundamentally different implementation structures, the Contractor **MUST** document and obtain OHIP approval of a platform-specific approach that demonstrates equivalent consistency, traceability, semantic alignment, metadata integrity, and governance outcomes relative to the enterprise conceptual and logical naming standards prior to development. Platform-inherent implementation differences do not require a SDR, provided the Contractor demonstrates equivalent governance effectiveness and obtains OHIP approval.

 **Technology Evolution:** This standard establishes governance objectives and control expectations independent of specific technologies. Where technologies evolve beyond the platforms explicitly referenced in this document, the Contractor **MUST** apply the governance intent, traceability expectations, and quality standards defined herein using platform-appropriate implementations subject to OHIP approval. The emergence of new technologies does not exempt the Contractor from governance obligations; it requires adaptation of implementation approach while preserving governance outcomes.

3.4 General Data Objects Naming Principles and Standards

3.4.1 Data Objects Naming Principles

When organizations adopt industry terminology, the risk of conflicting with established naming standards is reduced. The Contractor **MUST** adhere to OHIP data naming conventions and standards, which include, at minimum, the following principles:

- Be unique within a reasonable scope (e.g., corporate versus worldwide) to avoid possible conflicts. Within the internal context of OHIP and its partner ecosystem, naming conventions **MUST NOT** result in conflict. Industry terminology and names can be derived from external standards such as ANSI X.12, HL7, and CMS.
 - For example, “CLIENT” would generally be understood to be within the scope of OHIP as long as it was not used by multiple organizations within OHIP with different meanings.
- Be precise, meaningful, and clear. Avoid terms that require additional explanation or clarification.
 - For example, “CLIENT” might be an ambiguous name that can refer to a patient, ecosystem partner, or client of another agency. If it is widely understood to refer to a Medicaid client of OHIP, then there might not be an issue, provided that was broadly understood.
- Be a representation of the intended business concept.
 - For example, “PHUB CTRLMS” is very unclear, understood possibly by a select group of technical staff. Names **MUST** be meaningful for a broad range of technical and non-technical staff while representing the business purpose. “CLIENT ID” is a clearer example.
- Not be based on a particular application, solution, or underlying information technology platform (e.g., “Oracle” instead of “Finance system” or “Mainframe” instead of “specific data component”).
 - For example, “ORACLE CODE” or “WINDOWS LOGIN” would be problematic if the solution were ported to a different platform.

- Be self-descriptive and self-documenting. The name **MUST** provide comprehension of what it represents without having to read the description. It **MUST** be extracted from the business usage and purpose of the element.
 - o *For example, “PATIENT ID” is clear, while abbreviations such as “P ID” are ambiguous.*
- Be unambiguous to the greatest extent possible. People from different backgrounds or business areas reading the name at different times **MUST** have the same understanding of what the name means.
 - o *For example, “CONSUMER” can be an ambiguous term. Questions may arise such as “Is this an end consumer or a business-to-business consumer?” The presence of such interpretation suggests the name is ambiguous.*
- Align with the repository where the data is stored and data modeling tools. Some level of reasonableness **MUST** be applied to naming logic.
 - o *For example, a name that is one hundred (100) characters long, combined with multiple suffixes and prefixes, could make the name unwieldy or prone to errors.*
- Not impede the implementation of software packages within the solution’s development scope of work. A lack of a name standard or a name that conflicts with an existing data element **MUST** be resolved immediately.
 - o *For example, an application data model defines “PROCEDURE ID” as a medical procedure; however, a previously designed solution uses the same name for an insurance claim’s routing procedure. Such conflicts **MUST** be addressed as soon as possible.*
- Leverage established data dictionary names and definitions where available.

The Contractor **SHOULD**:

- Use consistent naming for common fields across different databases.
- Consider business context when naming data elements.
- Align names with policy documents where applicable.
- Provide descriptive in-line and data dictionary comments for all named elements.

3.4.2 Data Objects Naming Standards

The Contractor **MUST** create data models that accurately represent business data usage and are easily readable and understandable by both business and technical reviewers. The Contractor **MUST** ensure that data model elements, such as entities, attributes, relationships, and XML schema tags, use full and descriptive English-language names that accurately convey the meaning of the corresponding business concepts.

The Contractor **MUST** ensure that data elements:

- Are meaningful and derived from the business use or purpose.
- Are drawn from commonly understood definitions to improve organization-wide understanding and consistency, if a corporate semantic model/glossary exists.
- Are a noun alone wherever possible (e.g., “AGREEMENT”).

- Do not use the name of the parent element (entity, for example) in the name unless necessary. Including the parent element name as a prefix **MUST** be limited to cases where the attribute **MUST** be referenced as a foreign key in another entity.
- Describe a standard qualifier at the end for the element where it is useful (e.g., character type [CT], numeric value [NUM], description [DESC], unique identifier name [NAME], unique identifier identity [ID], currency date [CD], indicator [IND], amount [AMT]).
- Clarify the noun meaning using adjectives sparingly (e.g., “PURCHASE AGREEMENT”); minimize adjectives when they are redundant or unnecessary (e.g., use “CONTRACTOR” instead of “CONTRACT EMPLOYEE”).
- Use singular names to reflect a single instance of the object being defined (e.g., “PATIENT” instead of “PATIENTS”), except subject areas, which **MUST** be expressed in the plural form.
- Do not use special characters (e.g., / * # \$) except an underscore to improve readability.
- Do not begin with a numeric character.
- Do not use abbreviation algorithms except in rare cases where name-size limits necessitate their use.
- Use abbreviations and short forms only where the Contractor or OHIP product naming limitations prohibit using a whole word.

To support consistent physical naming, the following class words (also called suffix types) **MUST** be used as the final term in physical data element names where they apply. Class words distinguish attributes from entities, column names from table names, and indicate whether an element is qualitative or quantitative in nature. The Contractor **MUST** use the approved class word abbreviations in the table below and **MUST NOT** create alternative abbreviations for these terms without OHIP approval.

Table 10 – General Naming Class Words

Class Word	Abbreviation	Type	Description and Usage
Account	ACCT	Qualitative	An account or record identifier (e.g., ACCT_NUM)
Amount	AMT	Quantitative	A monetary or numerical sum (e.g., CLAIM_TOT_AMT)
Code	CD	Qualitative	A coded value from a reference or lookup table (e.g., DIAG_CD)
Count	CNT	Quantitative	A numeric count of occurrences (e.g., CLAIM_LINE_CNT)
Date	DT	Qualitative	A calendar date value. MUST be stored as a DATE data type (e.g., SRVC_DT)
Description	DESC	Qualitative	A text description or label (e.g., PRVDR_TYPE_DESC)
Identifier	ID	Qualitative	An externally assigned unique identifier (e.g., MEMBER_ID).
Indicator	IND	Qualitative	A flag or yes/no indicator, typically boolean or single character (e.g., ACTV_IND)
Name	NM	Qualitative	A person, entity, or object name (e.g., PRVDR_LAST_NM)
Number	NUM	Quantitative	A numeric value that is not a count or amount (e.g., ADJSTMT_NUM)
Percent	PCT	Quantitative	A percentage value (e.g., COPAY_PCT)
Quantity	QTY	Quantitative	A non-monetary measurable quantity (e.g., SRVC_UNIT_QTY)

Timestamp	TS	Qualitative	A date and time value. MUST be stored as a TIMESTAMP data type (e.g., RCRD_UPDT_TS)
Text	TXT	Qualitative	Free-form text content exceeding a single name or label (e.g., NOTE_TXT)

Class words that are not in this table **MUST** be submitted to OHIP and the BEDG for approval before use. Once approved, they will be added to the Enterprise Glossary and this table will be updated at the next revision cycle.

3.4.3 Abbreviations and Acronyms

The Contractor **MUST**:

- Use well-documented or broadly understood abbreviations and acronyms rather than abbreviations that rely on tribal knowledge.
- Apply abbreviations consistently across models, databases, and development artifacts.
- Avoid abbreviating words fewer than eight (8) characters in length.

3.4.3.1 Rules for Abbreviating Single Word Names

The Contractor **MUST**:

- Check the Enterprise Glossary before creating any abbreviation. Industry acronyms (e.g., SSN) are well understood; however, they may still cause problems if the acronym conflicts with a technical acronym in the developed solution
- Remove all special characters (e.g., -, &, ', etc.).
- Remove all vowels, except the leading vowel or leading two vowels, depending on the circumstance.
- Replace multiple consonants with a single consonant.
- Truncate consonants after a sufficiently descriptive number of letters (e.g. "Cerebrovascular" abbreviated as "Crbrvscrl").

3.4.3.2 Rules for Abbreviating Multiple Word Names

The Contractor **MUST** apply the abbreviating rules for single-word names for each individual word in a multiple-word name.

For names with more than two words, the Contractor **MUST** use the first letter of each word, if appropriate (e.g., "Authorized Representative" abbreviated as "Auth Rep;" "Social Security Number" abbreviated as "SSN").

3.4.3.3 Other Abbreviation Considerations

The Contractor **MUST** adhere to the following requirements when abbreviating words:

- If the resulting abbreviated word conflicts with a popular acronym, avoid the acronym and create a new abbreviation (e.g., "Auth" commonly represents "Authorized"; do not use "Auth" for "Authenticated;" instead abbreviate as "Authntc").

- Leverage an OHIP-prescribed review process for abbreviations and acronyms as part of the initial data model review.

The following worked examples illustrate the application of the abbreviation rules. Contractors **MUST** use these as references when developing new abbreviations not already present in the Enterprise Glossary.

Table 11 – Application of Abbreviation Rules

Full Word	Abbreviation Process	Result
Entitlement	Remove vowels (except leading); keep first letter	ENTTLMNT
Eligibility	Remove vowels (except leading); keep first letter	LGBLTY
Provider	Remove vowels (except leading)	PRVDR
Enrollment	Remove vowels (except leading)	NRLLMNT
Grantee	Remove vowels; retain trailing “ee”	GRNTEE

The abbreviation for a word **MUST** always begin with the same first letter as the full word, even when that letter is a vowel (e.g., “Eligibility” abbreviates to “LGBLTY”; the leading “e” is retained per rule).

Where two independently derived abbreviations produce the same result, the Contractor **MUST** create a distinguishing variant and document it in the Enterprise Glossary for OHIP approval.

3.5 Conceptual and Logical Naming Standards

Conceptual and logical data models focus on capturing business-level entities and attributes without considering the limitations and constraints of physical implementation. However, physical models **MUST** inherit naming from the logical models to the greatest extent possible. ⁴

Therefore, the Contractor **MUST**:

- Adhere to business-focused naming policies at the conceptual and logical data model levels.
- Ensure the names used to define entities are unique within the enterprise.
- Leverage any existing, applicable names from the Enterprise Data Model (EDM) and Enterprise Glossary when developing new conceptual and logical models. ⁵
- Resort to abbreviations/acronyms only in cases well understood in the industry (e.g., SSN, W-2) but continue to spell out names fully where appropriate.

3.5.1 Subject Areas and Entities

The Contractor **MUST**:

- Name subject areas and entities using the general naming standards provided in this document.
- Conform to ISO/IEC 11179-5:2015.¹⁶

¹⁶ Refer to [ISO/IEC 11179-5:2015](#), Information technology—Metadata registries (MDR)—Part 5: Naming principles, International Organization for Standardization / International Electrotechnical Commission, 2015.

- Develop entity names that are unique across all OHIP models. If a chosen name already exists, the two entities **MUST** be combined, or one **MUST** be renamed where OHIP has control over the models.
- Use business terminology. Where different groups of staff have different terminology, select the more widely used or accepted term and include any synonyms in the entity definition.
- Create meaningful names. The name **MUST** reflect the thing or concept that the entity represents rather than any medium or method in which it is handled.
- Only use acronyms or abbreviations that are fully recognizable by the business communities.
- Name entities using singular nouns (e.g., “Provider” or “Organization”) and name subject areas using the plural form.
- Use mixed case or uppercase characters. If using mixed case, capitalize the first letter of each word (e.g., “Status”).
- Use only an underscore or space as a delimiter between words (e.g., “Provider Name”).
- Use only alphabetic characters and an underscore. Numbers are discouraged; include them only when they add essential clarity (e.g., “401K Eligibility,” “AddressLine1, AddressLine2”).

Entity names **MUST NOT** begin with a numeric character.

3.5.2 Attributes

The Contractor **MUST**:

- Comply with the general naming standards provided in [Section 3.4](#).
- Use acronyms or abbreviations that comply with the standards in [Section 3.4.3](#).
- Ensure attribute names are unique within their parent entity.
- Use business terminology and be aligned to an EDM model and/or glossary.
- Not use the name of the parent entity in the attribute name. The table name may be repeated in the column name for foreign keys.
- Ensure attribute names contain only singular nouns (e.g., “First Name” and “Last Name”).
- Use mixed case or uppercase characters for attribute names. If using mixed case, capitalize the first letter of each word (e.g., “First Name” and “Last Name”).
- Use a space as the delimiter between words (e.g., “First Name,” “Last Name,” and “Birth Date”) for attribute names.
- Use only alphabetic characters, spaces, and underscores for attribute names. Numbers are discouraged; include them only when they add essential clarity.

3.5.3 Relationships

The relationship between two entities is always directional and sometimes bi-directional. The Contractor **MUST** adhere to the following requirements when naming relationships:

- Name both directions of each relationship separately.
- Use a clear, concise phrase that is meaningful to the business user.

- Use meaningful names that convey the business rule that exists between the entities. Names such as “related to” or “associated with” do not provide enough meaning (e.g., use “Client has Diagnosis” or “Status describes Claims”).
- Name the active relationship direction first. For most relationships, there is an active and a passive direction (e.g., “one physician may prescribe one or more treatments” is active; “one treatment must be administered by one physician” is passive).
- Use clear, concise phrases.
- Use lowercase characters (e.g., “belongs to,” “has,” and “describes”) and a space as the delimiter between words.
- Use only alphabetic characters, spaces, and underscores. Numbers are discouraged; include them only when they add essential clarity.
- Stay within character limits. Relationships **MUST NOT** be longer than thirty (30) characters in length.

 **Note:** Naming supertype/subtype relationships is optional and **MAY** not be necessary in all cases.

3.6 Physical Naming Standards

Physical data models translate logical designs into database-specific implementations and **MAY** introduce objects (e.g., denormalized tables, materialized views) that do not appear in the logical model. Physical object names are subject to the naming limitations imposed by the database platform. The Contractor **MUST** maximize name clarity within those constraints.

3.6.1 Database Schemas

The Contractor **MUST**:

- Name schemas using the general data naming standards provided in this document.
- Be descriptive and meaningful, reflecting the purpose or concept of the data contained within the schema.
- Avoid reserved keywords or special characters not supported in database solutions.
- Use a consistent naming convention to make schemas easily identifiable and organized, including using prefixes.
- Use singular or plural form for schema names and follow the chosen convention consistently throughout the solution.
- Keep names reasonably short and concise, ensuring they do not exceed the limitations imposed by the database management solution.
- Use alphanumeric characters along with underscores.
- Avoid spaces, special characters, or punctuation marks that complicate schema referencing.

3.6.2 Tables

The Contractor **MUST**:

- Name tables using the general naming standards provided in [Section 3.4](#).

- Leverage the data modeling tools to derive physical names from logical names, using the glossary of abbreviations when possible.
- Use unique naming within the database schema. Enterprise data **MUST** be unique within OHIP.
- Use a unique, short name. The short name **MUST** be used with the physical naming standards to name database objects such as foreign keys, packages, and triggers.
- Repurpose the logical (entity/class) names, subject to DBMS naming limitations (e.g., length and reserved words).
- Use singular form in uppercase or mixed case, depending on the capabilities of the database management software used. If using mixed case, capitalize the first letter of each word (e.g., EligibilityStatus).
- Be aware of the database modeling standard features used when naming tables so that special prefixes can be used without changing the table name.

 **Note:** Naming supertype/subtype constraints is optional.

3.6.2.1 Views/Materialized Views

The Contractor **MUST**:

- Be descriptive, reflecting the purpose, content, or functionality of the view or materialized view.
- Include a suffix to indicate views or materialized views when necessary (e.g., “vw” or “view” for views and “mv” or “mat view” for materialized views).
- Use consistent naming conventions to ensure views and materialized views have a similar naming pattern.

The Contractor **MUST** also comply with the following requirements for custom view naming:

- Where a view draws from a single source table, the view name **MUST** be derived from that table name using the approved suffix (e.g., a view of the MEMBER table is named MEMBER_V).
- Where a view draws from multiple tables or uses complex joins and filters, the view name **MUST** reflect the combination of primary source tables to make the data sources identifiable from the name alone. Table names **MAY** be abbreviated to accommodate DBMS length restrictions, but the combination **MUST** remain intelligible (e.g., a view joining CLAIM and MEMBER tables is named CLAIM_MEMBER_V).
- Generic or non-descriptive view names such as RPT_V, TEMP_V, or VIEW1 are not permitted. Each view name **MUST** convey its business purpose or source data without requiring reference to external documentation.
- Where public synonyms are created for views, the synonym **MUST** use the exact view name, including its suffix.

3.6.3 Columns

The Contractor **MUST**:

- Name columns using the general naming standards provided in [Section 3.4](#).
- Use names that are unique and singular within the parent table.

- Derive column names from the attribute name, using the glossary of abbreviations, subject to DBMS limitations when the logical data model is present.
- Show the proper separation of multiple words, using an underscore delimiter (e.g., DATE_OF_BIRTH).

3.6.3.1 Foreign Key Column Names

When a column references the primary key of another table, the column is a foreign key. The Contractor **MUST** comply with the following requirements when naming foreign key columns:

- Use the same name and data type as the column in the referenced table unless multiple relationships exist between the two entities or the column flows to the dependent table via different relationship paths.
- Define any foreign key constraint in the column to ensure data integrity.
- Use the following column name format when the foreign key is a system-generated primary key from the parent table (i.e., “ID”): (parent table name)_ID (e.g., ORGANIZATION_ID, PERSON_ID, FACILITY_ID, LOCATION_ID).
- Resolve duplicate column names by role naming when there are multiple foreign key relationships between two tables.
- Derive the column name from the attribute name.

3.6.3.2 Surrogate Key Naming

Surrogate keys and system-generated identifiers **MUST** follow consistent naming conventions to distinguish them from externally assigned identifiers and to communicate their role in the data model. The Contractor **MUST** adhere to the following conventions when naming surrogate keys and related identifier columns:

Table 12 – Surrogate Key Naming Class Words

Class Word	Physical Suffix	When to Use	Example
Identifier	ID	An identifier assigned by an external source or authority (e.g., NPI, SSN, Medicaid ID). Not system-generated.	MEMBER_ID, PRVDR_NPI_ID
Key	KEY	A surrogate primary key that is system-generated and has no business meaning. Used in OLTP tables.	CLAIM_KEY, ENRLMNT_KEY
History Key	HIST_KEY	A surrogate key assigned to a slowly changing dimension record in a data warehouse or Lakehouse environment. Enables point-in-time lookup.	MEMBER_HIST_KEY, PRVDR_HIST_KEY
Dimension Key	DIM_KEY	A surrogate key assigned within an OLAP data mart for a dimension table. Distinct from the history key, which operates at the warehouse layer.	MEMBER_DIM_KEY, PLAN_DIM_KEY

- Surrogate key columns **MUST NOT** carry business meaning and **MUST NOT** be exposed in application-facing interfaces or reports unless they serve as a stable reference for downstream systems with OHIP approval.
- Where a table contains both a surrogate key (KEY) and an externally assigned identifier (ID), both **MUST** be present and distinctly named. The surrogate key serves as the physical primary key; the external identifier **MUST** have a unique constraint applied at the database layer.
- History keys and dimension keys **MUST NOT** be used interchangeably. History keys support time-series tracking at the enterprise data warehouse (EDW) layer; dimension keys are mart-specific and **SHOULD** not appear in warehouse-layer tables.

 **Cross-Reference:** The naming conventions defined in [Sections 3.4 through 3.6](#) govern how data elements are named at each modeling level. The following sections ([3.7](#) through [3.12](#)) define the architecture modeling standards that govern the structure, content, quality, and completeness of the models themselves. The Contractor **MUST** apply the naming conventions defined above when creating all architecture artifacts below.

3.7 General Modeling Standards

The Contractor **MUST** maintain architecture models that align with MITA 3.0¹⁷, TOGAF¹⁸, and ArchiMate 3.1 standards¹⁹. All models **MUST** be stored in an OHIP-approved enterprise architecture repository and comply with the naming conventions defined in [Section 3.4](#).

3.7.1 Assessment

The Contractor **MUST** develop architectural models that allow OHIP to assess a solution's compliance with:

- OHIP's business requirements (including its enterprise architecture strategic vision and objectives), standards, and policies.
- CMS requirements, such as MITA and CMS SMC requirements.
- Any new or complex integrations or entities and uncertain or expensive aspects of a solution.

3.7.2 Modeling Tools

OHIP will provide the Contractor with the necessary repository access for the project. The Contractor **MUST**:

- Develop and maintain all architecture models (including conceptual data models) in OHIP's Enterprise Architecture Repository.
- Ensure all models are kept current.

¹⁷ For more information, refer to the Centers for Medicare & Medicaid Services, [Medicaid Information Technology Architecture \(MITA\) Framework 3.0](#), released March 28, 2012.

¹⁸ For more information, refer to [The Open Group, TOGAF® Standard](#) (Version 9.2 or 10th Edition—confirm which your program has adopted).

¹⁹ For more information, refer to [The Open Group, ArchiMate® 3.1 Specification](#), November 2019.

- Register all modeled and implemented APIs in OHIP's API catalog.²⁰
- Capture the following metadata elements, at a minimum, for all systems, applications, and components stored in the OHIP-designated repository:
 - o Unique identifier
 - o Name
 - o Description
 - o Architecture domain/category
 - o Operational support (e.g., the Contractor, ITS, etc.)
 - o System owner (e.g., DOH, OTDA, DOL, etc.)
 - o Core system as a component of (e.g., eMedNY, NYSOH, MDW, MECM, AMS, PAI, etc.)
 - o Data sensitivity classification
 - o Criticality rating
 - o Current disposition and target state (e.g., contain, core, emerging, retire)
- Develop logical and physical data models, as applicable, in a format compatible with OHIP's designated data modelling tool.

3.7.3 Reference Architecture Models

The Contractor **MUST**:

- Evaluate existing Authoritative Reference Models for the domain under development, incorporating where appropriate to minimize the creation of custom models and promote consistency across implementations.
- Adopt prescribed models for the applicable project where federal funding is involved, and the federal government mandates the use of specific reference architectures and models.
- Align their modeling efforts with MITA guidance and utilize the standard models identified within the MITA framework.

3.7.4 Model Maturity and Patterns

The Contractor **MUST**:

- Collaborate with OHIP to select a relevant maturity model and associated modeling patterns before beginning any modeling activity. Patterns **MUST** provide cloud-native and microservices guidance.
- Apply consistent entity, attribute, component, and interface naming for all models in accordance with the OHIP Data Architecture and Naming Standard and applicable federal conventions.
- Adhere to the MITA Maturity Model (MMM) and its associated capability matrices.

²⁰ For more information, refer to the OHIP Infrastructure and Operations Domain Standard, Application Integration Standard.

3.7.5 Model Quality Standards

The Contractor **MUST**:

- Conduct a formal review process for all models.
- Submit a report of the model's compliance with the standards, including:
 - o Peer review by at least one architect or analyst.
 - o Validation against use cases, rules, and source documentation.
 - o Approval from business owners (for business-facing models).
 - o Storage in the designated model repository with proper metadata.
 - o Validation of the quality dimensions described in Table 14.
- Submit all models to the Architecture Review Board (ARB) for review and approval.

Table 13 – Model Quality Validation Criteria

Quality Dimension	Validation Criteria
Completeness	All required elements (e.g., actors, flows, entities, and data stores) are present and defined.
Consistency	Terminology, naming conventions, and structure are consistent across models and domains.
Correctness	Models accurately reflect documented requirements and align with business capabilities.
Traceability	Model elements map to source documents (e.g., use cases, capabilities, and rules).
Standards Compliance	Models use approved notations (e.g., Business Process Model and Notation [BPMN] 2.0, Unified Modeling Language [UML], and Entity-Relationship Diagram [ERD]) and comply with enterprise conventions.
Clarity	Models are easy to understand by both technical and business stakeholders.
Modularity	Complex models are broken into logically grouped, reusable components.
Validation	Models have been peer-reviewed and validated with domain experts.
Change Control	Version control is applied; model changes are documented and approved.
Tooling Compatibility	Models are created using enterprise-approved modeling tools and exported in reusable formats (e.g., XML and XMI).

3.7.6 Architecture Description

For each model, the Contractor **MUST** develop an accompanying architecture description which:

- Includes stakeholders, concerns, rationale, correspondence, mapping between the models (views), element descriptions, variability, constraints, directory (catalog), etc.

- Complies with ISO/IEC/IEEE 42010:2022²¹ and MITA.

3.7.7 Architecture Repository and Catalog

The Contractor **MUST**:

- Maintain all Enterprise Architecture in OHIP's Enterprise Architecture Repository.²²
- Document all services and architecture elements within the proper solution context in accordance with the applicable user guides.
- Not duplicate components in the Architecture Repository and Catalog unless explicitly justified.
- Extend solutions from the prescribed reference models.
- Document rationale explaining why reuse was not possible for every new element.
- Ensure components are traceable to business drivers and no orphaned elements exist.
- Verify that architecture metadata is complete, well-formed, and structured to ensure assets are discoverable and reusable.
- Minimize service sprawl and reduce integration debt through appropriate reuse of existing services.
- Synchronize all architectural views to consistently represent the same authoritative content.
- Document exceptions, as well as their justification and OHIP approval.
- Identify and contribute reusable components of new solutions back to the enterprise knowledge base.
- Capture the service metadata prescribed by MITA as part of the Business Service Definition Package²³ and Technical Service Definition Package.²⁴

3.8 Data Architecture Artifacts

When creating Data Architecture Artifacts, the Contractor **MUST** apply the naming conventions defined in [Section 3.4](#), [Section 3.5](#), and [Section 3.6](#) at every modeling level. All entity, attribute, and relationship names **MUST** be consistent across the Conceptual, Logical, and Physical Data Models.

3.8.1 Data Model Completeness Requirements

The Contractor **MUST**:

- Reverse engineer existing databases for applications without existing data models prior to any database updates or modifications.
- Include all tables in the model, including unused tables present in the database, with their detailed metadata.

²¹ Refer to [42010-2022 - IEEE/ISO/IEC International Standard for Software, Systems and Enterprise](#).

²² Refer to the OHIP Infrastructure and Operations Domain Document, Preferred Operational Tools Standard for more information.

²³ For more information, refer to [Appendix C](#).

²⁴ Refer to [Medicaid CHIP Part III Zip File](#) for more guidance.

- Include subject areas in the model, color-coded for focused review of data model sections.
- Document all unique, foreign, and primary key constraints showing relationships between tables and fields in all data models.
- Create a single comprehensive diagram including all tables with connecting relationship lines in a foundationally sound layout.
- Include other database objects such as indexes and views in the data model if the model generates Data Definition Language (DDL).
- Use the data model to generate data dictionary information for efficient tracking and management of all metadata.

3.8.2 Conceptual Data Model (CDM)

A CDM is a type of data model that represents a high-level view of the data requirements for a system or organization. The purpose of a CDM is to define the entities, relationships, and constraints needed to describe the data needed for the organization or the system. It is not concerned with implementation details. This model is a technology-independent representation of the data requirements needed by a particular organization or system and is not tied to any specific hardware, software, or platform. It is a representation of business domains and their relationships.

The CDM is elaborated into a Logical and Physical Data Model.

Table 14 – Conceptual Data Model

Solution Design	Data Architecture
Format	Diagram (entity-relationship notation or UML class diagram) and Data Dictionary
Applicable Phase	Requirements Review and Design Phase
Applicable Deliverable	Conceptual Data Model and Dictionary

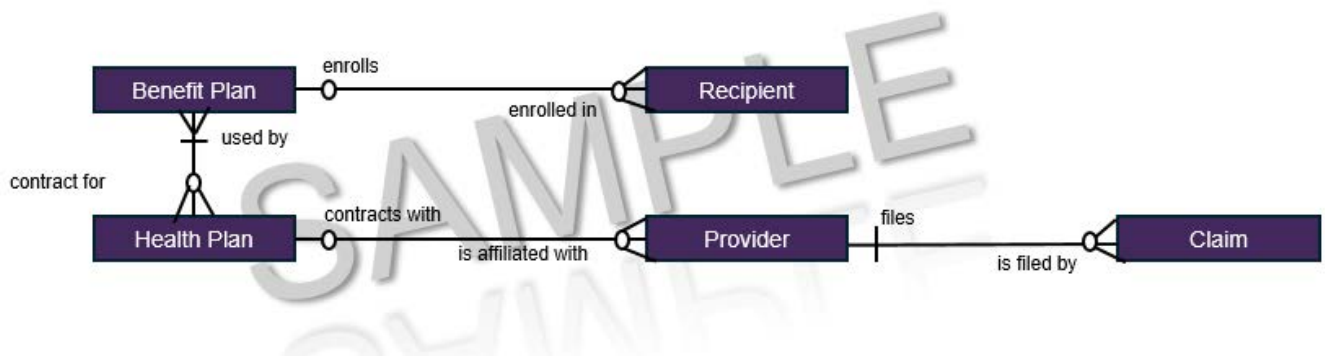


Figure 1 - Sample Conceptual Data Model

A CDM typically includes the following data elements:

- **Entities:** These represent the main objects or concepts about which data is collected. Entities are typically nouns, such as "Customer," "Product," or "Order."
- **Attributes:** Attributes are the properties or details of an entity. For example, a "Customer" entity might have attributes, such as "Customer ID," "Name," "Email," and "Phone Number."

- Relationships: These describe how entities are related to one another. For example, a relationship might describe that a "Customer" places an "Order," indicating an association between the Customer entity and the Order entity.

When developing a CDM, the Contractor **MUST** adhere to the following guidelines:

- Cover all the information in scope for the solution, implementing one or more databases.
- Provide a one-page summary diagram.
- Display large CDMs with multiple detailed diagrams.
- Prioritize readability.
- Use plural forms for subject area names and singular forms for entity names.²⁵
- Use normalization principles as a guide to group information into meaningful entities.
- Prioritize generalization for clarity but avoid generalization that is too abstract for business understanding.
- Provide relationship names (verb phrases, such as benefit plan enrolls recipient or benefit plan is used by health provider).
- Show one-to-one, one-to-many, or many-to-many relationship cardinalities.
- Avoid showing relationship optionality as this detail **MAY** make the notation less business friendly.
- Create the diagram in a well-known Entity Relationship notation, such as Information Engineering or Oracle/Barker, which uses the familiar "crow's feet" relationship line ends.
 - o A UML-class diagram is also acceptable.
- Provide a Data Dictionary containing:
 - o Subject Area/Entity Name
 - o Subject Area/Entity Definition
 - o Data Sensitivity (e.g., an entity that contains Personal Health Information [PHI])
 - o Any representative attributes that have been named

3.8.3 Logical Data Model (LDM)

An LDM is a type of data model that represents the data requirements for a specific system or application. The purpose of an LDM is to describe the data requirements and relationships needed for an application to function. This model type is technology-independent and platform-independent, meaning it is not tied to a specific hardware or software platform. It represents a high-level data view and is not concerned with the physical implementation details. This model includes entities, attributes, and relationships between entities. It defines the structure of the data but not the way the data is stored or accessed.

Table 15 – Logical Data Model

²⁵ Refer to [Section 3.0](#) for additional guidance on naming standards for models.

Solution Design	Data Architecture
Model Type and Notation	Diagrams (entity-relationship notation, such as Information Engineering or Oracle/Barker, which use the familiar “crow’s feet” relationship line ends OR UML class diagram) and Data Dictionary.
Applicable Phase	Requirements Review and Design Phase
Applicable Deliverable	Logical Data Model and Dictionary

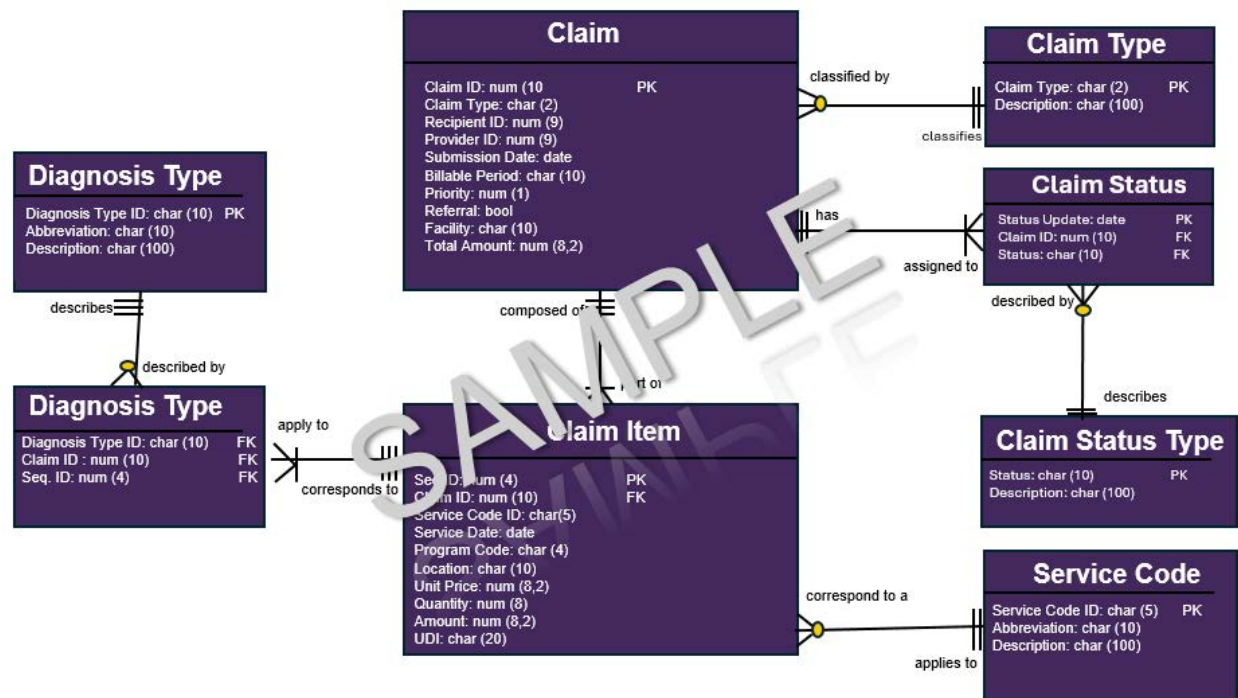


Figure 2 - Sample Logical Data Model

When developing a LDM, the Contractor **MUST**:

- Provide a Data Dictionary.²⁶
- Follow the guidance for naming, notation, and data dictionaries if modeling an analytical database (e.g., a data warehouse, etc.).
- Ensure the LDM is traceable.
- Group LDM entities into the same subject areas as in the CDM. One CDM entity will often be transformed into multiple LDM entities when the attributes are detailed and structured according to the approved modeling methodology (e.g., normalized for transactional systems, hub-link-satellite for Data Vault, or dimensional for analytical systems), defining tables, columns, and

²⁶ For information on the minimum required properties of the Data Dictionary, refer to [Section 6.0](#).

relationships at a level of detail sufficient for physical implementation. The chosen methodology **MUST** be documented and approved by OHIP before logical modeling begins.

- Assign a unique ID to each LDM entity (i.e., a primary key).
- Ensure all relationships follow [Section 4.0](#).

3.8.4 Physical Data Model

The Physical Data Model (PDM)²⁷ shows the physical layout of data in a database as tables, fields (columns), and relationships (foreign keys) accompanied by definitions and specifications for the database, tables, fields, domains, and indexes.

The PDM artifact has four parts:

- Database Descriptors
- Diagrams
- Data Dictionary
- Specifications, such as indexes and partitioning (only required for custom-developed databases, not for externally managed databases)

Table 16 – Physical Data Model

Solution Design	Data Architecture
Model Type and Notation	Diagrams (entity-relationship notation, such as Information Engineering or Oracle/Barker, which use the familiar “crow’s feet” relationship line ends OR UML class diagram) and Data Dictionary and specification charts.
Applicable Phase	Development, Implementation, and Maintenance and Operations (M&O) Phase
Applicable Deliverable	Physical Data Model and Dictionary

Figure 3 provides an example of a PDM diagram, which does not represent current OHIP architecture. Within each table (yellow box), the fields are listed with their data type and length in green. “P” indicates a primary key, and “F” indicates a foreign key. The diagram is one of many acceptable variations on entity-relationship notation.

²⁷ This artifact can also be referred to as Data Layout or as the TOGAF artifact Physical Data Model and Dictionary.

- Ensure data element names reflect the element's meaning.
- Use the approved abbreviated terms and construction rules defined in CMS guides if the database is custom designed for OHIP.
- Comply with the DM OP-045 Operating Procedure for Constructing Physical Table, Column, Element and File Names.²⁸
- Structure the PDM so it is possible to implement it in the chosen technology.
- Resolve subtypes and optimize the Logical Data Model as required for the performance characteristics of the target platform, including denormalization for relational databases, document structuring for document stores, or graph traversal optimization for graph databases.
- Note and manage any redundancy that is introduced by denormalization.

Table 18 – Sections and Necessary Properties of the Physical Data Model

Section of PDM	Usage	Object	Property of Object
Database Descriptors	<p>The following information about the whole database relates to the technology architecture and is used to update the Data Landscape.</p> <p>Specify the descriptors in the “Property of Object” column for each database in the solution, whether managed by OHIP, the Contractor, or a third party.</p>	Database	<ul style="list-style-type: none"> • Name • Description • Data owner (organization name) • Database manager (organization name) • Types of sensitive data (PHI, PII, FTI, etc., or “none”) • Schema name in each environment • Volumetrics required for costing (e.g., initial total data size, annual growth rate)
		Database	<ul style="list-style-type: none"> • Hosting provider • Location: IP address (Internet Protocol), URL (/Uniform Resource Locator), (IP/URL) or File name and path • Data structure (relational, XML, columnar, etc.) • Database engine (e.g., Oracle)

²⁸ For more information, refer to the [Data Management Operating Procedures and Guidelines](#).

Diagrams	<p>Depending on complexity, the PDM MAY be displayed on one page or multiple diagrams.</p> <p>The PDM diagram MUST be in an Entity Relationship notation, such as Information Engineering or Oracle/Barker, which use the familiar "crow's feet" relationship line ends. A UML Class diagram is also an acceptable notation.</p> <p>The diagram MAY include other metadata listed below, repeating properties from the Data Dictionary or the Specifications. For example, datatype and length MAY be shown on the diagram.</p> <p>If the PDM is generated by reverse engineering from the database, ensure it is human-readable. Relationship lines MAY need to be added to the diagram if the database does not have referential integrity.</p>	Diagram	<ul style="list-style-type: none"> • Table names • Field (column) names • Indication of mandatory and optional fields • Indication of primary key fields • Relationship lines between tables, with cardinality and optionality • Indication of foreign key relationships (shown as migrated keys in the field lists, on the relationship line, or both) • Indication of tables that are inactive or not used for OHIP • A notation legend or reference to a standard notation • Other metadata (repeated from the Data Dictionary or Specifications)
Physical Data Dictionary	<p>All properties MUST be stored in a modeling tool (see Section 3.7) with the capability of automatically extracting a data dictionary.</p> <p>If there is a Logical Data Dictionary that provides the business meaning of attributes, and the physical fields are traceable to the logical attributes, then the physical data dictionary does not need to include field definitions. The other properties listed above are still required to access and manage the data.</p> <p>If a database includes inactive or unused tables for OHIP, do</p>	Table	<ul style="list-style-type: none"> • Name • Definition • Security Classification • Schema and/or domain name if available • Archiving/purge settings • Volumetrics (initial or current size and annual growth rate) • Authoritative source table(s) or file(s) (Non-RDBS data sources can include table or entity) • Name of the entity in the Logical Data Model or Conceptual Data Model that is implemented by this table (traceability)

	<p>not detail them in the Data Dictionary. To prevent confusion, provide a list of the inactive table names that will be visible to OHIP.</p> <p>If the PDM is reverse-engineered from the database, ensure the data dictionary includes all the metadata listed above.</p>		<ul style="list-style-type: none"> • Sensitive Data Flag: Indicator if any sensitive data (PHI, PII, FTI, MCD) is stored in the table. Required for compliance with the OHIP Security and Privacy Compliance Standard. • Obsolete Flag: Indicator if the table is present in the database but no longer actively used. Required to support maintenance and cleanup activities per Section 4.0.
		Field	<ul style="list-style-type: none"> • Name • Definition, including: Derivations and/or calculations, Code value meanings and reference table lineage, Indication if this field is a key in another database, requiring integration • Primary Key flag • Foreign Key flag • Authoritative source field(s) or file(s) • Optionality (Null/not null rule) • Data Type (using the datatypes provided by the database engine) • Length (where applicable) • Precision (where applicable) • Data sensitivity (e.g. contains Personal Health Information) • Name of the attribute in the Logical Data Model that is implemented by this field (traceability), if available
		Database	<ul style="list-style-type: none"> • Database hosting location

Database Specifications	The following physical data specifications are needed if OHIP is developing or managing the database. These specifications for relational databases MAY be adapted for other technologies.		<ul style="list-style-type: none"> Database partitions
		Varies	<ul style="list-style-type: none"> Access Rights At the database, table, field, or row level, as required Other security requirements Volumetrics Current size and growth rate for: Database, Tables, Rows, Indexes
		Table	<ul style="list-style-type: none"> Table alias Table partitions Managed redundancy Referential integrity constraints History of changes to table or fields
		Views	<ul style="list-style-type: none"> Source table(s) and fields Selection criteria Purpose (access security, performance, etc.)
		Stored Procedures	<ul style="list-style-type: none"> Name Description
		Domains	<ul style="list-style-type: none"> Data Type, Length, and Precision Allowable Values Default Values Edit Mask (e.g., date or area code)
		Indexes	<ul style="list-style-type: none"> Index identification Type of Key identification Field Level Constraints

Additionally, the following physical aspects of a data platform apply:

- Storage Parameters

- o The database system **MUST** be efficiently designed to accommodate the expected data load while allowing for future scalability. This includes:
 - Capacity Planning: Estimate the current data needs and forecast future growth to ensure sufficient storage capacity. Utilize industry tools for monitoring and predicting data growth, adjusting storage allocations accordingly.
 - Performance Optimization: Implement best practices to arrange data storage to maximize read/write performance.
 - Data Backup and Recovery: Establish a robust backup and recovery strategy, ensuring regular backups are automated and stored securely, considering geographic redundancy to protect against localized data loss incidents.
- Table Partitioning Details
 - o Table partitioning is a critical database design technique used to manage large datasets efficiently. Partitioning **MUST** focus on improving query performance and enabling easier maintenance:
 - o Partitioning Strategy: Choose an appropriate partitioning strategy (e.g., range, list, hash) based on the data access patterns and the specific use case of the database. Common strategies include:
 - Range Partitioning: Useful for time-series data, where partitions are created based on date ranges.
 - List Partitioning: Suitable for categorical data, where each partition corresponds to a list of discrete values.
 - Hash Partitioning: Provides uniform distribution of data across partitions, useful for load-balancing and parallel processing.
 - Automation and Maintenance: Automate the creation and management of partitions to reduce manual overhead. Regularly review and optimize partitioning to reflect changing data patterns.
 - Monitoring and Optimization: Implement monitoring tools to analyze partition performance and identify hotspots or imbalances. Optimize partitions periodically based on workload diagnostics.
- Encryption Levels and Methods
 - o Data encryption is vital to maintain confidentiality and compliance with regulations such as HIPAA. Encryption **MUST** include:
 - In-Transit Encryption: Ensure all data transmitted across networks is encrypted using protocols, such as Transport Layer Security (TLS). This protects data from interception during transmission between clients and servers.
 - At-Rest Encryption: Encrypt data stored in the database using AES (Advanced Encryption Standard) with a key length of at least 256 bits. This ensures that data is secure even if physical storage media is accessed.
 - Encryption Key Management: Implement a secure key management system adhering to industry standards, such as NIST SP 800-57. This includes policies for key generation, distribution, rotation, and retirement.

- Compliance and Audit: Regularly audit encryption practices to ensure compliance with relevant regulations. Implement logging and monitoring to detect and respond to unauthorized data access.
- Materialized views
 - o Materialized views are an advanced database feature to improve query performance by storing precomputed results. When defining the requirements for database platform materialized views, several factors **MUST** be applied:
 - Compatibility and Support: Ensure that the database platform supports materialized views. This includes understanding the specific features and limitations associated with materialized views on that platform.
 - Refresh Mechanism:
 - Support for various refresh strategies, such as:
 - On demand (manual refresh)
 - Periodic (scheduled refresh)
 - Fast refresh (incremental updates)
 - Complete refresh
 - Automatic and manual refresh capabilities **MUST** be available.
 - Performance:
 - Efficient computation and storage mechanism to ensure materialized views improve query performance without disproportionate overhead.
 - Optimization techniques for materialized views, such as indexing and partitioning options.
 - Storage Factors:
 - Efficient use of storage, including compression options
 - Ability to handle large datasets
 - Consistency and Data Accuracy
 - Mechanisms to ensure data consistency between the materialized view and the source tables
 - Support for transaction consistency

When custom-designing a database, the PDM translates business terms and concepts from the Logical Data Model into a platform-specific data structure with the technical details. For relational databases, the PDM **MUST** be similar to the LDM but adapted for the selected database engine. For non-relational targets (e.g., document stores, graph databases, key-value stores) or semi-structured data layers within the Data Lakehouse (e.g., JSON, Parquet, Avro), the Contractor **MUST** produce a physical data model appropriate to the target technology that demonstrates equivalent completeness, traceability to the LDM, and compliance with the naming conventions defined in [Section 3.6](#). The full PDM, including diagrams, data dictionary, database descriptors, and specifications, is needed for the Solution Design Review and for OHIP to manage the implemented database.

3.8.5 Canonical Data Model

A Canonical Data Model defines a unified, Contractor-neutral representation of enterprise data that supports consistent information exchange across the OHIP Medicaid Enterprise. It enables interoperability by establishing a common vocabulary of entities, attributes, and semantics that all systems use when sharing, transforming, or integrating data. A Canonical Data Model ensures that disparate solutions communicate using a standardized set of data definitions, even when their internal data structures differ.

The Canonical Data Model also serves as the foundation for data governance, data quality, API design, and API interface specification. The Canonical Data Model **MUST** remain aligned with federal reference models and industry standards to facilitate reuse, minimize duplication, and support MITA maturity advancement.

Table 19 – Canonical Data Model

Solution Design	Information Architecture
Format	Entity–Relationship (ER) Model, UML Class Model, ArchiMate 3.1 Information Structures or Swagger API View
Applicable Phase	Design and Implementation Phase
Applicable Deliverable	Enterprise Data Models, Interface Specifications, and Solution Design Documents

Figure 4 below provides a representative example of a conceptual canonical data model illustrating how common entities and their shared attributes are normalized and reused across systems. This sample is illustrative and does not represent the current OHIP Medicaid Enterprise Canonical Data Model.

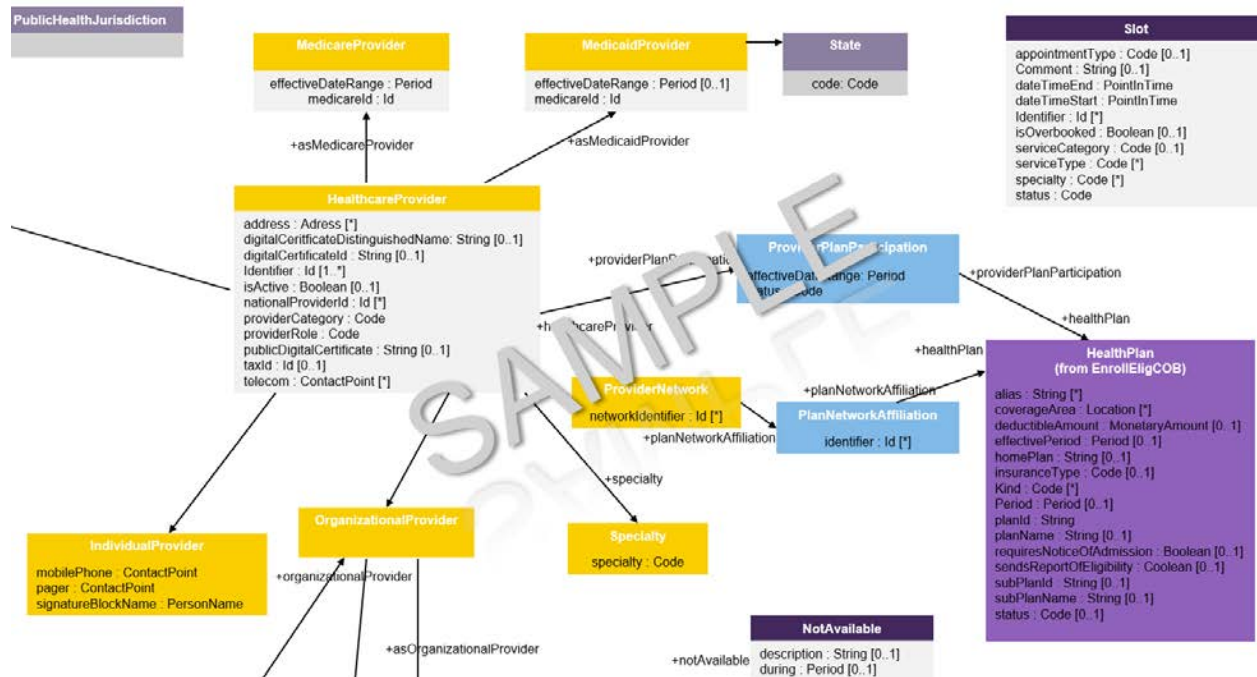


Figure 4 - Sample Canonical Data Model

For implementation, the model **MUST** be represented in a graphical form easily read by humans and implemented in API clients of the interfaces. See Figure 5 for a Swagger example.

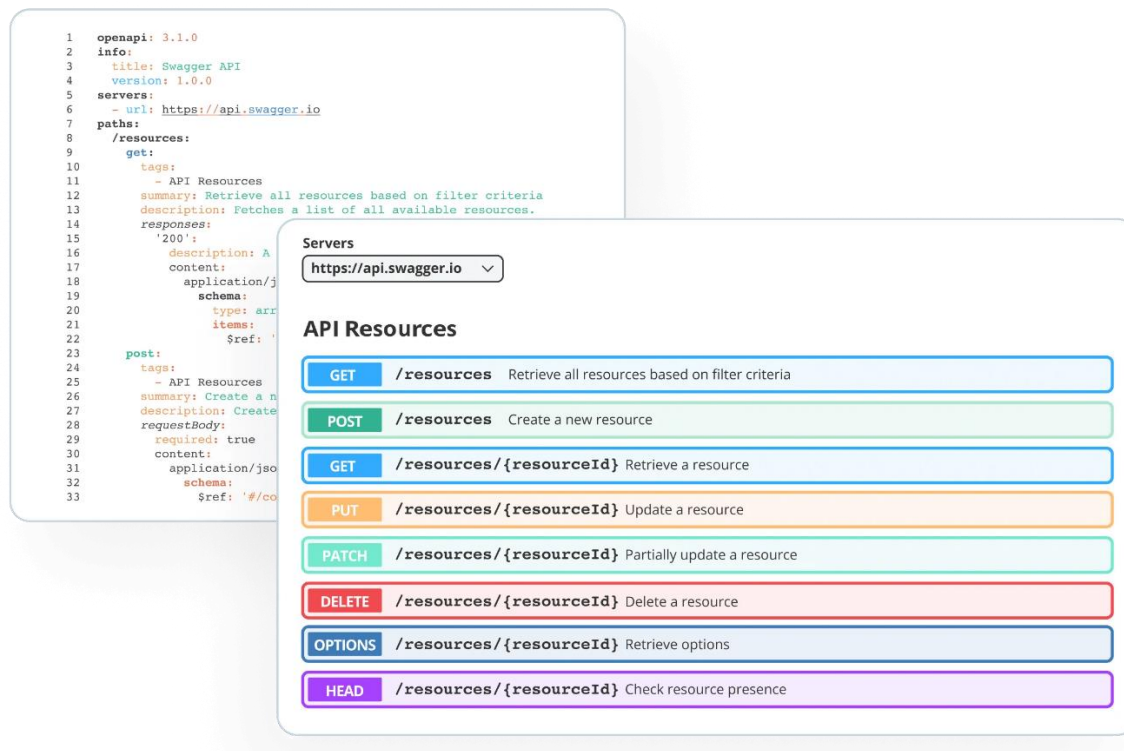


Figure 5 - Sample Canonical Data Model in Swagger

Canonical Data Model elements define the enterprise-level structures that establish a consistent meaning for all shared data exchanges. Key data elements include:

- **Entity:** A business object represented as a defined, governed concept (e.g., Person, Provider, Claim, Eligibility, Authorization). Entities form the structural backbone of the Canonical Data Model and represent reusable concepts across programs and systems.
- **Attribute:** A property or characteristic of an entity. Attributes include data type, description, constraints, permissible values, and semantic meaning. Attributes **MUST** follow OHIP Data Architecture and Naming Standard.
- **Relationship:** The logical association between two or more entities. Relationships define cardinalities, ownership, and dependencies (e.g., Person–has–Eligibility, Provider–submits–Claim).
- **Standardized Naming Convention:** Names and definitions aligned to MITA²⁹, FHIM³⁰, NIEM³¹, and FHIR³² resources. Names **MUST** be consistent, unambiguous, and governed by the OHIP Data Architecture and Naming Standard.

²⁹ For more information, refer to [Medicaid Information Technology Architecture | Medicaid](#).

³⁰ For more information, refer to [FHIM Model | fhim.org](#).

³¹ For more information, refer to [Home | NIEM Open](#).

³² For more information, refer to [Welcome to the HL7 FHIR Foundation](#).

- **Data Type Definitions:** Standardized data types and controlled vocabularies (e.g., code sets, enumerations, value sets) aligned with federal standards and terminologies such as ICD-10³³, HCPCS³⁴, CPT³⁵, and FHIR ValueSets.
- **Semantic Definitions:** Precise definitions that describe the meaning and usage of each entity and attribute. Semantics ensure a common understanding across business units, the Contractor, and integration partners.
- **Cardinality Rules:** Specification of one-to-one, one-to-many, and many-to-many relationships. Cardinality helps define how data **MUST** be structured within interfaces and APIs.
- **Cross-Model Alignment References:** Mappings that show alignment between the Canonical Data Model and MITA Part II Data Standards, federal models such as FHIM, NIEM, FHIR, and/or industry models used by managed care organizations, providers, and federal partners
- **Reusable Component Definitions:** Reusable entities, attributes, code lists, and definitions that support multiple solutions. Examples include: Person, Organization, Address, Identifier, Event, and Status.
- **Governance Metadata:** Metadata describing lineage, ownership, steward, version, and authoritative source, maintained in OHIP's Enterprise Architecture Repository.

When developing a Canonical Data Model, the Contractor **MUST** follow the principles below to ensure consistency, interoperability, and alignment with enterprise architecture standards:

- Use Industry-Recognized Modeling Standards
 - o Model entities and relationships using:
 - UML Class Diagrams
 - ER Diagrams
 - ArchiMate 3.1 Information Structures
 - o Data modeling **MUST** follow best practices for normalization and semantic clarity.
- Align to Federal and MITA Data Models
 - o The Canonical Data Model **MUST** incorporate structures, definitions, and patterns from:
 - MITA Part II Data Standards
 - FHIM (Federal Health Information Model)
 - NIEM (National Information Exchange Model)
 - HL7 FHIR R4+ resources for clinical and administrative transactions
 - o Where discrepancies exist, the Canonical Data Model **MUST** document the rationale for deviations.

³³ For more information, refer to [ICD-10-CM | Classification of Diseases, Functioning, and Disability | CDC](#).

³⁴ For more information, refer to [Healthcare Common Procedure Coding System \(HCPCS\) | CMS](#).

³⁵ For more information, refer to [Need medical billing and codes resources? | American Medical Association](#).

- Establish Enterprise Naming and Definition Standards
 - o All conceptual, logical, and physical elements **MUST** comply with the OHIP Data Architecture and Naming Standard, including:
 - Naming conventions
 - Definition standards
 - Data type conventions
 - Controlled vocabularies
 - Prioritize Reuse of Common Entities
 - o Entities that represent common statewide concepts **MUST** appear once in the canonical model and be reused across all exchanges. Examples include:
 - Person
 - Provider
 - Encounter
 - Eligibility
 - Service Authorization
 - Claim / Claim Line
 - Model Data at the Logical Level for Interoperability
 - o The Canonical Data Model **MUST** represent business-oriented semantics, not system-specific database schemas. It **MUST NOT** contain Contractor-specific fields or proprietary attributes.
- Support API and Message Exchange Design
 - o API specifications, interface control documents, and data exchange standards **MUST** be derived from the canonical model. The model ensures consistency of:
 - JSON schemas
 - XML schemas
 - FHIR resource profiles
 - REST API payloads
- Document Data Lineage and Authoritative Sources
 - o Each entity and attribute **MUST** indicate its authoritative business source. This supports governance, data quality, and cross-system data reconciliation.
- Maintain the model in the OHIP's Enterprise Architecture Repository.

3.8.6 Metadata Model

The Metadata Model serves as a framework to manage metadata across various systems and processes efficiently.³⁶ This comprehensive structure includes definitions and guidelines for the following Metadata Model parts:

- Structural Metadata
- Syntactic Metadata
- Semantic Metadata
- Integration Metadata
- Business Metadata
- Operational Metadata

Specifications, such as indexes and partitioning, are only required for custom-developed databases, not for externally managed databases. When creating the Metadata Model, the Contractor **MUST** adhere to standards for metadata such as HL7/FHIR, DCAT, and ISO/IEC 11179.

Table 20 – Metadata Data Model

Solution Design	Data Architecture
Model Type and Notation	Diagrams (entity-relationship notation, such as Information Engineering or Oracle/Barker, which use the familiar “crow’s feet” relationship line ends OR UML class diagram).
Applicable Phase	Development, Implementation, and M&O Phase
Applicable Deliverable	Metadata Model and Dictionary

The metadata architecture serves as a guiding structure for comprehensive metadata management, leading to improved data governance, interoperability, and utilization across the organization. A sample metadata model is included in Figure 6.

³⁶ This artifact can also be referred to as Data Layout or as the TOGAF artifact Physical Data Model and Dictionary.

Semantic	Provide meanings and context for data, enabling better understanding and interoperability.	Ontologies, taxonomies, vocabularies, and data dictionaries.	<ul style="list-style-type: none"> • Establish a common vocabulary and terms glossary. • Implement tools for semantic enrichment, such as metadata tagging. • Ensure alignment with industry standard ontologies where applicable.
Integration	Facilitate data integration across systems by describing data linkages and movement.	Data mapping rules, transformation logic, data lineage.	<ul style="list-style-type: none"> • Define integration pathways and data flow mappings. • Capture and visualize data lineage across systems. • Utilize middleware and extract, transform, and load (ETL) tools to manage integration metadata.
Business	Associate data with business context and rules.	Business definitions, policy rules, data source information.	<ul style="list-style-type: none"> • Create business glossaries that map data assets to business terminologies. • Develop metadata-driven business rules management. • Implement dashboards for business metadata visualization.
Operational	Describe details about data processing and operational context.	Data processing logs, operational metrics, system performance data.	<ul style="list-style-type: none"> • Monitor and document data processing events and logs. • Analyze operational metrics for system performance and improvement. • Set up alerting for operational anomalies.
Sensitivity	Classify data based on its sensitivity level, identifying sensitive data, including	Sensitive data definitions	<ul style="list-style-type: none"> • Review data against definitions • Implement appropriate security and privacy

	personally identifiable information (PII)		measures, such as encryption, or strict access controls, to protect sensitive data from unauthorized access or misuse.
--	---	--	--

3.9 Maintaining the Data Architecture and Naming Standard

The Data Architecture and Naming Standard is a living artifact. The Contractor **MUST** support the annual review and maintenance of this standard by identifying proposed revisions based on evolving business needs, technology changes, updates to federal reference architectures (MITA, TOGAF, ArchiMate), the introduction of new modeling patterns or artifact types, and lessons learned during implementation. The Contractor **MUST** submit proposed revisions to OHIP for review and approval, including changes to naming conventions, abbreviation registries, modeling tool configurations, and architecture repository structures. OHIP will determine the classification of changes (e.g., minor versus major revisions) and their required approval path.

3.10 Related Data Maintenance Topics and Best Practices

3.10.1 Enterprise Glossary

Enterprise Glossary requirements, including ownership, governance, catalog integration, and definition standards, are consolidated in [Section 2.12.2](#).

3.10.2 Data Descriptions

Data definition standards governing how entities, attributes, and glossary terms are defined are established in [Section 2.12.2](#).

4.0 Database Design Standard

4.1 Introduction

This Database Design Standard establishes mandatory requirements for the physical implementation, operational maintenance, and change governance of databases within the OHIP MES. This standard applies to all database types across all OHIP systems, with specific requirements for Online Transaction Processing (OLTP) and Online Analytical Processing (OLAP) databases.

This standard governs what happens after the design and modeling phase: the physical realization of data models into functioning databases, the enforcement of data integrity at the database layer, and the ongoing maintenance and change control of database structures throughout their lifecycle. Data modeling and architectural design artifacts are governed by [Section 3.12](#).

This standard addresses:

- **Physical Implementation Standards:** Normalization requirements for OLTP databases, dimensional modeling requirements for OLAP databases, and referential integrity enforcement at the database layer.
- **Database Lifecycle Management:** Ongoing versioning, maintenance, and cleanup of database structures and their associated data models to prevent model decay and structural drift.
- **Change Control and Data Fix Governance:** Requirements for how database structure changes are proposed, approved, tested, and promoted across environments, and strict governance over direct data modifications.

4.2 Objectives

The objectives of this Standard are to:

- Enforce normalization and denormalization standards appropriate to each database type, ensuring OLTP databases maintain data integrity and OLAP databases support efficient analytical queries.
- Mandate referential integrity enforcement at the database layer through constraints, keys, and governed relationships, not solely through application code.
- Require lifecycle maintenance of all databases and their associated data models, preventing model decay and ensuring models accurately reflect the production database at all times.
- Govern database change control processes including Data Architect approval, automated DDL promotion, and synchronized release management.
- Establish strict governance over data fixes, ensuring direct data modifications are exception-based, tested, and tracked.
- Provide enforceable requirements that establish clear expectations for the Contractor implementing and maintaining databases on behalf of OHIP.
- Accommodate current OLTP and OLAP database types while establishing a framework that extends to emerging technologies such as graph databases, document stores, and vector databases as OHIP's ecosystem evolves.

4.2.1 Database Type Applicability

OHIP currently operates OLTP and OLAP databases as its primary database types. This standard provides specific requirements for both. Where requirements in this standard do not specify OLTP or OLAP, they apply to all database types. As OHIP's technology landscape evolves, the following database types **MAY** come into scope:

- Data Lakehouse databases (governed primarily by [Section 7.0](#), with this standard applying to physical implementation aspects)
- Graph databases
- Document stores (e.g., MongoDB, JSON-based systems)
- Vector databases (e.g., for machine learning and AI workloads)
- NoSQL databases

For database types not specifically addressed by the OLTP or OLAP requirements in this standard, the Contractor **MUST** follow the MITA Information Architecture, an industry or Contractor design standard approved by the Department, and a Data Management Plan per [Section 2.9](#). The Department reserves the right to require adherence to additional standards as new database types are introduced.

4.2.2 Multi-System Applicability

Multi-system applicability requirements are defined in [Section 1.4.1](#). The Contractor **MUST** demonstrate compliance with this standard within their specific system and database technology context.

4.3 Relationship to Other OHIP Standards

This section assumes that data models have been created and approved per [Section 3.0](#) before database construction begins. This section also operates in coordination with the Data Governance framework ([Section 2.0](#)), Data Object naming conventions ([Section 3.0](#)), and Data Lakehouse architecture ([Section 7.0](#)). For the complete list of external companion standards, see [Section 1.6](#).

This standard does not duplicate the data modeling, data dictionary, security, or quality gate requirements established in those companion standards. Cross-references are provided where requirements from other standards directly apply to database implementation activities.

4.4 Database Implementation Standards

The requirements in this section govern the physical implementation of databases after data models have been created and approved. All database designs, whether for OLTP or OLAP systems, **MUST** have a clearly defined purpose matched to specific business needs. The Department reserves the right to require alterations in database implementations that do not align with design best practices or when insufficient justification for the current implementation has been presented by the Contractor. The Department **MUST** have access to all data, tools, code, and solutions in all environments.

4.4.1 Normalization and Schema Standards

Database schemas **MUST** be designed according to their intended workload and usage patterns to maintain an appropriate balance between data integrity, consistency, and system performance. The normalization or denormalization approach **MUST** be documented in the Data Management Plan and

MUST follow an industry standard approved by the Department (e.g., DAMA DMBOK, Contractor-provided guidelines).

4.4.1.1 OLTP Normalization Requirements

OLTP database schemas **MUST** adhere to normalization principles and **MUST** conform to at least Third Normal Form (3NF) to minimize redundancy, prevent update anomalies, and ensure referential integrity. For a table to be considered compliant with 3NF, all of the following conditions **MUST** be met:

- The table **MUST** have a clearly defined primary key that uniquely identifies each row.
- All non-key attributes **MUST** be fully functionally dependent on the entire primary key.
- The table **MUST NOT** contain transitive dependencies, such that non-key attributes do not depend on other non-key attributes.
- All columns **MUST** contain atomic (indivisible) values, with each field representing a single data element.
- The table **MUST NOT** contain repeating groups or multi-valued attributes within any column.

While normalization is the default requirement for OLTP systems, selective denormalization **MAY** be permitted when justified by performance, scalability, or operational considerations. Any such denormalization **MUST** be explicitly documented, reviewed, and approved by the Department, and **MUST NOT** compromise data accuracy, auditability, or long-term maintainability.

4.4.1.2 OLAP Dimensional Modeling Requirements

OLAP database schemas **MUST** be implemented using dimensional modeling to support efficient querying, aggregation, and analytical reporting.

OLAP databases **MAY** employ controlled denormalization through dimensional modeling techniques rather than fully normalized relational structures. An OLAP dimensional schema **MUST** meet the following conditions, at minimum:

- Quantitative, measurable data elements are stored in a central fact table and represented as numeric, additive, or semi-additive measures.
- Descriptive and contextual business information is stored in separate dimension tables that provide meaning and analytical context for fact data.
- Dimension tables are denormalized to reduce join complexity and support intuitive query patterns.
- Each dimension table is related to the fact table through a single, well-defined primary–foreign key relationship.
- The overall schema supports predictable query performance, efficient aggregation, and drill-down or roll-up operations across defined dimensions.

Dimensional models **MUST** be designed to align with established business processes and reporting requirements and **MUST** be validated to ensure consistency, accuracy, and scalability across analytical use cases.

4.4.2 Referential Integrity and Constraint Standards

The Contractor **MUST** utilize database constraints in addition to primary and unique keys to ensure data uniqueness and integrity. Handling data quality controls exclusively in the application layer is not

acceptable. Referential integrity **MUST** be managed via inherent relational database capabilities and **MUST** be clearly documented in the data model via relationships and constraints in the database.

The following minimum requirements apply to all databases:

Table 22 – Referential Integrity and Constraint Standards

Requirement	Description
Primary Key (PK) Requirement	Every table MUST have a unique identifier (primary key) with a governed definition. Any table without a primary key MUST provide an explanation in the data dictionary and receive specific approval from the Department.
Natural Key Requirement	Every table MUST have a natural key: the minimum set of business attributes that uniquely identifies a row without relying on system-generated values. The natural key MUST be documented in the data dictionary and enforced through a unique constraint at the database layer, even when a surrogate key serves as the primary key.
Surrogate Key Requirement	Tables MAY use a surrogate key (a system-generated identifier with no business meaning) as the primary key. When a surrogate key is used, the natural key MUST still be identified, documented, and enforced with a unique constraint. Surrogate key naming MUST follow the conventions defined in Section 3.6.5 (Surrogate Key Naming)
Foreign Key Requirement	The Contractor MAY define foreign key constraints for all parent-child relationships between tables. Foreign keys MUST enforce referential integrity at the database layer, not solely through application logic
Primary Key Naming	All primary keys MUST follow the naming convention defined in Section 3.0 , including table name reference, as opposed to simply “ID.”
Foreign Key (FK) Naming	Foreign keys MAY use the same key name as the primary key in the referenced table to ensure clarity and traceability across the data model.
PK–FK Relations	Designers MUST establish a primary key–foreign key relation between main and secondary tables to prevent orphan records. If such a relationship does not exist, reasons MUST be provided in the data dictionary and be approved by the Department.
Database Constraint Usage	Designers MUST use default value constraints, check constraints, and unique, primary, and foreign key constraints at the database layer to ensure data accuracy and limit incorrect data entry. Developers and Data Architects MUST handle errors generated at the database layer in their code for data inserts, updates, or deletes that break constraint requirements.

4.4.3 Database Elements and Partitioning

For the database platform chosen, all elements (e.g., tables, views, indexes, stored procedures, functions, triggers, constraints, sequences, identity columns, schemas, synonyms, materialized views, and partitions) **MUST** have proper usage, optimization, and maintenance standards documented in the Data Management Plan and approved by the Department.

Database partitioning strategies, when applicable, **MUST** be documented in the Data Management Plan with justification for the chosen approach. The partitioning strategy **MUST** account for transactional consistency, cross-partition querying, query performance, and operational complexity. Partitioning approaches include but are not limited to horizontal (sharding), vertical, range, list, hash, and hybrid partitioning. The selected approach **MUST** be approved by the Department before implementation.

Database performance monitoring and capacity planning requirements are governed by the OHIP Infrastructure and Operations Domain Document, ITSM Standard (Capacity and Performance Management). The Contractor **MUST** establish database performance baselines, monitor query execution times, and track resource utilization trends. Performance degradation exceeding agreed thresholds **MUST** be reported and addressed per the OHIP incident management process.³⁷

4.4.4 Data Storage Standards

Data in the database **MUST** be stored in a user-friendly, readable format where no application layer translation is required to understand or properly view the data. A query against a field in the database **MUST** be able to retrieve the correct data without further translation. For instance, special characters **MUST** be stored the way they are expected to be presented to the user, not in ASCII format in the database.

All database object names, entity names, attribute names, and index names **MUST** comply with [Section 3.0](#). Index names **MUST** use a clear prefix such as IDX_ and include table and column references (e.g., IDX_Customers_LastName). Composite index names **MUST** reflect their components (e.g., IDX_Orders_Date_Status).

4.4.5 Audit Column Standards

All database tables **MUST** include standard audit columns to support traceability, accountability, and data change tracking. These columns are a database design requirement: they **MUST** be built into the table structure, not added after the fact.

The following audit columns **MUST** appear in every table unless a specific exception is documented in the data dictionary and approved by the Department:

Table 23 – Audit Column Standards

Audit Column	Data Type	Description
created_date	Datetime	Timestamp of record creation. MUST be system-generated and not modifiable by application users.
created_by	Varchar	Identifier of the user or process that created the record. MUST reference the authenticated user or batch process ID.
last_updated_date	Datetime	Timestamp of the most recent modification to the record. MUST be system-generated and updated automatically on any change.
last_updated_by	Varchar	Identifier of the user or process that last modified the record.
logical_delete_flag	Char(1) or Boolean	Indicates whether the record has been logically deleted. Physical deletion of records is prohibited unless explicitly approved by the Department and documented per Section 4.5.4 . Default value MUST be 'N' or FALSE.

Audit column naming **MUST** comply with [Section 3.0](#). Where the Contractor proposes alternative audit column implementations (e.g., temporal tables, change data capture, or history tables), the approach

³⁷ For more information, refer to the OHIP Infrastructure and Operations Domain Document, ITSM Standard.

MUST be documented in the Data Management Plan, **MUST** provide equivalent traceability, and **MUST** be approved by the Department prior to implementation.

Audit columns **MUST NOT** be included in application-facing data entry screens or made editable by end users. Audit column values **MUST** be populated and maintained exclusively by the database layer or application middleware.

4.4.6 Index Management Standards

Indexes are critical to database performance and **MUST** be governed with the same rigor as other database objects. The following requirements apply to all index creation, maintenance, and removal activities:

- **Index Creation and Justification:** Every index **MUST** have a documented business or technical justification in the data dictionary. Indexes **MUST NOT** be created speculatively. Each index **MUST** be associated with a specific query pattern, report requirement, or application function. The justification **MUST** document the query or workload the index supports and the expected performance impact.
- **Index Naming:** All indexes **MUST** comply with [Section 3.0](#). Index names **MUST** use the IDX_ prefix and reference the table and column(s) they support (e.g., IDX_Claims_ServiceDate, IDX_Members_LastName_DOB).
- **Index Monitoring and Optimization:** The Contractor **MUST** monitor index usage on a regular cadence and report on unused or underutilized indexes. Unused indexes (indexes that have not been referenced by any query within two consecutive reporting periods) **MUST** be flagged for review and removal. Index fragmentation **MUST** be monitored and addressed through rebuild or reorganization schedules documented in the Data Management Plan.
- **Index Removal:** Removing an index follows the same change control process as any DDL change ([Section 4.5.3](#)). Before removal, the Contractor **MUST** verify that no active query, report, or application function depends on the index.
- **Composite Index Governance:** Composite indexes (indexes spanning multiple columns) **MUST** document the column order rationale and the query patterns that justify the specific column sequence. Composite index names **MUST** reflect their components (e.g., IDX_Orders_Date_Status).

4.5 Database Lifecycle Management

Database structures and their associated data models **MUST** be actively maintained throughout the life of each system. Model decay, where the data model no longer reflects the actual database, is not acceptable. The requirements in this section establish enforceable standards for versioning, maintenance, change control, and data fix governance.

Database stabilization activities performed during post-Go-Live stabilization occur within the broader Hypercare period governed by the Solution Delivery Domain Document, which is the authoritative standard for the Hypercare lifecycle including the Hypercare Plan, SLA obligations, incident management, KPIs, exit criteria, and Hypercare Completion Report. The database-specific requirements in this section supplement and operate within that framework: they do not replace it. DBAs and data engineering staff engaged in post-Go-Live stabilization **MUST** comply with both this section and the Solution Delivery Domain Document concurrently.

4.5.1 Model and Database Versioning

Database designers **MUST** provide data model versioning. Prior to any release, the designer **MUST** create versions (delta and named) to keep track of all changes. Versioning **MUST** be included in the Data Management Plan.

Database versions **MUST** be updated on a regular cadence based on how often the specific database is updated. The Contractor **MUST** provide a regular schedule of updates in an organized fashion, ensuring that there are no negative impacts by testing every environment up the stack. Tests **MUST** include any application that uses the databases being updated.

Updated data dictionaries and data models (in the Department's preferred modeling tool) **MUST** be provided for review and validation with each major application release. The data model **MUST** accurately reflect the production database at all times.

 **Note:** [Section 3.0](#) mandates that all models remain up to date. This standard provides the operational requirements for how that mandate is enforced at the database level.

4.5.2 Database Maintenance and Cleanup

The minimum requirements in Table 25 below ensure databases remain clean, current, and free of unnecessary objects:

Table 24 – Database Maintenance and Cleanup

Requirement Area	Description
Maintenance of Database Design	Database Designers MUST identify tables or objects no longer in active use. These MUST be marked as obsolete in the data dictionary and be removed in a subsequent release.
Removal of Legacy Tables and Objects	Legacy tables or objects cluttering the database or causing unnecessary storage use MUST be removed. If they contain data for long-term retention, data MUST be archived before dropping the table or object.
Timely Removal of Obsolete Objects	Obsolete tables and objects MUST be removed within two (2) major revisions. An obsolete object is an object no longer updated or selected by application code.
Removal of Temporary Tables	Temporary tables MUST be removed regularly as part of the process that created them. Any table in the database without records is considered temporary or obsolete and MUST be removed.
Data Dictionary Currency	The data dictionary MUST reflect all data elements, including those no longer in active use (marked as obsolete). The data dictionary MUST be updated concurrently with any database structural change.

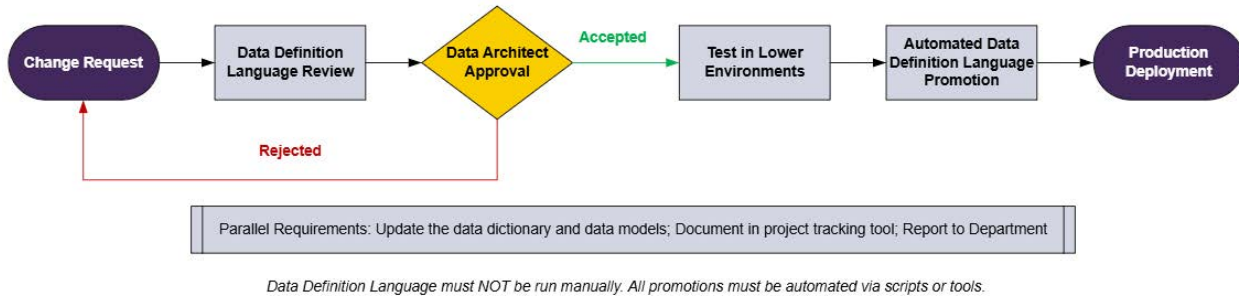


Figure 7 - Database Change Control Workflow

Figure 7 provides the Contractor the process before the rules.³⁸ Database implementations **MUST** support the retention periods, archival procedures, and purge processes defined in [Section 2.8.1](#). The Contractor **MUST** ensure that database designs accommodate retention-based partitioning, archival extraction, and controlled purge execution without compromising referential integrity or audit traceability.

4.5.3 Database Change Control

Database schema changes, model changes, and configuration changes to production database environments are subject to a two-layer change control obligation. This section governs the database-layer process, including change classification, impact assessment, version control of database objects, and rollback planning.

Any database change that affects a production or production-adjacent environment **MUST** also be packaged, scheduled, and approved through the OHIP Solution Delivery Domain Document, Release Management Standard, which governs release scheduling, scope freeze, change request submission, Go/No-Go approval, and deployment coordination across all integrated modules. A database schema change **MUST NOT** be promoted to production outside of an approved release package except where explicitly authorized as a HotFix or Expedited Release under the Release Management Standard. Where the two processes impose conflicting timelines or requirements, the Contractor **MUST** escalate to OHIP in writing for a written determination before proceeding.

All database models and structural changes **MUST** undergo review and approval by OHIP. Database change control operates within the general change management framework defined in the OHIP Infrastructure and Operations Domain Document, ITSM Standard, with the following database-specific requirements:

Table 25 – Database Change Control

Requirement Area	Description
Synchronizing Database Changes with Application Releases	Database structure or design changes MUST be updated in line with application changes and releases. Changes MUST be given the same release numbers for consistency. An updated data dictionary and data model MUST be provided for review and validation with each major application release.
Data Architect Approval for DDL Changes	All DDL changes MUST be approved by a Data Architect. Data Architects MUST ensure that proposed design changes follow approved naming

³⁸ See [Section 2.0](#) for database retention and purge requirements and [Section 7.0](#) for information on Data Lakehouse environments.

	conventions, include necessary audit columns, use correct data types, and do not create duplicate or conflicting data in the overall data model.
Documentation of Database Changes	Database structural changes MUST be documented in separate tickets from application changes in the Department's approved project tracking tool, tagged as database change requests for easy sorting, filtering, and status tracking.
Tracking and Reporting	The Contractor MUST provide reports on the volume of database change requests to the Department on a regular cadence.
Automated DDL Execution	DDL scripts MUST NOT be run manually. DDL MUST be implemented via automated processes and tools or scripts for promotion from lower- to higher-level environments and into production. Errors from executing DDL MUST be reported and addressed prior to production. Any parameters for DDL MUST be automated, not manually provided.

4.5.3.1 Backward Compatibility and Schema Migration

Database schema changes **MUST** be designed to minimize disruption to downstream consumers, including applications, reports, ETL processes, and analytical queries. The following requirements apply, at minimum:

- **Breaking Changes (Requires Impact Analysis):** Breaking changes (such as renaming or removing columns, changing data types, modifying primary or foreign key structures, or altering stored procedure signatures) require a documented impact analysis before approval. The impact analysis **MUST** identify all affected consumers (applications, reports, ETL jobs, APIs) and **MUST** include a communication plan for coordinating the change with affected teams. All consumers **MUST** be updated or remediated concurrently with the schema change.
- **Deprecation Protocol:** When a database object (table, column, view, stored procedure) **MUST** be removed, the Contractor **MUST** follow a deprecation protocol: mark the object as deprecated in the data dictionary, notify all known consumers, maintain the deprecated object for a minimum of one (1) major release cycle, and remove the object only after confirming that no active consumer depends on it.
- **Migration Scripts:** All schema migrations **MUST** be implemented through versioned, idempotent migration scripts that can be executed repeatedly without error. Migration scripts **MUST** include both the forward change and a documented rollback procedure. Rollback procedures **MUST** be tested in a non-production environment before the migration is approved for production deployment.
- **Non-Breaking Changes (Preferred):** Where possible, schema changes **SHOULD** be non-breaking; that is, existing consumers **SHOULD** continue to function without modification after the change is deployed. Non-breaking changes include adding new columns with default values, adding new tables, adding new indexes, and extending column lengths. Non-breaking changes still require Data Architect approval and change control documentation.

All database changes **MUST** be tested in accordance with the OHIP Solution Delivery Domain Document, Test Management Standard. Testing requirements applicable to database changes include DDL validation testing, regression testing of affected application functions and reports, constraint and referential integrity verification, and performance baseline comparison to ensure schema changes have not degraded query response times. Test results **MUST** be documented and provided to the Department as part of the change request documentation.

For Lakehouse environments, additional schema evolution requirements, including schema registries, table format evolution (Apache Iceberg, Delta Lake), versioned schema capabilities, and consumer notification timelines, are defined in [Section 7.11](#).

4.5.4 Data Fix Governance

Data fixes applied to production systems are subject to the same two-layer change control obligation described in [Section 4.5.3](#). This section governs the data-layer process for identifying, authorizing, executing, and documenting data corrections.

Any data fix that modifies production data or triggers a schema change **MUST** also be submitted as a change request under the OHIP Solution Delivery Domain Document, Release Management Standard and **MUST** comply with the release scheduling, scope freeze, and deployment coordination requirements of that standard. Data fixes **MUST NOT** bypass the Release Management change control process on the basis that they do not involve code changes.

Data fixes via external scripts are strictly exception-based and **MUST NOT** be used as routine operational practice. The Contractor **MUST** prioritize resolving the underlying application logic defect over performing recurring manual data fixes. All data fix events **MUST** be logged and tracked.

Data **MUST** only be updated through the application GUI or automated application processes, such as batch jobs and interfaces. Data fixes outside of the application **MAY** have unanticipated effects. Any situation that requires a data fix **MUST** generate a problem ticket and lead to changes in application logic to avoid repetitive data fixes. Figure 8 below demonstrates the decision tree for data fixes.

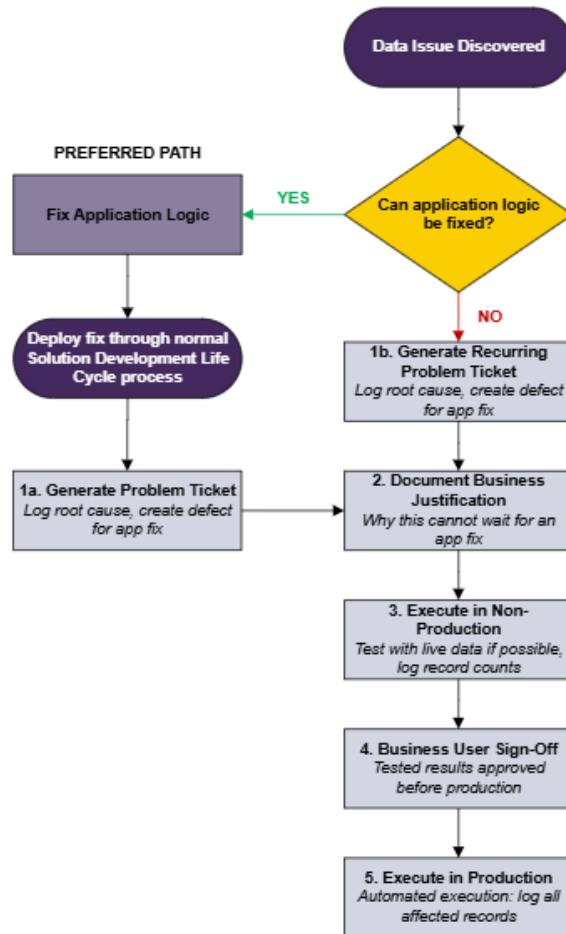


Figure 8 - Data Fix Decision Tree

The following requirements apply to all necessary data fixes:

- The data fix **MUST** track the number of records it adjusts via a logging mechanism.
- The fix **MUST** be thoroughly validated by business users and approved before running in production to mitigate unexpected impacts.
- All data fixes **MUST** be documented, including the business justification, the records affected, the testing results, and the business user sign-off.
- Recurring data fixes addressing the same root cause **MUST** trigger an application defect ticket with a required remediation timeline.

Note: The fix **SHOULD** be executed in a non-production environment with live data, if possible, to ensure the expected impact on production is known.

4.5.5 Database Environment Management

Database environments **MUST** ensure schema consistency, data protection, and reliable promotion of changes from development through production. The following requirements supplement the general

environment management requirements defined in the OHIP Solution Delivery Domain Document, SDLC Standard:

- **Environment Parity:** Database schemas **MUST** be consistent across all environments (development, test, staging, production). Schema drift, where lower environments diverge from the production schema, is not acceptable. The Contractor **MUST** implement automated schema comparison tools or processes to detect and remediate drift on a regular cadence.
- **Protected Data in Non-Production Environments:** Production data containing PHI, PII, FTI, or MCD **MUST NOT** be copied to non-production environments without masking or de-identification in accordance with the OHIP Security and Privacy Compliance Standard. Where the Data Fix Governance process ([Section 4.5.4](#)) requires testing with live data, the Contractor **MUST** use a masked or de-identified copy of production data unless a specific exception is approved by the Department and the OHIP Security and Privacy Officer. The exception **MUST** document the business justification, the duration of access, and the controls in place to protect the data.
- **Environment Refresh:** The Contractor **MUST** document and maintain a schedule for refreshing lower environments from production (with appropriate masking). Refresh schedules **MUST** be included in the Data Management Plan and approved by the Department. After each refresh, the Contractor **MUST** verify that all application-specific configurations, test accounts, and non-production data fixtures are restored.
- **Environment Access:** Database access permissions **MUST** differ by environment. Developers **MAY** have elevated object-level permissions (e.g., CREATE, ALTER) in development environments only. In test, staging, and production environments, all DDL changes **MUST** follow the automated promotion process defined in [Section 4.5.3](#). Direct DDL execution by developers in production is prohibited.

4.6 Database Security

All database activities **MUST** comply with the OHIP Security, Privacy, and Compliance Domain Document, Security and Privacy Compliance Standard. Database-specific security requirements including identity and access management integration, role-based access control, encryption at rest and in transit, DBA and service account governance, auditing, and connection logging are defined in that standard.

The Contractor **MUST** ensure that database security implementations are coordinated with the Department's enterprise Identity Access Management (IAM) solution. After initial database deployments and integration with the Department's IAM solution, local database accounts **MUST NOT** be used. For the complete set of database security requirements, refer to the OHIP Security, Privacy, and Compliance Domain Document, Security and Privacy Compliance Standard.

5.0 Data Conversion Standard

5.1 Introduction

This Data Conversion Standard establishes mandatory requirements for converting and migrating data between information technology systems within or on behalf of OHIP, including systems operated by Contractors and partner agencies in support of OHIP MES operations. It governs the end-to-end conversion lifecycle: from source data profiling and cleansing through transformation, validation, cutover, and post-go-live stabilization. Deliverables and artifacts associated with this Standard can be found in [Appendix A](#).

This standard governs conversion-specific activities. Requirements for database design,³⁹ data pipeline quality,⁴⁰ security controls,⁴¹ and testing methodology⁴² are defined in their respective companion standards. [Section 1.6](#) provides the complete list of companion standards and cross-references.

 **Note:** This standard assumes source and target environments have been designed and validated per the [Section 4.0](#) and [Section 7.0](#). It governs the process of moving data between those environments.

This standard applies to:

- One-time legacy data migrations (e.g., system decommissioning and replacement)
- Ongoing data exchanges and interfaces between OHIP systems
- Ongoing data synchronization between legacy and target systems during phased conversion deployments, where data bridging is required to maintain operational continuity until full cutover is achieved
- Data conversions required for system modernization, including SaaS/COTS transitions

 **Note:** Standing operational interfaces and recurring data exchanges that are not part of a conversion or migration effort are governed by the applicable integration and data pipeline standards, not this standard.

As OHIP's technology landscape evolves, this standard may be extended to address cloud-native migration patterns (e.g., containerized ETL/ELT), API-first conversions, and streaming data pipelines. These areas will be addressed in future revisions as adoption matures.

³⁹ For more information, refer to [Section 4.0](#).

⁴⁰ For more information, refer to [Section 7.0](#).

⁴¹ For more information, refer to the OHIP Security, Privacy, and Compliance Domain Document, Security and Privacy Compliance Standard.

⁴² For more information, refer to the OHIP Solution Delivery Domain Document, Test Management Standard.

5.2 Objectives

The objectives of this standard are to:

- Ensure accuracy, completeness, and consistency of converted data by applying rigorous profiling, cleansing, transformation, and validation protocols.
- Maintain alignment with federal requirements, including the CMS MITA framework, CMS CEF, HIPAA transaction standards, and Acceptable Risk Controls for ACA, Medicaid, and Partner Entities (ARC-AMPE).
- Enable interoperability for data exchange using standardized formats including HIPAA EDI X12, HL7v2, FHIR, and USCDI v6 as appropriate to the transaction type.
- Maintain full auditability through the conversion lifecycle by maintaining comprehensive logs for data lineage, error tracking, reconciliation, and compliance reporting.
- Minimize risks associated with data conversion through robust backup, recovery, and audit mechanisms with full OHIP visibility.
- Support modernization efforts for the transition from legacy systems to modern SaaS and COTS solutions by defining open integration approaches and conversion best practices.

5.3 Companion Standards

This standard operates within the OHIP MES Standards Framework. Data conversion activities **MUST** comply with the target database design ([Section 4.0](#)), data governance and retention policies ([Section 2.0](#)), and Lakehouse quality gates for post-conversion data ([Section 7.0](#)). Source and target data models **MUST** conform to [Section 3.0](#). All conversion activities involving sensitive data **MUST** comply with the OHIP Security, Privacy, and Compliance Domain Document, Security and Privacy Compliance Standard. For the complete list of companion standards, see [Section 1.6](#).

5.4 Guiding Principles

All Contractors **MUST** adhere to the following guiding principles during data conversion. These principles establish the baseline expectations for all conversion activities.

Table 26 – Data Conversion Guiding Principles

Principle	Requirement
Real-Time Transfers	Where source and target solutions can support real-time or near real-time API (API) or event-based data transfers, real-time transfer MAY be used. Scheduled exchanges are permissible only for legacy system decommissioning or when the target system cannot support real-time transfers.
Federal Compliance	All data exchanges and conversions MUST comply with relevant federal standards, including: <ul style="list-style-type: none">• CMS MITA Framework• CMS CEF• Acceptable Risk Controls for ACA, Medicaid, and Partner Entities (ARC-AMPE)• HIPAA transaction standards (EDI X12)

	<ul style="list-style-type: none"> Health Level Seven (HL7) v2 and Fast Healthcare Interoperability Resources (FHIR) as applicable to transaction type United States Core Data for Interoperability (USCDI) v6 where applicable <p>Conversion MUST maintain or improve data governance and compliance with all applicable federal requirements.</p>
Data Quality	Conversion activities MUST maintain or improve data quality. Source data MUST be profiled and cleansed during conversion. Converted data MUST retain its accuracy in the target system. See Section 5.5 and 5.6 for specific requirements.
Data Accountability	Data Owners (or an authorized authority) MUST approve all conversion decisions including source-to-target mappings, cleansing rules, and final results. The Data Owner is accountable for resolving conversion questions and accepting final results.
Appropriate Transaction Types	Conversion MUST use the appropriate data exchange standard for each transaction type: <ul style="list-style-type: none"> Clinical Transactions: FHIR preferred for patient-centric clinical data (EHR sharing, lab results, patient updates) Administrative Transactions: X12 standards for billing, claims processing, eligibility inquiries, and remittance advice Legacy Integration: HL7v2 for systems with established HL7v2 implementations
Data Governance Compliance	Converted data MUST adhere to Section 2.0 and all OHIP data governance and management practices. This Standard does not define the complete set of data governance requirements; Section 2.0 is the authoritative source for enterprise governance, data quality dimensions, Master Data Management (MDM) keys, and ethical data use requirements.
Auditability and Lineage	All conversion activities MUST maintain full traceability from source through transformations to target. 100% of target records MUST carry audit metadata (source system, extract timestamp, load timestamp, job ID, transformation version). Column-level lineage MUST be able to be queried for all converted attributes.
Reproducibility	The Contractor MUST be able to reproduce any conversion run with its exact rule set, code baseline, and configuration. All conversion scripts MUST be version-controlled.
Transparency	At OHIP's request, all conversion activities, data, environments, artifacts, and related materials MUST be provided and demonstrated. This standard does not require OHIP to perform reviews.

5.5 Pre-Conversion Standards

Before conversion execution begins, the Contractor **MUST** complete the following pre-conversion activities. These activities form the foundation for a successful conversion and **MUST** be documented in the Data Conversion and Migration Plan.

5.5.1 Data Profiling

The Contractor **MUST** profile 100% of in-scope source data at the attribute level before conversion design begins. Data profiling establishes the baseline understanding of source data quality and drives all subsequent cleansing, mapping, and transformation decisions.

Profiling Requirements

The data profiling assessment **MUST** include, at minimum:

- **Statistical Profiling:** Distributions, outliers, null ratios, distinct value counts, and value frequency analysis for every in-scope attribute.
- **Completeness Analysis:** Percentage of populated values for every attribute. Attributes below 95% completeness **MUST** be flagged for remediation or documented exception.
- **Uniqueness Analysis:** Identification of duplicate records by business key. Duplicate rate **MUST** be quantified per entity.
- **Referential Integrity Analysis:** Identification of orphan records, broken references, and cross-table relationship violations.
- **Pattern Analysis:** Format patterns, value ranges, and domain validation for attributes with expected formats (dates, codes, identifiers).
- **Temporal Profiling:** Overlaps, gaps, and anomalies in effective-dated records and time-series data.
- **Current vs. Historical Data Classification:** The profiling assessment **MUST** identify and document whether each data domain contains current-state records only, historical records, or a combination of both. The temporal scope of each domain **MUST** be defined, including effective date ranges, applicable archival policies, and any records that are superseded, inactive, or retired. This classification directly informs conversion scope decisions, effective dating strategy, and post-conversion data retention requirements.
- **Constraint Discovery:** Identification of functional dependencies, implied business rules, undocumented data relationships, and natural key determination. Natural keys **MUST** be identified for each in-scope entity to inform deduplication logic, survivorship rules, and source-to-target mapping decisions.

Profiling Deliverable

The Contractor **MUST** produce a Data Quality Assessment Report with severity-ranked issues, heatmaps by domain, and remediation assignments. Baseline data quality scorecards **MUST** be produced for each data domain. In the event a domain is below agreed quality thresholds, the Contractor **MUST** develop a CAP⁴³ to meet agreed quality thresholds during profiling. The plan **MUST** identify quality issues by severity and a remediation approach (source fix, ETL transformation, or documented exception). Conversion design or development **MUST NOT** begin for a domain without an approved CAP.

 **Note:** The Data Quality Assessment Report is a required deliverable (see [Appendix A](#)). Conversion design or development **MUST NOT** begin without an approved profiling report.

⁴³ For more information on the Corrective Action Plan, refer to the OHIP Security, Privacy, and Compliance Domain Document, Security and Privacy Compliance Standard.

5.5.2 Data Cleansing Strategy

Based on profiling results, the Contractor **MUST** develop and execute a Data Cleansing Strategy that defines how source data quality issues will be remediated before or during conversion.

Cleansing Requirements

- **Cleansing Categorization:** Each identified quality issue **MUST** be categorized as: (a) **MUST**-fix at source before extraction, (b) **MUST**-fix via transformation rules during ETL, or (c) acceptable with documented exception. The rationale for each categorization **MUST** be logged.
- **Deduplication and Householding:** The Contractor **MUST** define match rules (exact and fuzzy) with confidence thresholds for deduplication. Survivorship rules **MUST** specify source precedence, recency rules, and rule-based aggregation logic. All match decisions and rule versions **MUST** be retained for audit traceability.
- **Cleansing Metrics:** The Contractor **MUST** publish quality scorecards per domain showing pre-cleansing baseline, post-cleansing improvement, and remaining known issues. Trend improvements **MUST** be tracked across conversion iterations.
- **Manual Remediation Workflow:** For issues that cannot be resolved programmatically, a ticketing-based workflow **SHOULD** be established with assigned owners, due dates, and verification steps. Data Stewards **SHOULD** approve manual remediation actions.

5.5.3 Reference and Master Data Mapping

Before transformation begins, the Contractor **MUST** identify and reconcile all reference data domains between source and target systems.

- **Code Mapping Strategy:** Every reference data domain (e.g., status codes, type codes, geographic codes, provider types) **MUST** have a documented crosswalk mapping legacy values to target values. Crosswalks **MUST** be version-controlled and approved by the Data Steward.
- **Effective Dating:** Reference data changes **MUST** be governed with versioning and effective dating. The conversion **MUST** correctly handle point-in-time lookups for historical data.
- **Master Data Management:** Where applicable, the Contractor **SHOULD** define the approach for creating golden records, including survivorship rules and source precedence logic, for all applicable domains including provider master data, member/patient master data, and organizational hierarchies.

5.5.4 Source-to-Target Mapping

The Contractor **MUST** produce comprehensive source-to-target mapping documentation before conversion development begins.

Mapping Requirements

- **100% Coverage:** 100% of target attributes **MUST** be mapped with business rules, transformation logic, defaulting rules, survivorship rules, and reference data crosswalks documented at the column level.
- **Full Lineage:** 100% of mappings **MUST** include source-to-target lineage, including source table/column, transformation version, and target table/column.
- **State Sign-Off:** OHIP sign-off is required on mapping specifications before development begins. Any change after sign-off requires a formal change control request and an impact assessment.

- **Transformation Rule Specification:** Each rule **MUST** include explicit handling for: nulls and defaults, truncation behavior, time zone conversions, rounding and precision, currency formatting, and character encoding normalization.
- **Version Control:** Each rule **MUST** be tagged with a version number and effective date. A change history **MUST** be maintained.
- **Conformed Dimensions (Target-Side Requirement):** For data destined for analytical systems (OLAP), source-to-target mappings **MUST** account for the requirement that reference data conform to shared dimensions across fact tables in the target environment, per [Section 4.0](#). Transformation logic **MUST** produce conformed dimension values that are consistent and reusable across all target fact tables that reference the same domain.

 **Note:** The Data Mapping and Transformation Specification Document and the Data Conversion and Migration Plan are required deliverables (see [Appendix A](#)). The Contractor **MUST** obtain OHIP approval for both before development begins.

5.5.5 Data Conversion and Migration Plan

Before development of any data transfer, the Contractor **MUST** develop a Data Conversion and Migration Plan and obtain OHIP approval before work commences. The complete list of required plan components is defined in [Appendix A](#). The plan **MUST** be updated throughout the conversion lifecycle to reflect current scope, schedule, and risk status.

5.5.6 SaaS/COTS for Data Exchange

Conversions **SHOULD** leverage SaaS or COTS tools that implement federal standards such as HL7/FHIR where available and appropriate. Contractors **SHOULD** first examine every avenue of using supported products for compliant data exchange before proposing custom-built conversion solutions.

5.6 Conversion Execution

This section defines requirements for the execution of data conversion activities. The data conversion process consists of three major phases: Pre Conversion, Conversion, and Post-Conversion.

5.6.1 Conversion Phases

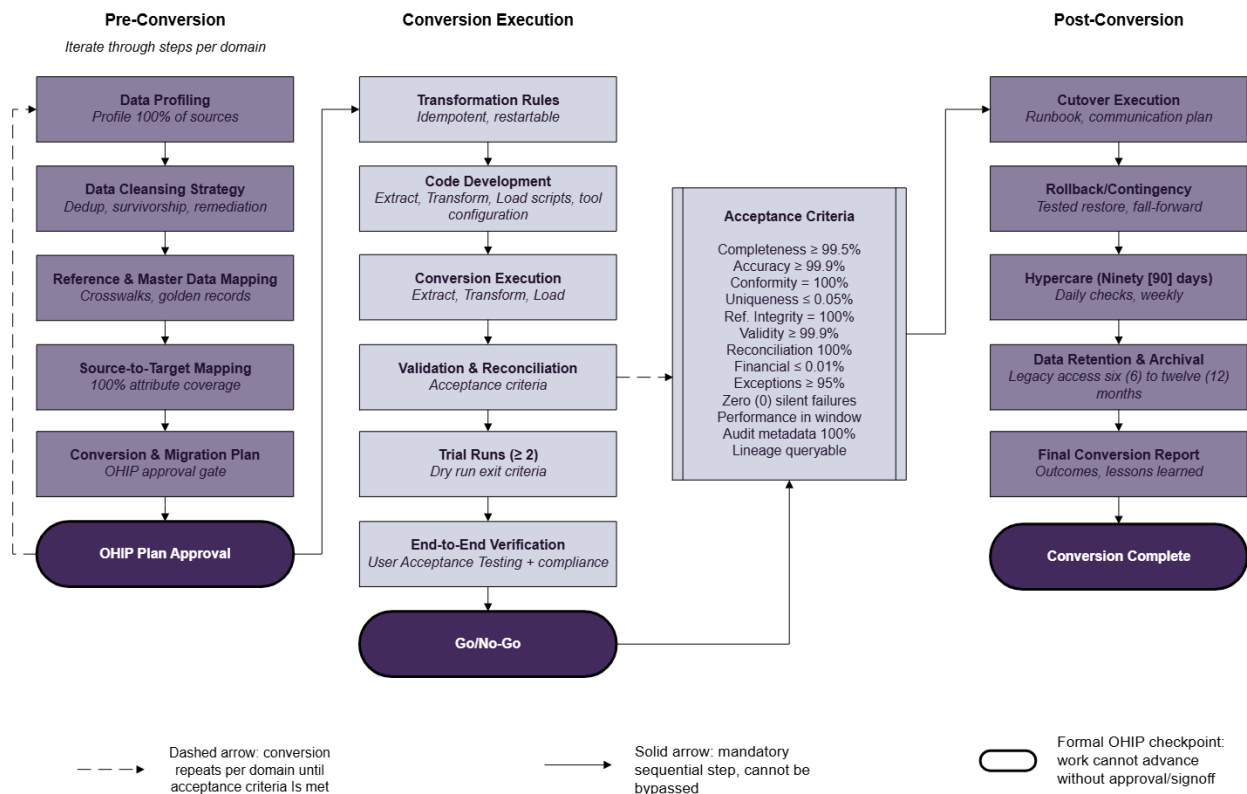


Figure 9 - Data Conversion Phases and Activities

The Artifacts column in Tables 28-30 identifies work products associated with each conversion phase activity. These artifacts are further defined in [Appendix A](#), which distinguishes between Deliverables (contractually binding work products subject to formal OHIP review and approval) and Artifacts (work products the Contractor **MUST** create, maintain, and retain, and provide to OHIP upon request). The Contractor **SHOULD** review [Appendix A](#) in conjunction with these tables to understand the full scope of required outputs.

The Contractor **MUST** conduct the following Pre-Conversion activities in alignment with OHIP expectations and the approved Data Conversion and Migration Plan. All activities **MUST** be completed and documented before Conversion execution begins.

Table 27 – Pre-Conversion Phase Activities

Activity	Description	Artifacts
Analyze and establish requirements	Profile and review source data per Section 5.5.1 . Identify quality issues, mapping requirements, and cleansing needs.	Data Quality Assessment Report, Data Profiling Reports
Define architecture	Determine the structure for each environment needed for conversion (dev, test, staging, production).	Environment Architecture Documentation
Establish environments	Create and configure environments as detailed in the Data Conversion and Migration Plan.	Environment Readiness Confirmation

Define iterations	Define and logically group data conversion iterations per the Data Conversion and Migration Plan. Prioritize high-impact domains.	(Documented within the Data Conversion and Migration Plan, Iteration Approach component)
-------------------	---	--

The Contractor **MUST** conduct the following Conversion activities in alignment with OHIP expectations. Any deviation from the approved plan **MUST** be documented, justified, and approved by OHIP before proceeding.

Table 28 – Conversion Phase Activities

Activity	Description	Artifacts
Transformation rules and mapping	The Contractor MUST create data transformation rules to cleanse, transform, and enhance data. Create source-to-target mappings. Incorporate format conversions, value mapping, and augmentations per Section 5.5.4 .	Data Mapping and Transformation Specification Document
Code development and tool configuration	The Contractor MUST develop scripts, code, and configure required tools for extraction, transformation, and loading.	Scripts, code, and configurations
Conversion execution	The Contractor MUST execute conversion with pre-conversion backup, iterative loading, validation at each checkpoint, and performance monitoring.	Logs, load reports, performance test results
Validation and reconciliation	The Contractor MUST validate converted data against acceptance criteria (Section 5.6.3) and perform reconciliation and balancing (Section 5.6.4).	Validation reports, reconciliation evidence

The Contractor **MUST** conduct the following Post-Conversion activities in alignment with OHIP expectations. Hypercare monitoring **MUST** remain active for the period defined in [Section 5.8.3](#).

Table 29 – Post-Conversion Phase Activities

Activity	Description	Artifacts
Data conversion cutover	The Contractor MUST verify production conversion success. Cutover to target system. Begin source system decommissioning.	Cutover Report, Unconverted Data Register
Post-go-live stabilization	The Contractor MUST execute Hypercare monitoring per Section 5.8.3 . Track and resolve post-conversion defects.	Hypercare reports, defect logs
Final conversion report	The Contractor MUST produce a comprehensive report summarizing outcomes, lessons learned, data quality achieved, and final approvals.	Final Conversion Report

Rollback / Contingency	The Contractor MUST execute the rollback plan if go/no-go criteria are not met or if critical defects arise during cutover. This includes restoring pre-conversion backups, reactivating legacy-to-target bridges for phased deployments, and initiating the fall-forward plan if rollback is not viable. All rollback actions MUST be logged and reported to OHIP (Section 5.8.2).	Deployment Plan ⁴⁴
Data Retention and Archival	The Contractor MUST maintain the legacy solution or a read-only copy of its database accessible for the period defined by the Department following confirmed production cutover. Temporary staging areas and intermediate files MUST be purged within an agreed-upon timeframe established between the Contractor and OHIP prior to production cutover, unless the data is subject to a legal hold or OHIP-directed retention requirement. The agreed-upon purge schedule MUST be documented in the Data Conversion and Migration Plan and confirmed in writing by OHIP. Evidence of purge completion MUST be provided to OHIP upon execution (Section 5.8.4).	Post-Conversion Data Retention documentation; purge confirmation evidence

5.6.2 Transformation Standards

All data transformations **MUST** adhere to the following requirements:

- **Idempotence:** All transformation and load processes **MUST** be rerunnable without side effects. Identical input **MUST** produce identical output. This **MUST** be verified through dry-run testing.
- **Restartability:** All loads **MUST** support restart without data loss or duplication. Checkpoints and transactional boundaries **MUST** be defined. Proof of restartability **MUST** be provided via dry-run test results.
- **Determinism:** Non-deterministic functions (e.g., random value generation, GETDATE() or SYSDATE() in business key logic, sequence-dependent operations) **MUST** be avoided in conversion transformation logic. Where a non-deterministic function is technically unavoidable or mandated by the target system's design, the Contractor **MUST** document the specific function, its purpose, and its potential impact on reproducibility, and obtain approval from OHIP before implementation. Approved exceptions **MUST** be noted in the Data Conversion and Migration Plan and re-validated during each dry run.
- **Modularity:** Conversion jobs **MUST** be modular with parameterized, configuration-driven connectors. Retry and backoff logic **MUST** be implemented for external system calls.

⁴⁴ For more information, refer to the OHIP Solution Delivery Domain Document, SDLC Standard.

- **Logging:** Every conversion job **MUST** log: job ID, run ID, start/end timestamps, record counts (extracted, transformed, loaded, rejected), reject reasons by category, latency metrics, and error codes.

5.6.3 Data Acceptance Criteria

Every data conversion **MUST** have clear, measurable, and testable acceptance criteria that are defined and approved by OHIP before conversion begins. The following criteria represent minimum thresholds. Contracts **MAY** specify tighter thresholds but **MUST NOT** loosen these minimums.

Table 30 – Data Acceptance Criteria

Criterion	Minimum Requirement
Data Profiling and Source Readiness	<ul style="list-style-type: none"> • Profile 100% of in-scope sources at the attribute level before design. • Produce baseline data quality scorecards for each domain. • Submit an approved CAP⁴⁵ for any domain below agreed thresholds; no build begins without an approved plan.
Mapping and Transformation Specifications	<ul style="list-style-type: none"> • Map 100% of attributes with business rules, transformation logic, defaulting, survivorship rules, and reference data crosswalks documented at the column level. • Document all transformation logic applied, including field-level mapping specifications, business rule definitions, derivation formulas, and conditional logic. Transformation documentation MUST enable independent verification and reproduction of conversion results. • Include source-to-target lineage for 100% of mappings. • Obtain OHIP sign-off on mapping specifications before development begins.
Completeness	<ul style="list-style-type: none"> • Mandatory fields: 100% non-null. • Conditionally required fields: ≥ 99.5% non-null when the condition is met.
Accuracy (field-by-field comparison)	<ul style="list-style-type: none"> • Critical elements: ≥ 99.90% match. • Important elements: ≥ 99.50% match.
Conformity (data type, format, domain values, code sets)	<ul style="list-style-type: none"> • Critical elements: 100% conformant. • Important/Standard elements: ≥ 99.50% conformant.
Uniqueness (by business keys)	<ul style="list-style-type: none"> • Duplicate rate: ≤ 0.05% for critical entities. • Duplicate rate: ≤ 0.10% for others.

⁴⁵ For more information, refer to the OHIP Security, Privacy, and Compliance Domain Document, Security and Privacy Compliance Standard.

Referential Integrity	<ul style="list-style-type: none"> • Parent-child integrity: 100% enforced with 0% orphan rate. • Any unavoidable exceptions MUST be quarantined to exception tables with documented rationale.
Validity (against reference/master data)	<ul style="list-style-type: none"> • Valid code/lookup: $\geq 99.90\%$ for critical entities. • $\geq 99.50\%$ for others.
Timeliness and Currency	<ul style="list-style-type: none"> • Snapshot-to-load lag: within agreed window (e.g., ≤ 24 hours in non-cutover). • During cutover: within the approved blackout window.
Reconciliation and Financial Balancing	<ul style="list-style-type: none"> • Record counts: 100% match between source extract and target landing. Any deltas MUST be explained, logged, and approved. • Financial balances and key aggregates: differ by $\leq 0.01\%$ or \$100 (whichever is smaller) at agreed aggregation levels. • Pre/post conversion control totals MUST be signed off for each conversion lot.
Exception Handling	<ul style="list-style-type: none"> • Route 100% of rejects to an exception repository with root cause code, record keys, timestamps, and reproducible steps. • Exception resolution timeframes MUST be defined in project-specific requirements and approved by OHIP prior to conversion execution. All exceptions MUST be tracked, assigned, and resolved within the timeframes established in those requirements. Any exception that cannot be resolved within the agreed timeframe MUST be formally deferred with documented OHIP approval, including a remediation plan and target resolution date. • Zero “silent failures”: any job that drops or alters data without logging is a Severity 1 defect.
Performance and Window Compliance	<ul style="list-style-type: none"> • All production conversion jobs MUST complete within the approved cutover window. • Throughput targets MUST be met or exceeded as agreed. • All loads MUST support restart without data loss or duplication; proof via dry-run test results.
Auditability and Lineage	<ul style="list-style-type: none"> • 100% of target records MUST carry audit metadata (source system, extract timestamp, load timestamp, job ID, transformation version). • Column-level lineage MUST be captured and queryable for 100% of converted attributes. • Any rule/version change MUST be traceable to the deployment package and approval record.

5.6.4 Reconciliation and Balancing

The Contractor **MUST** implement, at minimum, the following reconciliation controls at every stage of the conversion process. OHIP **MAY** require additional controls based on the complexity or sensitivity of the data domain:

- **Record Count Controls:** Record counts **MUST** be verified by entity and key subgroups at each conversion stage (extract, transform, load). Any variance **MUST** be immediately investigated and logged.
- **Reconciliation Reports:** Automated reconciliation reports **MUST** be produced for each conversion lot. Reports **MUST** include source counts, target counts, variance analysis, and root cause for any discrepancies.
- **Sign-Off:** Data Owners **MUST** validate and approve reconciliation results for each conversion lot before the lot is considered complete. Approval **MUST** be documented with the Data Owner's name, title, date, and the specific version of the reconciliation report reviewed. Reconciliation approval is a prerequisite for advancing to the next conversion lot.

The following controls **SHOULD** be applied:

- **Statistical Sampling:** Statistically valid samples per domain **SHOULD** be selected for manual verification to complement automated checks.

The following controls **MAY** be implemented:

- **Control Totals:** Numeric sums (e.g., financial balances, benefit amounts) **MAY** be calculated at source and verified at target. Hash totals **MAY** be used for non-financial attributes where appropriate.

5.6.5 Exception Handling

All conversion exceptions **MUST** follow a structured resolution process, which includes and is not limited to:

1. Capturing exceptions in real-time during conversion execution with root cause codes, record keys, timestamps, and reproducible steps.
2. Routing exceptions to a centralized exception repository accessible to both the Contractor and OHIP teams.
3. Classifying exceptions by severity and assigned to the appropriate resolver based on root cause.
4. Re-processing resolved exceptions through the conversion pipeline and validating against acceptance criteria.
5. Escalating unresolved exceptions within defined SLA windows per the OHIP incident management process.⁴⁶

⁴⁶ Refer to the OHIP Infrastructure and Operations Domain Document, ITSM Standard for more information.

5.7 Testing and Validation

Conversion testing **MUST** follow the OHIP Solution Delivery Domain Document, Test Management Standard for general testing methodology. This section defines conversion-specific testing requirements that supplement the OHIP Solution Delivery Domain Document, Test Management Standard.

 **Note:** The OHIP Solution Delivery Domain Document, Test Management Standard defines general test types, defect severity classifications, and test management processes. This standard defines conversion-specific test requirements. Where this standard specifies a tighter requirement, the tighter requirement applies.

5.7.1 Conversion Trial Runs

The Contractor **MUST** perform a minimum number of successful end-to-end dry runs in a production-like environment before cutover, as follows:

- Standard conversions: minimum two (2) successful dry runs.
- Legacy system decommissioning or high-volume migrations (exceeding [TBD: confirm threshold with OHIP, e.g., 10M records or 5+ source systems]): minimum three (3) successful dry runs, unless OHIP approves a reduced number in writing.

The Contractor **MUST** obtain formal OHIP approval for the dry run plan, including the number of dry runs, scope per run, and exit criteria, before the first dry run is executed. Additional dry runs **MAY** be required by OHIP at any time if results reveal systemic quality issues.

5.7.2 Exit Criteria per Dry Run

- All data quality, reconciliation, and performance thresholds in [Section 5.6.3](#) **MUST** be met or exceeded.
- Zero Severity 1 defects remain open.
- ≤ 5 Severity 2 defects remain open, each with an approved workaround.
- All Severity 3+ defects are tracked with remediation dates.
- Conversion completes within the approved cutover window.

5.7.3 Data Validation Process

The Contractor **MUST** maintain a data validation checklist covering all acceptance criteria from [Section 5.6.3](#). Any record that fails validation **MUST** be routed to an error queue or exception report for remediation. Only data that passes all validation checks **MAY** be committed to the production database or forwarded to the receiving system.

5.7.4 End-to-End Verification

Converted data **MUST** be validated through all downstream endpoints and use cases, including:

- Reports and data deliveries that consume converted data
- Application screens and fields that display converted data
- Interfaces and data exchanges that transmit converted data to downstream systems

- Federal reporting submissions (e.g., Transformed Medicaid Statistical Information System [T-MSIS]) that depend on converted data

End-to-end verification **MUST** include involvement from both technical staff and end-users. End-users **MUST** perform their typical tasks on the new solution using converted data and confirm the data supports their business processes.

5.7.5 Automated Compliance Checking

The Contractor **SHOULD** use automated tools, when available, to check compliance with applicable standards (e.g., FHIR resource validation, X12 transaction validation). Continuous Integration/Continuous Delivery (CI/CD) deployment **SHOULD** be used for conversion deployments. Automated tools **SHOULD** be used for configuration management in conversion environments and for reconciliation and reporting.

5.7.6 Converted Data Identification and Integrated Validation

All records loaded into the target system through conversion **MUST** be identifiable as converted data. The Contractor **MUST** implement a tagging or flagging mechanism within the target system that distinguishes converted records from records created natively within the target system. The tagging approach **MUST** be approved by OHIP prior to the first dry run and **MUST** persist in the target system following production cutover.

Following any dry run or production cutover in which both converted data and natively-created target data coexist, the Contractor **MUST** perform integrated validation comparing the two populations. This validation **MUST** include:

- **Population Reconciliation:** Confirmation that record counts, key distributions, and domain value frequencies are consistent between converted and native data populations where consistency is expected, and that any differences are explainable and documented.
- **Anomaly Detection:** Identification of any values, formats, codes, or relationships present in converted data that deviate from patterns observed in native target data. All detected anomalies **MUST** be assessed for root cause and either resolved prior to production cutover or formally accepted with OHIP approval.
- **Business Process Verification:** End-users and OHIP staff **MUST** confirm that converted data and native target data behave equivalently within target system workflows, reports, and interfaces, with no adverse interaction between the two populations.

5.8 Cutover, Rollback, and Post-Go-Live

5.8.1 Cutover

The Contractor **MUST**:

- Develop a Deployment Plan⁴⁷ that encompasses the full strategic approach to production cutover, including communication planning, go/no-go governance, and rollback strategy.

⁴⁷ For more information on the Deployment Plan, refer to the OHIP Solution Delivery Domain Document, SDLC Standard.

- Provide a detailed Cutover Runbook with step-by-step tasks with owners, commands, checkpoints, start/end times, dependencies, success criteria, rollback triggers, and backout points for each step.
- Verify record counts and control totals between major conversion steps. Any variance beyond tolerance triggers a checkpoint review.

Acceptance Criteria and Sign-Off

Before production cutover **MAY** proceed, the following conditions **MUST** be met:

- All data quality, reconciliation, performance, and security thresholds from [Section 5.6.3](#) are met or exceeded.
- All test cases are successful or have an OHIP-approved remediation plan.
- No Severity 1 defects remain open. Exceptions within agreed tolerances are documented.
- OHIP Leadership has provided explicit go/no-go authorization.

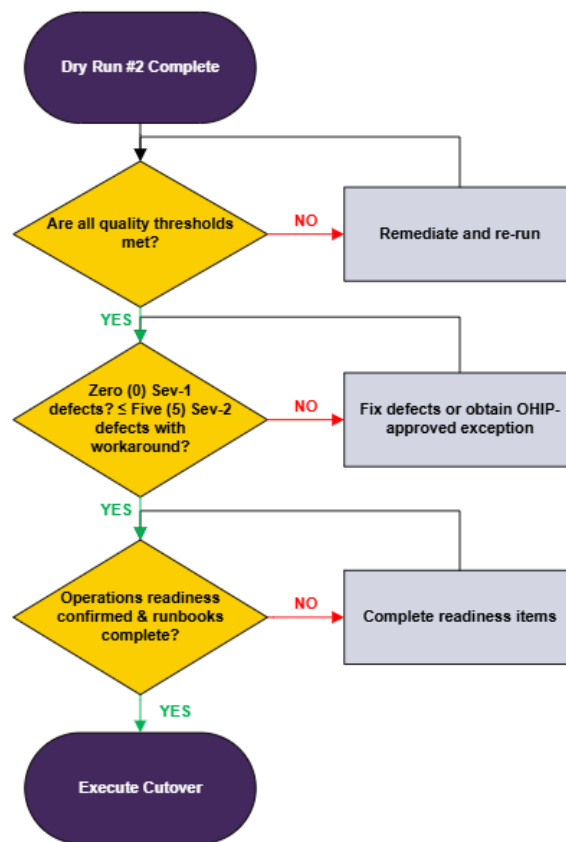


Figure 10 - Cutover Go/No-Go Decision Flow

5.8.2 Rollback and Contingency

The Contractor **MUST** develop and test a Rollback Plan before executing cutover activities. The Rollback Plan **MUST** include:

- **Rollback Triggers:** Defined conditions that initiate rollback, including quality threshold breaches, missed conversion window, and critical defects without workaround.

- **Pre-Conversion Backup:** Full backup of source and target solutions before initiating conversion. For ongoing interfaces, each batch file or message **MUST** be backed up before processing.
- **Post-Conversion Backup:** Full backup of the target solution once conversion is successfully completed and verified.
- **Tested Restore:** Verified backups with rehearsed restore procedures. Recovery Time Objective (RTO) and Recovery Point Objective (RPO) **MUST** be defined and tested (e.g., RTO ≤ 4 hours; RPO = 0 during conversion window).
- **Bridge Reactivation:** For phased deployments, the plan **MUST** include procedures to reactivate legacy-to-target bridges if rollback is required.
- **Backfill and Catch-Up:** A plan for handling deltas accrued during the blackout window, with measured reconciliation.

 **Note:** If rollback is not viable, the Contractor **MAY** define a Fall-Forward Plan with compensating controls and a remediation timeline in lieu of full rollback.

5.8.3 Hypercare and Post-Go-Live Stabilization

The Contractor **MUST** provide an enhanced support period (Hypercare) following production cutover for the duration defined in the applicable contract or as directed by the Department.

Hypercare Requirements

- **Monitoring:** The Contractor **MUST** develop data quality monitoring rules, which include dashboards, alerts, and automated data quality rules to monitor data health and process stability in production. Automated data quality monitoring rules for critical domains **MUST** be active and running, at minimum, daily. Alerts **MUST** be configured for quality threshold breaches, job failures, and performance degradation.
- **Conversion Defect Resolution:** The Contractor is responsible for correcting data errors introduced by the conversion process. Backfills **MUST** be performed with no downtime or minimal downtime as agreed, at no additional cost during the warranty period.
- **Stakeholder Reporting:** Weekly Hypercare status reports **MUST** be provided to OHIP, including data quality metrics, defect counts by severity, resolution times, and trend analysis.
- **Knowledge Transfer:** The Contractor **MUST** document post-go-live operational procedures, monitoring tasks, incident triage, escalation paths, and knowledge articles. Knowledge transfer **MUST** be completed and runbooks handed over prior to go-live, within the timeframe specified by the Department. Training **MUST** be delivered to OHIP staff (data operations, governance, and support). Knowledge transfer sessions **MUST** be completed with attendance and sign-offs recorded.
- **Operations Readiness:** Production Standard Operating Procedures (SOPs), on-call schedules, alert thresholds, and escalation paths **MUST** be in place before cutover.
- **Daily Checkpoints:** The Contractor **MAY** also conduct daily data quality and operational health checks during the initial Hypercare period, transitioning to a reduced cadence as agreed with OHIP and specified in the contract.

5.8.4 Post-Conversion Data Retention

The Contractor **MAY** keep the legacy solution (or a read-only copy of its database) accessible for a period defined by the Department, following confirmed production cutover, unless the contract specifies a longer or shorter period based on risk and data sensitivity. The retention period **MUST** be documented in the Data Conversion and Migration Plan and approved by OHIP during conversion planning. This provides a comparison source in case of discrepancies and supports data that was not converted. If maintaining the legacy solution is impractical, the legacy data **MAY** be exported to an archival format with indexing for lookup.

 **Note:** Post-conversion data retention periods **MUST** align with [Section 2.0](#). Specific retention periods **MUST** be confirmed with OHIP during conversion planning.

5.9 Audit, Reporting, and Transparency

5.9.1 Conversion Audit Trails

The Contractor **MUST** implement detailed audit logging for the conversion process. At minimum, audit logs **MUST** document:

- Who performed the conversion (operator identity)
- When it ran (start and end timestamps)
- What records were affected (counts, keys, scope)
- What transformation rules and versions were applied
- Any errors, exceptions, or manual adjustments made

Audit logs **MUST** be protected against unauthorized modification and retained per OHIP record retention policies. Conversion audit logs **MUST** be accessible for compliance audits by federal partners (e.g., during CMS SMC of MES solutions).

5.9.2 Reporting Requirements

The Contractor **MUST** provide the following reports to OHIP:

- **Data Quality Reports:** Conversion data quality assessment reports prior to user testing and on an ongoing basis after go-live.
- **Issues Log:** A log of all records failing data validation, with root causes and resolution status.
- **Reconciliation Reports:** Source-to-target reconciliation evidence for each conversion lot, including record counts, control totals, variance analysis, and approvals.
- **Final Conversion Report:** A comprehensive report summarizing the entire conversion project, including outcomes, lessons learned, data quality achieved, and final approvals.
- **Post-Production Compliance Dashboard:** A dashboard showing the status of data exchange service levels, timeliness, and error rates. The dashboard **MUST** visually flag when interfaces are down or when data quality falls below thresholds.

5.9.3 Conversion Transparency

At OHIP's request, all conversion activities, data, environments, artifacts, and related materials **MUST** be provided and demonstrated. All vendor-provided conversion tools **MUST** be available and licensed (with support and updates) to OHIP staff. The Contractor **MUST** provide knowledge transfer and documentation on conversion tools to OHIP.

5.9.4 Compliance Audits

The Contractor **MUST** be prepared for audits by federal partners on data exchanges and conversions. The Contractor **MUST** maintain documentation of conversion processes, mapping specifications, validation results, and security measures. This documentation, along with audit trails, **MUST** be sufficient to demonstrate adherence to standards and policy and facilitate certification reviews (e.g., during CMS SMC of MES solutions).

5.10 Security and Privacy

All data conversion activities **MUST** comply with the OHIP Security, Privacy, and Compliance Domain Document, Security and Privacy Compliance Standard, which is the authoritative source for encryption, access control, authentication, vulnerability assessment, incident response, and HIPAA compliance requirements. This standard does not duplicate those requirements.

In addition to the OHIP Security, Privacy, and Compliance Domain Document, Security and Privacy Compliance Standard, the following conversion-specific security requirements apply:

- **Data Minimization:** The Contractor **MUST** convert and exchange only the minimum necessary data required for the specific business purpose.
- **Non-Production Data Governance:** Non-production conversion environments **MUST** have 100% masking, tokenization, or de-identification of PII, PHI, Payment Card Industry (PCI) data, FTI, and MCD in accordance with the OHIP Security and Privacy Compliance Standard. No raw production data containing any of these data classifications **MAY** be present in non-production environments without explicit written approval from the OHIP Security and Privacy Officer.
- **Staging Environment Security:** Any intermediate files or staging databases used during conversion **SHOULD** be secured using equivalent or risk-appropriate controls commensurate with data sensitivity
- **De-identification:** When feasible and appropriate for development, testing, or research purposes, PHI **MUST** be de-identified or anonymized in accordance with HIPAA standards and the OHIP Security and Privacy Compliance Standard's de-identification requirements.
- **Intermediate File Purge:** Temporary staging areas and intermediate files containing sensitive data **MUST** be purged within timeframes defined in project-specific requirements and approved by OHIP.

 **Note:** For all other security and privacy requirements, refer to the OHIP Security, Privacy, and Compliance Domain Document, Security and Privacy Compliance Standard. Any conversion-related security incidents **MUST** be handled per the OHIP Infrastructure and Operations Domain Document, ITSM Standard's incident management process.

6.0 Data Lakehouse Architecture Standard

6.1 Introduction

This Data Lakehouse Architecture Standard establishes mandatory requirements for the design, implementation, and operation of Data Lakehouse environments within OHIP. The standard ensures that the MES can support high-scale analytics, machine learning, regulatory reporting, and operational intelligence while maintaining the data integrity, security, and auditability required for Health Insurance Portability and Accountability Act (HIPAA) governed health data.

The Data Lakehouse paradigm combines the flexibility and scale of data lakes with the reliability and governance capabilities of traditional data warehouses. This approach enables OHIP to store vast quantities of raw data economically while also providing the structured, transactional capabilities required for accurate reporting and analytics. By establishing clear architectural layers, quality gates, and governance requirements, this standard ensures that data flowing through OHIP systems remains trustworthy from ingestion through consumption.

6.2 Objectives

The primary objectives of this standard are to safeguard data integrity and trustworthiness by:

- Preserving raw data immutably and enforcing governed transformation processes.
- Establishing a layered architecture that isolates raw source data from transformed business data and curated analytics datasets, ensuring changes at one layer do not compromise data at other layers.
- Enabling scalable analytics by providing architectural patterns that support both traditional business intelligence and advanced machine learning workloads.
- Supporting regulatory compliance by maintaining complete data lineage and audit trails required for federal reporting including Transformed Medicaid Statistical Information System (T-MSIS) submissions to CMS.
- Promoting interoperability by requiring open table formats that prevent Contractor lock-in, enable diverse compute engines to access OHIP data, and support health care data exchange standards including HL7 FHIR.
- Establishing quality assurance through mandatory data quality gates that prevent corrupt or incomplete data from contaminating downstream analytics and reports.
- Ensuring robust security and governance through granular access control, multifactor authentication, encryption at rest and in transit, dynamic data masking for sensitive data elements, and comprehensive audit logging to protect PHI, PII, and other regulated data.

This standard applies to all data stored in or processed through the MES cloud environment. The Contractor **MUST** adhere to these architectural patterns for any solution that ingests, processes, stores, or serves Medicaid data. The standard governs data ingestion pipelines that bring data into the Lakehouse environment, transformation processes that cleanse and conform data for analytical use, data storage structures including table formats and partitioning schemes, data quality validation and enforcement mechanisms, metadata management and data cataloging, and consumption layer design for analytics and reporting.

6.3 Functional Responsibilities

Successful implementation of a Data Lakehouse requires clear assignment of roles and responsibilities. The Contractor **MAY** use different job titles but **MUST** assign the functional roles defined in this standard to specific individuals or teams. OHIP will verify that the roles in Table 33 map appropriately to the organizational structure and document specific job titles that fulfill each function.

Table 31 – Data Lakehouse Functional Responsibilities

Functional Area	Responsibilities	Accountable For	Decision Rights
Security Administration	Manage access roles and privileges. Monitor for access abuse and vulnerabilities. Configure row-level and column-level security policies. Oversee authentication and authorization controls.	Access control integrity and security policy enforcement	Grant or revoke role assignments. Approve security policy changes.
Data Architecture	Design Lakehouse architecture. Define layer boundaries. Establish table formats and partitioning strategies.	Architectural integrity and scalability	Approve architectural designs and deviations.
Data Engineering	Build ingestion pipelines. Implement quality gates. Manage ETL/ELT processes.	Pipeline reliability and data quality enforcement	Deployment to non-production environments.
Data Stewardship	Define data quality rules. Approve remediation approaches. Maintain data dictionary.	Data quality and business rule accuracy	Approve quality thresholds and exception handling.
Data Ownership	Authorize data access. Define retention requirements. Approve data classifications.	Data security, compliance, and appropriate use	Grant or revoke data access permissions.
Platform Administration	Manage cloud infrastructure. Configure compute resources. Monitor system health.	Platform availability and performance	Infrastructure provisioning and scaling.
Program Oversight	Set architectural priorities. Approve external integrations. Oversee compliance.	Solution compliance and strategic alignment	Final approval for production deployments.

6.3.1 Data Consumer Functional Responsibilities

In addition to the platform governance functions defined above, the Contractor **MUST** establish data consumer access functions that govern how end-users access and utilize Lakehouse data. These functions operate within the access controls enforced by the platform but do not participate in platform governance or data engineering activities.

Data Consuming Entity (DCE)

A Data Consuming Entity is an organization-based group of users authorized to request, schedule, and receive data from the Lakehouse environment. The Contractor **MUST**:

- Register each DCE in the Data Catalog with designated points of contact and approved data access scope.⁴⁸
- Document the business justification for data access, the specific datasets and data elements required, the intended use of the data, and any downstream systems or processes that will receive the data for each registered DCE.

Consumer Access Tiers

Data consumers are assigned to access tiers based on their role requirements and data sensitivity authorizations. The access tiers in Table 34 apply to Lakehouse data consumption:

Table 32 – Consumer Access Tiers

Access Tier	Description	Typical Use Cases
Standard Access	Access to de-identified or aggregated data only; no PHI or PII visibility	Public reporting, trend analysis, operational dashboards
PHI View	Access to datasets containing PHI for authorized purposes	Clinical analytics, care management, quality measurement
PHI Export	PHI View permissions plus ability to extract data for approved downstream use	Research datasets, regulatory submissions, authorized data sharing
Gatekeeper	Administrative access to approve or deny data access requests within a designated domain	Data governance, access management, compliance oversight

Access tier assignments **MUST** be approved by the Data Owner for each dataset and enforced through row-level and column-level security policies as defined in [Section 7.15.4](#). The Contractor **MUST** implement technical controls to prevent users from accessing data beyond their authorized tier.

6.4 Layered Data Architecture

The OHIP Data Lakehouse employs a three-layer architecture to enable controlled data transformation. This layered approach ensures that raw data is preserved for auditability, transformations are applied systematically, and consumption-ready data meets quality standards. The Contractor **MUST** implement this architecture for any solution that stores or processes Medicaid data.⁴⁹

⁴⁸ Refer to the OHIP Security, Privacy, and Compliance Domain Document, Security and Privacy Compliance Standard for more information.

⁴⁹ For more information, refer to [Section 5.0](#).

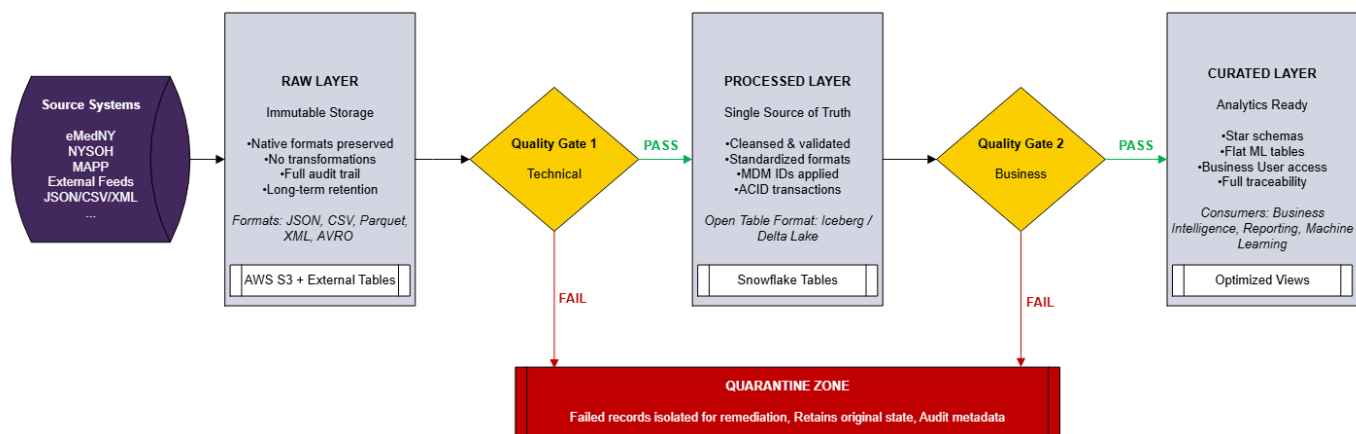


Figure 11 - OHIP Data Lakehouse Architecture

6.4.1 Raw Layer

The Raw Layer⁵⁰ serves as the initial entry point for all data entering the Lakehouse environment. This database functions as an immutable historical record, preserving data exactly as it was received from source systems. The Raw Database is essential for auditability, enabling OHIP to demonstrate exactly what data was received, when it was received, and from which source system it originated.

In the Raw Layer, the Contractor **MUST**:

- Store all incoming data in its native format without modification. Whether data arrives as JavaScript Object Notation (JSON), Comma-Separated Values (CSV), Parquet, Extensible Markup Language (XML), or any other format, it **MUST** be preserved exactly as received.
- Ensure no business transformations, data cleansing, or field modifications are performed in this layer. Even corrections to clearly erroneous data **MUST NOT** be applied in the Raw Layer; such corrections occur in subsequent layers, while the original record remains intact.
 - Format conversion for storage optimization (e.g., to Parquet) is permitted, provided data content and structure remain unmodified.
- Comply with OHIP data retention policies.⁵¹
- Ensure each data load into the Raw Layer includes ingestion metadata documenting the source system, extraction timestamp, file format, record count, and data sensitivity classification. Specific metadata requirements are detailed in [Section 3.0](#) of this standard. Schema versioning is covered in [Section 7.11](#). Batch IDs are captured via Ingestion_Job_ID. File size and location are operational metadata captured per [Section 7.10](#).

⁵⁰ The Raw Layer refers to the initial storage tier where data is preserved exactly as received from source systems, with no transformations applied. This Layer serves as the immutable historical record and enables full reprocessing when business rules change.

⁵¹ For more information, refer to the OHIP Infrastructure and Operations Domain Document, Log Management Standard.

6.4.2 Processed Layer

The Processed Layer⁵² serves as the single source of truth for core business entities within the OHIP data ecosystem. In this database, data that has passed the Raw-to-Processed quality gate undergoes standardization, validation, and enrichment to create consistent, reliable datasets that can be trusted for operational and analytical purposes.

Within the Processed Layer, the Contractor **MUST**:

- Convert data into an open table format⁵³ that supports Atomicity, Consistency, Isolation, and Durability (ACID) transactions.
- Apply standardization transformations consistently across all data entering this database.
 - Convert dates to International Organization for Standardization (ISO) 8601 format.
 - Standardize and validate addresses against postal standards.
 - Normalize currency values to consistent precision.
 - Map code values to enterprise-standard reference data.
 - Apply identical transformation rules uniformly across all instances of a data type (e.g., all date fields use ISO 8601 format, all Boolean fields use the same true/false representation, and all currency fields use the same decimal precision).⁵⁴
- Resolve and apply Master Data Management identifiers. Every member record **MUST** be assigned the appropriate member identifiers such as Client Identification Number (CIN). Provider records **MUST** be linked to appropriate provider identifiers such as National Provider Identifiers (NPI).
- Document the record linkage methodology used for each data domain, including the matching algorithm, matching fields, confidence thresholds, and procedures for resolving ambiguous matches. Record linkage results **MUST** be auditable.
- Maintain linkage audit logs that capture match decisions, confidence scores, and any manual overrides.
- Maintain crosswalk tables mapping source identifiers to enterprise master identifiers that will be made available to OHIP upon request.

 **Note:** Apache Iceberg is the preferred format for the current OHIP implementation, though Delta Lake **MAY** be used through the exception process defined by OHIP if technical requirements cannot be met with Iceberg.

⁵² The Processed Layer represents the standardized data tier where data has been validated, cleansed, and conformed to enterprise standards. This layer serves as the single source of truth for core business entities and applies Master Data Management identifiers.

⁵³ An Open Table Format refers to storage formats such as Apache Iceberg or Delta Lake that provide ACID transaction capabilities while remaining readable by multiple compute engines without proprietary lock-in. ACID Transactions ensure that database operations are Atomic, Consistent, Isolated, and Durable, preventing partial updates and data corruption.

⁵⁴ For more information on data transformation, refer to [Section 5.0](#).

6.4.3 Curated Layer

The Curated Layer⁵⁵ is optimized for consumption by end-user tools including business intelligence platforms, reporting applications, and machine learning frameworks. The Curated Layer includes data that has passed all quality gates and has been modeled into structures to support efficient analytical queries and accurate reporting.

In the Curated Layer, the Contractor **MUST**:

- Model data into structures appropriate for its intended use.
 - o For traditional business intelligence and reporting workloads, star schemas consisting of fact tables surrounded by dimension tables, provide optimal query performance and intuitive navigation for business users.
 - o For machine learning and advanced analytics workloads, flat, denormalized tables are preferred. A one-row-per-entity structure simplifies joins and improves model training performance.
- Restrict Curated Layer access to technical personnel responsible for data engineering and platform administration. This access separation ensures that business users work with validated, consumption-ready data rather than intermediate processing artifacts.⁵⁶
- Maintain complete traceability for curated datasets back to their source records in the Raw Layer to support audit inquiries, enable investigation of data anomalies, and provide the foundation for demonstrating data provenance to federal regulators.^{57,58}

6.5 Data Ingestion Standards

To ensure auditability and enable automated ingestion processing, every data load into the Raw Layer **MUST** be accompanied by comprehensive metadata. This ingestion manifest provides the Lakehouse with essential context about what data was received, when it was received, and from where it originated. The manifest supports troubleshooting, enables quality monitoring, and provides the foundation for data lineage tracking.

6.5.1 Mandatory Ingestion Metadata

The Contractor **MUST**:

- Include the mandatory metadata fields, regardless of source system, data format, or ingestion frequency, with every data load.
- Apply the mandatory metadata fields, as shown in Table 35, to all ingestion patterns.
- Capture metadata at the micro-batch or transaction batch level, rather than the file level, for streaming and Change Data Capture ingestion.

⁵⁵ The Curated Layer is the consumption-ready tier optimized for end-user analytics and reporting. Data in this layer is modeled into structures appropriate for business intelligence tools, including star schemas and wide flat tables for machine learning applications.

⁵⁶ For more information, refer to the OHIP Security, Privacy, and Compliance Domain Document, Security and Privacy Compliance Standard.

⁵⁷ For more information on data lineage, refer to [Section 7.8](#).

⁵⁸ Refer to [Section 7.10](#) for data catalog requirements that enable business discovery and understanding.

Table 33 – Mandatory Metadata Fields

Field Name	Type	Description
Source_System_ID	String	Unique identifier of the originating system (e.g., eMedNY, NYSOH, Medicaid Analytics Performance Portal [MAPP]). MUST match the registered system ID in the OHIP data catalog.
Extraction_Timestamp	ISO 8601	The exact date and time when the data was extracted from the source system, which MUST include time zone designation.
Ingestion_Job_ID	UUID	Unique identifier of the pipeline execution that performed the ingestion to enable troubleshooting and correlation with pipeline logs.
File_Format	Enumeration (enum)	Native format of the source data. Valid values: JSON, CSV, PARQUET, AVRO, XML, FIXED_WIDTH, OTHER. *Formats not explicitly listed require OHIP approval for exceptions.
Record_Count	Integer	Total count of records transmitted in this data load, which are used for reconciliation and completeness validation.
Data_Sensitivity	Enum	Classification of the data type for the ingested dataset. Determines baseline security controls applied at the platform layer. Valid values: PHI, PII, MCD, INTERNAL (non-public operational data with no special regulatory designation), PUBLIC (approved for unrestricted public release). A single dataset MAY carry multiple values; use a pipe-delimited list (e.g., PHI PII). Aligns with ITS Information Classification data type indicators.
Compliance_Indicator	Enum (multi-value)	One or more regulatory or program compliance flags applicable to the dataset. Separates legal/program obligations from data type classification, consistent with the ITS Information Classification framework. Each compliance indicator carries its own governance, auditing, and access control requirements that supplement the baseline controls determined by Data_Sensitivity. ⁵⁹ Leave blank or set to NONE if no specific compliance program

⁵⁹ Valid values include: FTI (IRS Federal Tax Information—governed by IRS Publication 1075; requires safeguard review), SSA (Social Security Administration data—governed by Social Security Act §1106; requires SSA DUA and audit controls), ACA (Affordable Care Act data—governed by ACA §1561/HITECH; requires additional privacy protections for Exchange data), HIPAA (HIPAA-covered PHI subject to the Privacy and Security Rules), 42-CFR-PART-2 (Substance Use Disorder treatment records—governed by 42 CFR Part 2; stricter re-disclosure restrictions than HIPAA), HIVSID (HIV-related information—governed by NYS Public Health Law §2782; strict confidentiality and disclosure limits), MH (Mental Hygiene data—governed by NYS Mental Hygiene Law §33.13; disclosure restrictions apply), MARS-1 (Medicaid Analytic eXtract/MARS data—CMS data use agreement (DUA) required).

		applies. Use a pipe-delimited list when multiple indicators apply (e.g., FTI SSA).
Checksum	String	SHA-256 hash of the source file. Enables verification that data was not corrupted or modified during transmission.
Manifest_Version	String	For schema evolution of the manifest itself
Source_File_URI	URI	Enables traceability to source
Dataset_Name	String	Identifies the data being described

6.5.2 Ingestion Pipeline Requirements

For ingestion pipelines, the Contractor **MUST** enforce the architectural requirements defined in this Standard. This includes requirements to:

- Design pipelines for reliability, observability, and recoverability.
- Ensure all ingestion pipelines are idempotent, so that re-running the same pipeline, with the same input, produces the same result, without creating duplicate records. This property enables safe retry behavior when pipeline failures occur and supports reprocessing scenarios without manual intervention to clean partial results.
- Design pipelines to achieve idempotent outcomes.
 - o For streaming ingestion where at-least-once delivery **MAY** create duplicates in the Raw Layer, de-duplication **MUST** occur at the Raw-to-Processed quality gate⁶⁰ to ensure idempotent results in downstream layers.
 - o The combination of raw storage plus Quality Gate 1 de-duplication achieves the required idempotent outcome.
- Include comprehensive logging that captures the start and end times of each processing stage.
- Count records at each transformation step, when an error or warning is encountered, and for resource utilization metrics. These logs **MUST** be retained for a minimum of ninety days (90) and **MUST** be accessible to OHIP for troubleshooting and audit purposes.⁶¹
- Align Standard Service Level Agreement (SLA) timeframes with OHIP operational standards and CMS reporting requirements. Specific regulatory requirements are specified in [Section 7.12](#).
- Implement automated monitoring and alerting for ingestion pipelines.
 - o Alerts **MUST** be triggered when pipelines fail to complete within expected timeframes, when record counts deviate significantly from historical norms, when error rates exceed defined thresholds, or when quality gate failures occur.
 - o Alert notifications **MUST** be routed to the appropriate personnel and integrated with the MES Incident Management System as defined in the OHIP Infrastructure and Operations Domain Document, Information Technology Service Management (ITSM) Standard.

⁶⁰ For more information, see [Section 7.6.1](#).

⁶¹ For more information, refer to the OHIP Infrastructure and Operations Domain Document, Log Management Standard.

6.5.2.1 Ingestion Pattern Requirements

The Contractor **MUST** implement ingestion patterns that comply with metadata and idempotency requirements and are appropriate to the data source and latency requirements defined below:

- **CDC/Near Real-Time Ingestion:** For Oracle sources, Oracle GoldenGate or equivalent CDC tools **MAY** be used. For PostgreSQL and other databases, AWS Database Migration Service (DMS) with S3 targets is the preferred approach.
- **Platform-Native Connectors:** Low-code platforms (e.g., Salesforce) with native Snowflake connectors **MAY** use those connectors directly, subject to metadata and audit requirements defined in [Section 7.5.1](#).
- **Batch File Ingestion:** Snowpipe is the preferred mechanism for batch file ingestion from S3 staging areas.
- **Streaming Ingestion:** For event-driven architectures, AWS Kinesis or equivalent streaming services **MAY** be used with appropriate de-duplication at the quality gate.

6.6 Data Quality Gates

A Data Quality Gate⁶² is a mandatory validation checkpoint that data **MUST** pass before being promoted from one architectural layer to the next to prevent corrupt, incomplete, or non-conforming data from contaminating downstream analytics and reports. Quality gate validation operates at the record level. Records passing validation proceed to the next layer; records failing validation are routed to the Quarantine Zone. This partial-batch handling ensures data quality issues do not block entire datasets from processing. The Contractor **MUST** implement all quality gates defined in this standard and **MUST NOT** promote data that fails to meet the specified thresholds.

 **Note:** This section defines the Data Quality Gates: mandatory validation checkpoints at each stage of the data pipeline that prevent non-conforming data from advancing to downstream consumers. The term 'gate' is used deliberately: unlike optional review steps, a quality gate enforces a pass/fail decision. Data that does not meet the defined thresholds is held for remediation, not passed through with exceptions.

⁶² A Data Quality Gate is a mandatory validation checkpoint that data must pass before being promoted from one architectural layer to the next. Gates enforce technical integrity and business logic requirements.

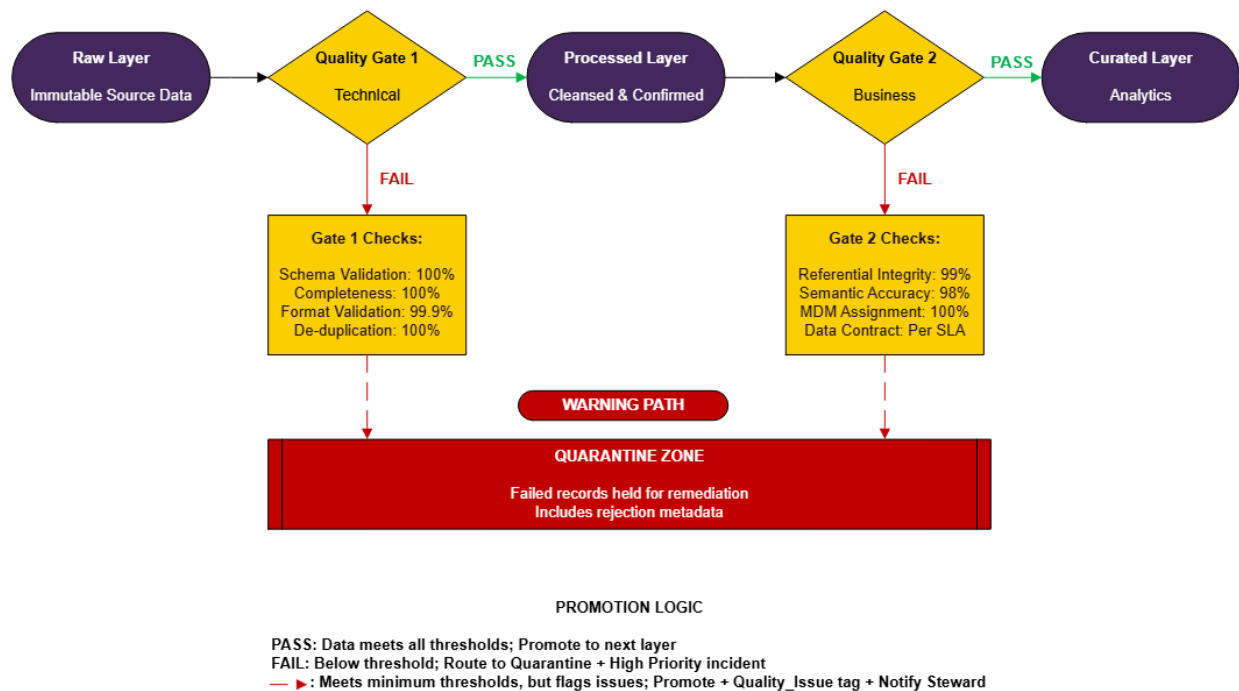


Figure 12 - Data Quality Gate Process Flow

6.6.1 Quality Gate 1: Raw to Processed Layer

The first quality gate ensures incoming data is structurally sound and fits the target schema before transformation processing begins. Records failing any validation with a 100% threshold are routed to the Quarantine Zone. Table 36 specifies the validation requirements and thresholds for Quality Gate 1, which are further outlined below:

- Schema validation prevents downstream processing errors by confirming data structure alignment with the target catalog definition before any transformations occur. Schema mismatches (including incorrect field names, data types, or nullability) constitute failure conditions.
- Completeness validation ensures that records can be reliably processed and linked to other datasets. Records missing primary keys or critical timestamps cannot support accurate reporting or analytics.
- Format validation confirms that field values conform to expected patterns. The 99.9% threshold accommodates minimal edge cases that can be corrected during the cleansing phase rather than blocking ingestion.
- De-duplication prevents duplicate records from inflating counts or skewing analytics. Duplicates are handled according to Data Steward rules established for each data domain.

Table 34 – Quality Gate 1: Data Validation Requirements and Thresholds

Check Category	Validation Requirements	Threshold
Schema Validation	Field names, data types, and nullability MUST match target catalog definition	100%

Completeness	Primary keys and mandatory timestamps MUST be non-null	100%
Data Format	Dates MUST conform to ISO 8601; numeric fields MUST NOT contain alpha characters	99.9%
De-duplication	Exact physical duplicates identified and handled based on business key	100%

6.6.2 Quality Gate 2: Processed to Curated Layer

The second quality gate ensures business integrity, validating that data is semantically correct and properly integrated with the broader Medicaid Enterprise data ecosystem before promotion to the Curated Layer. Table 37 specifies the validation requirements and thresholds for Quality Gate 2, which are further outlined below:

- Referential integrity validation confirms that transaction records link to valid dimension records. The 99% threshold acknowledges that some orphaned records **MAY** result from timing differences in data availability across systems.
- Semantic accuracy validation ensures that field values fall within business-defined reasonable ranges (i.e. Age values below 125 years, transaction dates not in the future, and paid amounts not negative unless permitted by transaction type). The 98% threshold accommodates legitimate edge cases while catching systematic data quality issues.
- Master Data Management validation confirms assignment of OHIP-standard master identifiers. This validation requires 100% compliance because master identifiers are essential for cross-system data integration.
- Data contract compliance validates that data meets specific completeness and accuracy levels defined by the Solution Owner for each data domain. These contracts document additional business rules beyond standard thresholds.

Table 35 – Quality Gate 2: Data Validation Requirements and Thresholds

Check Category	Validation Requirements	Threshold
Referential Integrity	Transaction records MUST link to valid dimension records (Provider, Member)	99%
Semantic Accuracy	Values MUST fall within business-defined reasonable ranges	98%
MDM Assignment	All records MUST have OHIP-standard master identifiers assigned	100%
Data Contract	Data MUST meet specific completeness levels defined by the Solution Owner	Per Contract

6.6.3 Promotion Logic

When data is evaluated against quality gate criteria, the result determines how the pipeline proceeds. Three outcomes are possible: Pass, Warning, and Fail, which are described below:

- A Pass result indicates that data meets all threshold requirements. The pipeline promotes the data to the next architectural layer and updates the metadata catalog with processing statistics and quality scores. No manual intervention is required.
- A Warning result indicates that data meets minimum thresholds but exhibits quality characteristics that warrant review. The pipeline promotes the data to the next layer, but flags affected records with a Quality_Issue tag. An automated notification is sent to the Data Steward for review. Processing continues without interruption, but the warning condition **MUST** be investigated and resolved to prevent accumulation of quality issues over time.
- A Fail result indicates that data does not meet minimum threshold requirements. The pipeline halts promotion of the affected batch. Records that pass validation proceed to the next layer while failing records are routed to the Quarantine Zone. A High Priority incident is logged in the MES ticketing system.⁶³ The remediation workflow defined in [Section 7.7](#) is initiated.

6.6.4 Data Quality Dimensions and Thresholds

The Contractor **MUST** assess data quality across the dimensions in Table 38.⁶⁴ Data criticality classification determines how strict quality requirements **MUST** be. The quality gate validation logic (Quality Gate 1: Raw to Processed, Quality Gate 2: Processed to Curated) **MUST** enforce these thresholds programmatically. Records failing threshold checks **MUST** be routed to the quarantine zone.

The Contractor **MUST** adhere to the minimum quality thresholds in Table 38:

The data quality dimensions below are defined in [Section 2.15](#). This table specifies the minimum quality thresholds by data criticality for Lakehouse quality gate enforcement.

Table 36 – Data Quality Thresholds by Criticality

Dimension	Critical Data	Important Data	Supporting Data
Completeness	100%	≥90%	≥80%
Accuracy	≥99%	≥95%	≥90%
Timeliness	<24 hours	<7 days	<30 days
Consistency	100%	≥98%	≥95%
Validity	100%	≥99%	≥95%
Uniqueness	100%	≥99%	≥98%
Integrity	100%	≥99%	≥98%

⁶³ For more information, refer to the OHIP Infrastructure and Operations Domain Document, ITSM Standard.

⁶⁴ Refer to [Section 2.0](#) for more information.

6.6.5 Data Quality Management Requirements

The Contractor **MUST** comply with the following data quality management requirements:

- **Routine Data Profiling and Quality Checks:** Implement scheduled data quality assessments on incoming data. At a minimum, with each monthly cycle of data, the Contractor **MUST** run a data-profiling process in addition to running automated editing checks that reflect the business rules above.
- **Data Quality Issue Logging:** Use a database table that maintains the history of data quality issues that can be queried for analysis and trends. Each issue entry **MUST** include a description (e.g., 'ten percent of addresses missing ZIP Code in January managed care file'), severity/priority, date identified, and status of resolution. The resolution history **MUST** provide a mechanism to log back the resolution details, including actions, dates, and target team/group.
- **Data Correction and Cleansing Procedures:** Adhere to a formalized process for identifying, tracking, and remediating data quality issues. Procedures **MUST** aim to fix errors at the source whenever possible. If it is not feasible, apply transformations or impute values within the analytic environment in a controlled and well-documented manner. Any data modifications **MUST** be auditable (e.g., keeping a column for 'imputed flag' or retaining original and new values). The Contractor **MUST** document all missing data patterns identified during quality gate processing. Missing data **MUST** be classified according to the mechanism of missingness: Missing Completely at Random (MCAR), Missing at Random (MAR), or Missing Not at Random (MNAR). For each dataset with significant missing data, the Contractor **MUST** document the classification of missingness, the method used to determine the classification, the handling strategy applied (e.g., imputation, exclusion, flagging), and the rationale for the selected approach. The Contractor **MUST NOT** apply imputation or exclusion methods without documented justification and OHIP approval for critical data elements.
- **Timeliness Monitoring:** Monitor and enforce timeliness in data availability by setting minimum thresholds. If data is missing or delayed beyond these thresholds, it is a quality issue that **MUST** be addressed with the responsible party.
- **Data Quality Management Tools:** Use a data quality profiling and validation tool⁶⁵ to automate complex checks.
- **Data Quality KPIs and Scorecards:** Develop KPIs for data quality and track such metrics over time. The Contractor **MUST** use scorecards to report on KPIs to program leadership⁶⁶ at a regular cadence as agreed upon by OHIP.
- **RCA and Preventative Action:** The Contractor **MUST** comply with the RCA and continuous improvement requirements defined in [Section 2.15.2](#).

⁶⁵ Data quality tools must be approved by OHIP prior to use. Refer to the OHIP Infrastructure and Operations Domain Document, Preferred Operational Tools Standard for more information.

⁶⁶ Refer to [CMS Outcomes Based Assessment](#) (OBA) for T-MSIS. States are assessed against targets (e.g., 100% of critical data elements complete, 99% of high-priority elements accurate).

6.6.6 Data Quality Measurement Methodology

6.6.6.1 Standardized Calculation Formulas

The Contractor **MUST**:

- Calculate each data quality dimension using standardized formulas to ensure consistent measurement across all OHIP systems.
- Automate calculations and ensure they are repeatable, using OHIP-approved data quality tools.
- Calculate accuracy (the number of correct values divided by the total number of values) as a percentage.
- Define validation rules that specify what constitutes a correct value for each data element, which **MAY** include range checks, format validation, or comparison against authoritative reference data.
- Measure completeness as the number of non-missing required values divided by the total number of expected values, expressed as a percentage.
- Maintain clear documentation of which fields are required versus optional.
- Calculate timeliness (the number of records updated within the acceptable time window divided by the total number of records) as a percentage. The acceptable time window is determined by the data criticality classification.
- Measure consistency as the number of records without conflicting data across sources divided by the total number of records, expressed as a percentage.
- Document which data elements are expected to be consistent across which systems and **MUST** establish rules for identifying conflicts.
 - Uniqueness is calculated as the number of unique records divided by the total number of records, expressed as a percentage, where duplicate detection **MUST** be based on defined business keys appropriate to each dataset.
- Calculate validity (the number of values conforming to defined formats or allowable domains divided by the total number of values) as a percentage.
- Maintain comprehensive validation rules defining acceptable formats and domains for each data element.
- Measure data integrity as the number of records with all required referential relationships intact divided by the total number of records, expressed as a percentage.
- Document all required referential integrity rules, including parent-child relationships and foreign key constraints that **MUST** be maintained across datasets.

6.6.6.2 Assessment Frequency and Scheduling

The assessment frequencies below implement the continuous monitoring principles established in [Section 2.15.3](#) for the Lakehouse environment.

When assessing data quality, the Contractor **MUST** implement automated measure processes to execute a schedule without manual intervention and ensure audit logs are maintained for all measurement activities. The Contractor **MUST** perform data quality assessments according to a defined schedule based on the criticality classification of each dataset. The Contractor **MUST** measure and monitor:

- Critical Data weekly, with automated alerts triggered immediately when measurements fall below established thresholds.
- High Priority Data monthly, with formal reporting of results to designated data stewards.
- Standard Data quarterly unless the nature of the data or its usage patterns warrant more frequent assessment.

6.6.6.3 Threshold Application Rules

For ongoing operational datasets that support continuous business processes, quality thresholds apply to the average measured performance over the assessment period rather than requiring every individual measurement instance to meet the threshold. The Contractor **MUST**:

- Calculate the average of all measurements taken during the period and compare this average against the established threshold.
- Document and investigate when any individual measurement period falls below the threshold, even if the average performance meets the threshold.
- Initiate appropriate corrective action within 10 business days of any threshold violation.
 - This approach acknowledges that temporary data quality fluctuations **MAY** occur while maintaining focus on sustained performance over time.
- Apply quality thresholds to each individual dataset instance rather than an average across multiple submissions for datasets received on a periodic basis from external sources, including federal reporting submissions such as T-MSIS and Contractor-provided deliverables.
 - Each submission represents a distinct deliverable with independent compliance obligations and **MUST** meet all applicable thresholds.
- Reject non-compliant dataset instances and return to the submitting organization for correction and resubmission.

Deliverables that fail to meet quality thresholds based on anticipated future improvement or average performance across multiple submissions **MUST NOT** be accepted by OHIP.

6.6.6.4 Documentation and Planning Requirements

The Contractor **MUST** document, within the Data Management Plan (Data Quality Management section), the comprehensive measurement approach for all datasets under the Contractor's control. Once OHIP approval has been obtained for the Data Management Plan, production measurement activities can begin. The plan **MUST** be updated annually at minimum or whenever significant changes occur to data sources, business processes, measurement tools, or organizational structure. Updated plans require resubmission to OHIP for approval before implementation.

6.6.6.5 Remediation Process

When quality measurements indicate that a dataset does not meet established thresholds, the Contractor **MUST**:

- Initiate a formal remediation process within two (2) business days of identification.
- Complete RCA within ten (10) business days to identify the underlying factors that caused the quality issue.

- Examine data collection processes, system interfaces, transformation logic, validation rules, and any other factors that **MAY** have contributed to the quality degradation.
- Submit a written CAP⁶⁷ to OHIP within 15 business days of the initial threshold violation. The CAP **MUST** specify remediation steps with assigned responsibilities, target completion dates for each step, and success criteria for determining when quality has been restored to acceptable levels.
- Complete remediation activities within thirty (30) days of the threshold violation unless an alternate timeline has been submitted to and approved by OHIP. In the event an extended timeline is necessary, the Contractor **MUST** provide clear justification demonstrating the extension is necessary and that interim measures have been implemented to mitigate the impact of the quality issue for OHIP to review.
- Perform follow-up measurement to verify data quality has improved to levels that meet or exceed the established thresholds upon completion of remediation activities.
- Document all quality issues, root cause analyses, CAPs, and remediation outcomes in the quality issue tracking system and make documentation available to OHIP upon request.

6.7 Data Remediation and Quarantine

In alignment with the OHIP Infrastructure and Operations Domain Document, ITSM Standard for incident management, any data failing a quality gate **MUST** be isolated to prevent contamination of downstream reports and analytics.⁶⁸

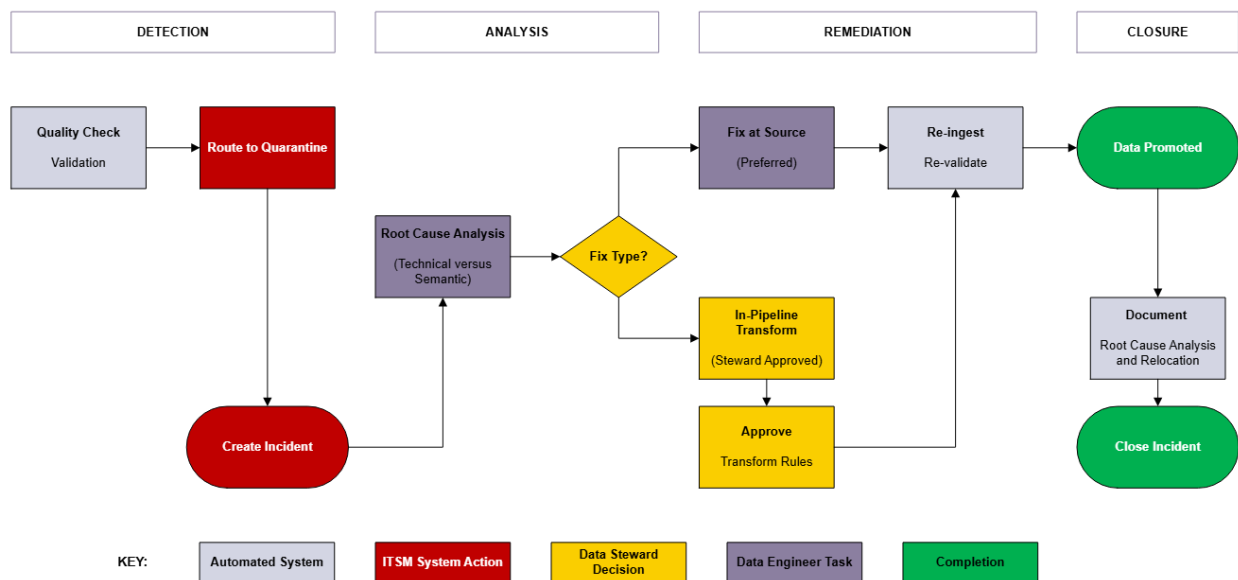


Figure 13 - Data Remediation Workflow

⁶⁷ For more information, refer to the OHIP Security, Privacy, and Compliance Domain Document, Security and Privacy Compliance Standard.

⁶⁸ For more information, refer to the OHIP Infrastructure and Operations Domain Document, ITSM Standard.

6.7.1 Quarantine Zone Requirements

The Contractor **MUST**:

- Retain the original raw state of quarantined records without modification.
- For each quarantined record, include a Rejection_Reason code that identifies which quality gate check failed, a Rejection_Timestamp that documents when the failure was detected, and the Ingestion_Job_ID that links the record back to the pipeline execution that attempted to process it.
- Restrict access to the Quarantine Zone to authorized personnel responsible for data quality remediation.
 - o Business users and analysts **MUST NOT** have access to quarantined data, as it has not passed validation and **MAY** contain errors that would corrupt analytical results.

The Contractor **MAY** implement the Quarantine Zone using any of the following mechanisms:

- Dedicated cloud storage prefix
- Separate database schema
- Another isolation mechanism approved by OHIP

6.7.2 Remediation Workflow

When a data quality violation occurs, the Contractor **MUST**:

- Execute a structured remediation workflow that identifies the root cause, implements appropriate corrections, and restores data flow.
- Integrate the workflow with the MES incident management processes defined in the OHIP Infrastructure and Operations Domain Standard, ITSM Standard.

Detection and isolation occur automatically when quality gate validation fails. The pipeline routes failing records to the Quarantine Zone while allowing conforming records to proceed through processing. This partial-batch handling ensures that data quality issues do not create complete pipeline blockages that would delay time-sensitive reporting.

Incident notification is triggered automatically when records are quarantined. An alert is sent to the Data Steward and Data Engineer responsible for the affected data domain. Incidents are classified by business impact rather than a fixed failure-rate threshold alone. The Contractor **MUST** assess each data quality incident and assign one of the following priority levels based on operational impact.

Remediation Timeframes

The Contractor **MUST** complete data quality remediation activities within the following timeframes based on incident priority. These timeframes align with the escalation triggers defined in the OHIP Infrastructure and Operations Domain Document, ITSM Standard.

Table 37 – Remediation Timeframes

Priority	Description	Initial Response	RCA	Remediation Complete
Critical	Data impacting regulatory reporting (T-MSIS, CMS-64, PERM), revenue-generating systems, or decisions with legal/financial consequences	2 Hours	4 Hours	24 Hours

High	Data affecting operational efficiency across multiple systems or significant gaps in core datasets	4 Hours	8 Hours	7 Business Days
Medium	Data affecting non-critical reporting or inaccurate data that does not halt business processes	1 Business Day	2 Business Days	4 Weeks
Low	Historical or rarely accessed data with minor quality issues not affecting business operations integrity	2 Business Days	5 Business Days	3 Months

Timeframes begin when the incident is logged into the MES ticketing system. Escalation procedures defined in the OHIP ITSM Standard apply when progress is not observed within the Initial Response window. If remediation cannot be completed within the specified timeframe, the Contractor **MUST** escalate to the Data Owner and provide a CAP with revised timeline for OHIP approval.

RCA is performed by the Data Engineer to determine whether the failure resulted from a technical issue such as a source schema change, network transmission error, or pipeline bug, or from a semantic issue such as invalid business data from the source system. The root cause determination guides the remediation approach.

The Contractor **MAY** accomplish remediation through either of the following approaches:

1. Fix the issue at the source, where the data producer corrects the underlying problem and resubmits corrected data in the next processing cycle.
2. Apply in-pipeline transformation when source correction is not feasible or would cause unacceptable delays and in-pipeline transformation is authorized by the Data Steward. Transformations **MUST** be documented, apply consistent rules, and **MUST NOT** mask underlying data quality problems that **MUST** be addressed at the source.

Re-ingestion occurs after the root cause has been addressed. Corrected data is reprocessed through the standard pipeline, including all quality gate validations. The re-ingestion **MUST** pass all gates before data is promoted to downstream layers.

Audit closure documents the complete remediation cycle. The incident record **MUST** capture the RCA findings, the remediation approach taken, any code changes or transformation rules applied, and confirmation that reprocessed data passed quality validation. This documentation supports audit inquiries and informs process improvements to prevent recurrence.

6.8 Data Lineage⁶⁹

The Contractor **MUST** document lineage across the Data Lakehouse architecture for critical data processes, enabling confidence in the data's reliability. Any data sent to CMS **MUST** be traceable through each stage of processing back to the raw source data. The Contractor's proposed method for documenting lineage **MUST** be approved by OHIP.

⁶⁹ The requirements in this section align with and extend the lineage requirements defined in [Section 6.0](#), which should be referenced for additional context on lineage within the broader OHIP analytics ecosystem.

6.8.1 Data Provenance

The Contractor **MUST** document data provenance for all datasets within the Lakehouse environment. To do this, the Contractor **MUST** capture the following for each dataset:

- The original source and SOR from which the data originated
- The collection context, which includes, and is not limited to the purpose, timeframe, and method of collection
- All processing steps applied to the data from collection through its current state
- The chain of custody identifying all systems, processes, and personnel that have handled the data.

The Contractor **MUST** also maintain provenance documentation in the data catalog and make it accessible to OHIP and designated auditors upon request. The provenance documentation **MUST** be sufficient to demonstrate compliance with applicable federal data submission requirements.

6.8.2 Lineage Documentation Requirements

The Contractor **MUST**:

- Develop and maintain high-level diagrams that illustrate the data flows across OHIP's systems. At a minimum, there **MUST** be an enterprise data flow diagram showing major systems and the interfaces between them including files, messages, and APIs. This diagram **MUST** cover all major solution data flows and highlight where data merges occur.
- Document the logic and input/output for each ETL job or data transformation process in the pipeline.
- Capture any filter conditions, joins, or derivations applied during transformation.
- Maintain a mapping of important data elements from source to target.

6.8.3 Version Control and Audit Trail

The Contractor **MUST**:

- Ensure all scripts, queries, or code that perform data transformations are under version control with appropriate change logs.
- Tie each dataset to a specific code version, enabling OHIP to reconstruct the exact transformation logic that produced any historical dataset.
- Ensure lineage information is readily available for auditing purposes. When auditors request documentation of how a particular report value was derived, the Contractor **MUST** be able to trace from the reported value back through each transformation stage to the original source record in the Raw Layer.

6.8.4 Lineage Tooling and Integration

The Contractor **MUST**:

- Employ a data lineage tracking tool that is approved by OHIP.⁷⁰ The selected tool **MUST** integrate with the data catalog and metadata repository to provide a unified view of data assets and their relationships. Manual lineage documentation in spreadsheets or documents is not acceptable for production systems; automated lineage capture is required.
- Perform an end-to-end reconciliation test to verify that lineage documentation accurately reflects actual data flows when major data transformations are developed, such as generating T-MSIS files or analytics datasets. These tests **MUST** trace sample records from source to target and verify that documented transformations match actual results.

6.8.5 Impact Analysis

Data lineage information **MUST** be used by entities making changes to perform impact analysis for proposed changes to data sources, systems, or transformation logic. For any proposed change, the Contractor **MUST**:

- Consult the lineage documentation to list all downstream objects that would be affected.
- Review and document the impact before changes are implemented to prevent unintended consequences to dependent systems and reports.

6.9 Data Versioning

The Data Lakehouse architecture **MUST** support data versioning to enable point-in-time queries, support audit requirements, and facilitate recovery from data quality issues.

6.9.1 Versioning Requirements

The Contractor **MUST**:

- Design and implement database structures that support data versioning.
- Use versioning capability that enables OHIP to query data as it existed at any prior point in time, supporting audit inquiries and regulatory compliance requirements.
 - At minimum, the versioning implementation **MUST** support querying historical snapshots by timestamp, comparing data between two points in time, and rolling back to a previous version if data corruption is discovered.
- Utilize table formats that provide native versioning capabilities in the Processed Layer and Curated Layer.
- Leverage Apache Iceberg and Delta Lake time travel functionality or demonstrate equivalent versioning capabilities and obtain OHIP approval for an alternative tool.

⁷⁰ Tool selection must align with the OHIP Infrastructure and Operations Domain Document, Preferred Operational Tools Standard where possible.

6.9.2 Version Retention

The Contractor **MUST**:

- Retain data versions in accordance with OHIP data retention policies.⁷¹
- Retain ninety (90) days of version history for the Processed Layer and Curated Layer to support operational troubleshooting and short-term audit requirements. Longer retention periods **MAY** be required for specific datasets based on regulatory requirements or business needs as specified by OHIP.

By design, the Raw Layer maintains permanent or long-term retention of all ingested data. Versioning in the Raw Layer is achieved through the preservation of distinct ingestion batches, rather than through table-level versioning mechanisms. Each ingestion batch is immutable and retains its identity throughout its retention period.

6.9.3 Version Access and Documentation

The Contractor **MUST**:

- Document the versioning capabilities available to data consumers and provide guidance on how to execute point-in-time queries. This documentation **MUST** include examples of common versioning use cases such as querying data as of a specific date, identifying records that changed between two dates, and understanding the version retention limits for each database layer.
- Govern access to historical versions using the same access controls that apply to current data.
 - Users with read access to a table **MAY** query historical versions of that table within the retention period.
- Require administrative access to perform version rollback operations that modify the current state of production tables.

6.10 Data Catalog and Metadata Management

The enterprise metadata taxonomy, including the four metadata categories (Business, Data Quality, Technical, and Operational) and their reconciliation with the OHIP Data Architecture and Naming Standard, is defined in [Section 3.0](#). The catalog attributes and ingestion metadata fields defined in this section represent the Lakehouse-specific implementation of that enterprise taxonomy. The Contractor **MUST** ensure that Lakehouse metadata satisfies both the enterprise requirements of [Section 2.10](#) and the Lakehouse-specific requirements below.

⁷¹ For more information, refer to the OHIP Infrastructure and Operations Domain Document, Log Management Standard.

6.10.1 Data Catalog Requirements

A comprehensive data catalog is essential for enabling data discovery, ensuring governance, and supporting the effective use of Lakehouse assets across OHIP. For Lakehouse implementations, the Contractor **MUST**:

- Maintain a centralized Data Catalog⁷² that provides visibility into all data assets, their definitions, ownership, and relationships. This Data Catalog **MUST** be approved by OHIP.
- Support registration of all tables, views, and data products across the Raw Layer, Processed Layer, and Curated Layer tiers.
- Enable search and discovery capabilities that allow users to find datasets based on keywords, data domains, ownership, or other relevant attributes.

Catalog Solution Standards

For Snowflake-based Lakehouse implementations, the Contractor **MUST**:

- Leverage the Snowflake Horizon Catalog for metadata management, data discovery, and governance enforcement within the Lakehouse environment.
 - Where business requirements exceed native Horizon Catalog capabilities (such as cross-platform lineage spanning non-Snowflake systems, advanced business glossary management, or enterprise-wide data marketplace functionality), the Contractor **MAY** propose third-party catalog solutions (e.g., Alation, Collibra, Select Star), which is subject to OHIP approval. Third-party catalog implementations **MUST** integrate with Snowflake metadata and **MUST NOT** create duplicate or conflicting metadata repositories.
- Ensure the Data Catalog can integrate with the Lakehouse platform to enable automated metadata harvesting. Manual catalog maintenance is not acceptable for production tables; metadata **MUST** be synchronized automatically when tables are created, modified, or retired.
- Capture technical metadata including schema definitions, partitioning schemes, storage locations, and row counts, as well as operational metadata including last refresh time, data freshness, and quality scores.

6.10.2 Mandatory Catalog Attributes

Each dataset registered in the catalog **MUST** include the attributes in Table 40. The Contractor **MUST** ensure these attributes are populated and maintained accurately throughout the data lifecycle.

Table 38 – Mandatory Data Catalog Attributes

Attribute	Description
Dataset Name	Unique identifier following OHIP naming conventions as defined in Section 3.0 .
Description	Business-friendly description of the dataset purpose and contents.
Data Owner	Named individual or role accountable for the dataset.
Data Steward	Named individual or role responsible for data quality and definitions.

⁷² Refer to [Section 6.0](#) for more information.

Data Classification	Sensitivity classification Moderate (IAL2, AAL2). ⁷³
Data Sensitivity	Data type classification for the dataset. Determines baseline platform security controls. ⁷⁴
Compliance Indicator	One (1) or more regulatory or program compliance flags applicable to the dataset, separate from data type. Each compliance indicator carries independent governance, auditing, and access control requirements. ⁷⁵ Enter NONE if no specific compliance program applies. Multiple values MAY apply; use pipe delimiter.
Source System	Originating system identifier from which data flows. For datasets derived from multiple sources, indicate the primary source system or 'Multiple' with reference to lineage documentation for full source details.
Refresh Frequency	How often the dataset is updated (real-time, daily, weekly, monthly).
Retention Period	How long data is retained per Section 7.12 requirements.
Quality Score	Current data quality metrics from quality gate evaluations.
Related Datasets	Links to upstream sources and downstream consumers.
Domain / Subject Area	Business domain classification for the dataset (e.g., Claims, Eligibility, Provider, Encounters, Finance). Enables filtering and discovery by functional area.
Last Refresh Timestamp	Date and time when the dataset was last updated. Enables users to assess data currency and identify stale datasets.

6.10.3 Enterprise Glossary

Enterprise Glossary requirements, including ownership, governance, catalog integration, and definition standards, are consolidated in [Section 2.12.2](#). The Lakehouse Data Catalog implementation **MUST** comply with all glossary requirements defined therein.

⁷³ For more information, refer to the OHIP Security, Privacy, and Compliance Domain Document, Security and Privacy Compliance Standard.

⁷⁴ Valid values for Data Sensitivity includes: PHI (Protected Health Information), PII (Personally Identifiable Information), MCD (Medicaid Confidential Data), INTERNAL (non-public operational data with no special regulatory designation), PUBLIC (approved for unrestricted public release). Multiple values may apply; list all that apply separated by a pipe delimiter (e.g., PHI|PII).

⁷⁵ Valid values for Compliance Indicators include: FTI (IRS Federal Tax Information—IRS Pub 1075), SSA (Social Security Administration—Social Security Act §1106), ACA (Affordable Care Act Exchange data—ACA §1561/HITECH), 42-CFR-PT2 (Substance Use Disorder—42 CFR Part 2), HIVSID (HIV-related—NYS Public Health Law §2782), MH (Mental Hygiene—NYS Mental Hygiene Law §33.13), HIPAA (HIPAA Privacy/Security Rules).

6.10.4 Reference Data Management

Enterprise master and reference data management requirements are established in [Section 2.10](#). The requirements below govern the implementation of reference data management within the Lakehouse environment and supplement the enterprise framework.

For reference data management, the Contractor **MUST**:

- Govern reference data, which includes code tables, lookup values, and cross-reference mappings, through the data catalog.
- Maintain a master list of all reference data assets used within the Lakehouse environment.
- Designate an owner, document update procedures, and maintain version history for each data asset.
- Follow a controlled process that includes impact analysis to identify dependent datasets and reports, approval from the reference data owner, notification to downstream consumers before changes take effect, and version retention to support historical queries, for changes to reference data.
- Obtain OHIP approval before modification of reference data values used for Master Data Management such as valid CIN formats and NPI validation rules.

6.11 Schema Evolution and Change Control

Enterprise change control processes are established in the OHIP Infrastructure and Operations Domain Document, ITSM Standard. This section defines change control requirements specific to Lakehouse schema evolution and establishes requirements for managing schema changes in a controlled manner that preserves backward compatibility, minimizes disruption to consumers, and maintains data integrity throughout the Lakehouse environment.

The general database change control framework, including DDL approval, automated execution, and release coordination, is established in [Section 4.5.3](#). The requirements below supplement that framework with Lakehouse-specific schema evolution capabilities, including table format evolution and consumer notification timelines. Where both sections apply, the Contractor **MUST** satisfy both.

6.11.1 Schema Change Classification

Schema changes **MUST** be classified according to their potential impact on downstream consumers. The Contractor **MUST** evaluate each proposed change and apply the appropriate classification before proceeding with implementation.

Table 39 – Schema Change Classifications

Classification	Definition	Examples
Non-Breaking	Changes that do not affect existing queries or downstream processes	Adding new optional columns; Adding new tables; Expanding column width
Potentially Breaking	Changes that MAY affect some consumers depending on their implementation	Renaming columns; Changing data types with implicit conversion; Adding NOT NULL constraints
Breaking	Changes that will cause failures or incorrect results for existing consumers	Dropping columns or tables; Changing primary keys; Removing valid code values

6.11.2 Schema Change Process

The Contractor **MUST**:

- Follow a documented change control process that ensures proper review, approval, and communication for all schema changes.
 - Non-breaking changes **MAY** proceed with Data Architect approval and documentation in the data catalog.
- Require impact analysis identifying all affected consumers, notification to consumer teams with at least fourteen (14) days advance notice, Data Architect and Data Steward approval, and a documented rollback plan, for all potentially breaking changes.⁷⁶
- Conduct an impact analysis with explicit acknowledgment from all affected consumer teams, provide a minimum of thirty (30) days advance notice with migration guidance, obtain approval from the Data Owner and OHIP Data Architecture leadership, and develop a documented migration plan with testing requirements and a rollback plan with defined trigger criteria, for all breaking changes.
- Obtain OHIP approval and post-implementation review for emergency schema changes that bypass normal timelines.

6.11.3 Backward Compatibility Requirements

The Contractor **SHOULD** design schemas to support backward compatibility whenever feasible. The Contractor **SHOULD** also:

- Add new columns as nullable or with default values to avoid breaking existing insert statements.
- Retain deprecated columns for a transition period before removal, with the deprecation timeline communicated to consumers.
- Develop versioned schemas or views that allow consumers to migrate at their own pace within a defined window when breaking changes are unavoidable.
- Leverage the schema evolution capabilities provided by Apache Iceberg or Delta Lake table formats for the Processed and Curated Layers.
- Document which schema evolution operations are supported by the implemented table format and any limitations that affect change planning.

6.12 Data Retention and Lifecycle Management

This section establishes retention requirements for business data stored in the Lakehouse (Raw, Processed, and Curated Layer)⁷⁷ and for defining, implementing, and enforcing retention policies across the Data Lakehouse environment. All retention periods **MUST** comply with applicable federal and state regulations governing Medicaid data.

⁷⁶ For more information, refer to [Section 4.0](#).

⁷⁷ This section governs retention of Medicaid business data. For retention requirements governing pipeline logs, system logs, audit logs, and operational logs, refer to the OHIP Infrastructure and Operations Domain Document, Log Management Standard.

This section implements the enterprise data retention, purge, and archival policies established in [Section 2.8.1](#) within the Lakehouse architecture. All retention periods defined in [Section 2.8.1](#) apply. The requirements below govern how those retention periods are enforced at each architectural layer.

6.12.1 Retention Requirements by Data Classification

Enterprise data retention periods by data classification are defined in [Section 2.8.1](#). The Contractor **MUST** implement these retention periods within the Lakehouse environment using the layer-specific and storage tier requirements defined in the subsections below. For datasets containing multiple classifications, the longest applicable retention period **MUST** be applied. The Contractor **MUST** document the retention period for each dataset in the data catalog and implement automated enforcement mechanisms.

6.12.2 Retention by Architectural Layer

For the Raw Layer, the Contractor **MUST**:

- Retain data for the full retention period applicable to its classification.
 - Data can only be deleted when a legal hold has been lifted or there is explicit OHIP authorization to do so.
- **NOT** delete data before the retention period expires except under legal hold release or explicit OHIP authorization.
- Ensure data remains accessible within the timeframes required for audit and reprocessing, even if leveraging storage cost optimization through tiered storage, such as Amazon Web Services (AWS) Simple Storage Service (S3) Glacier (hereafter, S3 Glacier).

For the Processed Layer:

- The Contractor **MUST** retain data for the required operational period plus a minimum of ninety (90) days of version history. Where longer retention periods are required by data classification, those periods govern.

The Curated Layer follows similar retention requirements, with consideration for the specific reporting and analytical instances the data supports.

6.12.3 Storage Tier Transitions

The Contractor **MUST** implement automated storage lifecycle policies that transition data to cost-appropriate storage tiers as it ages.

The following transition guidelines apply to AWS S3 (hereafter, S3) storage implementations. Data accessed frequently for operational queries **MUST** remain in S3 Standard or S3 Intelligent-Tiering. Data older than ninety days (90) that is accessed infrequently **MAY** transition to S3 Standard-IA or S3 Glacier Instant Retrieval. Data older than one year (1) that is retained primarily for compliance **MAY** transition to S3 Glacier Flexible Retrieval or S3 Glacier Deep Archive.

The Contractor **SHOULD**:

- Ensure that data remains accessible within timeframes required for an audit response.
 - PHI, FTI, and MCD data **SHOULD** be retrievable timeframes agreed with OHIP based on data criticality and access frequency.

The Contractor **MUST**:

- Storage transition policies **MUST** be documented and approved by OHIP before implementation.
 - o Implementation uses S3 Lifecycle policies or equivalent platform capabilities.

6.12.4 Data Purge Procedures

When data reaches the end of its retention period, the Contractor **MUST** execute a controlled purge process. This includes verifying that no legal hold applies to the data (by confirming the retention period has been satisfied for all applicable classifications). The Contractor **MUST** also:

- Document the datasets to be purged, which also includes record counts and date ranges, upon receiving approval from the Data Owner. An audit record of the purge action **MUST** be created before any data is deleted.
- Retain data under legal hold regardless of the retention period expiration.
- Implement legal hold flags and automated controls that prevent deletion of held data in the Data Catalog.
- Suspend normal retention schedules indefinitely until legal holds are formally released by OHIP's Legal Counsel.

6.12.5 Archival and Historical Access

The Contractor **MUST**:

- Maintain the capability to restore archived data to an accessible state within defined timeframes.
- Fulfill restoration requests for data within cold storage within twenty-four (24) hours for standard requests and four hours (4) for urgent requests.
- Document the process for requesting data restoration and the associated costs.

6.13 Business Continuity and Disaster Recovery⁷⁸

Enterprise business continuity and disaster recovery requirements are established in the OHIP Infrastructure and Operations Domain Document, ITSM Standard. This section defines disaster recovery requirements and business continuity measures specific to Lakehouse data platforms.

6.13.1 Recovery Objectives

The Contractor **MUST** design and implement disaster recovery capabilities that meet the following recovery objectives. The business owner is responsible for performing a Business Impact Analysis (BIA) to determine the business criticality for various components and services. These objectives apply to the Lakehouse infrastructure and all data stored within it.

Table 40 – Recovery Objectives

Tier	Objective	Target	Description
------	-----------	--------	-------------

⁷⁸ For more information on incident management, escalation procedures, and communication requirements, refer to the OHIP Infrastructure and Operations Domain Document, ITSM Standard.

Tier 1: Critical	Recovery Point Objective (RPO)	< 1 hour	For highly critical processes (e.g., Revenue, Regulatory) where data loss MUST be minimal. A near-zero RPO MUST be the goal.
	Recovery Time Objective (RTO)	< 8 hours	Systems supporting highly critical processes MUST be restored rapidly to avoid significant business impact.
Tier 2: Standard	RPO	4 hours	For medium criticality processes (e.g., Operational Analytics) that can tolerate a moderate amount of data loss.
	RTO	24 hours	Systems supporting standard business operations MUST be restored within one business day.
Tier 3: Low	RPO	24 hours	For low-priority processes (e.g., Informational, Historical, Development) where data is less volatile.
	RTO	48 hours	Non-essential systems where extended downtime has minimal business impact.

The Contractor **MUST** document which datasets are classified into each tier, based on the business owner's BIA. This classification **MUST** be approved by OHIP and documented in the data catalog.

6.13.2 Backup Requirements

The Contractor **MUST** implement comprehensive backup procedures for all three architectural layers of the Lakehouse that protect against data loss from system failures, human error, malicious activity, and natural disasters. Additionally, the Contractor **MUST**:

- Use continuous protection through S3 versioning and cross-region replication of the Raw Layer.
- Enable S3 versioning on all raw data buckets to protect against accidental deletion or overwrite.
- Copy raw data to a secondary AWS region to protect against regional outages during cross-region replication.
- Not exceed the replication lag RPO target for the data classification.
- Implement database-level backup procedures that include periodic full exports of critical tables to durable storage, continuous replication of table metadata and data files to a secondary region, and documented recovery procedures for restoring tables from backups.
- Implement a tiered backup retention strategy that aligns with data criticality classifications and regulatory retention requirements for all Lakehouse databases.
- **Tier 1: Point-in-Time Recovery (PITR):** The Contractor **MUST** configure AWS Backup for Amazon RDS, or the platform-equivalent backup service for Snowflake-managed databases, to enable continuous point-in-time recovery with a minimum 30-day retention window. This tier supports rapid recovery from accidental deletion, human error, or system failure within the operational window and **MUST** be applied to all Processed and Curated Layer resources.
- **Tier 2: Long-Term Regulatory Compliance Archive:** The Contractor **MUST** generate and store monthly full database snapshots in AWS S3 Glacier Deep Archive, or equivalent cost-optimized cold storage, for the full retention period applicable to each dataset's classification as defined in [Section 7.12](#). This tier replaces the prior approach of retaining data in AWS S3 warm storage for ten years, which is not cost-effective for compliance-only retention. The monthly snapshot schedule **MUST** be automated and monitored for successful completion.

The backup tier assigned to each dataset **MUST** be documented in the data catalog, mapped to the dataset's criticality classification from the BIA, and approved by OHIP during the implementation planning phase.

6.13.3 Cross-Region Replication

The Contractor **MUST** implement cross-region replication to protect against regional AWS outages or disasters. The primary region for OHIP Lakehouse operations is US-East-1 (N. Virginia). The designated secondary region is US-East-2 (Ohio). The Contractor **MUST** also ensure:

- All Raw Layer objects are replicated to the secondary region within the applicable RPO window.
- The Processed Layer and Curated Layer critical tables are replicated to the secondary region.
- The Data Catalog metadata is replicated or accessible from the secondary region, and encryption keys and access credentials are available in the secondary region.

6.13.4 Disaster Recovery Procedures

The Contractor **MUST** develop and maintain:

- Documented disaster recovery procedures that enable restoration of Lakehouse operations in the event of a disaster.
- Procedures that address declaration criteria and authority for declaring a disaster, communication procedures for notifying OHIP and stakeholders, failover procedures for activating secondary region resources, data validation procedures to verify integrity after recovery, and fallback procedures for returning to the primary region.

The Contractor **MUST** review and update disaster recovery procedures annually and whenever significant infrastructure changes occur.⁷⁹

6.13.5 Disaster Recovery Testing Requirements

The Contractor **MUST** conduct regular disaster recovery testing to validate that recovery procedures work as designed and meet recovery objectives. The Contractor **MUST** conduct the following types of tests for Disaster Recovery:

- An annual full failover test that activates secondary region infrastructure and validates end-to-end data access
- A quarterly recovery drill that tests specific recovery procedures without full failover, and,
- A tabletop exercise following any significant infrastructure changes.

The test results **MUST** be documented—this includes the actual recovery times achieved, any issues encountered, and remediation actions taken. The Contractor **MUST** provide OHIP with the test reports within thirty (30) days of test completion.

⁷⁹ For more information, refer to the OHIP Infrastructure and Operations Domain Document, ITSM Standard.

6.14 Data Quality Testing Standards

General testing methodology, test environment management, and test documentation requirements are established in the OHIP Solution Delivery Domain Document, Test Management Standard. This section defines testing requirements specific to Lakehouse data pipelines and quality gates.⁸⁰

6.14.1 Pipeline Testing Requirements

The Contractor **MUST** implement comprehensive testing for all data ingestion and transformation pipelines before production deployment. The Contractor **MUST** also:

- Conduct unit testing of individual transformation functions and components, validating that transformation logic produces expected outputs for known inputs, edge cases are handled appropriately, and error conditions are detected and handled correctly.
- Ensure that testing logs are retained for the periods specified in [Section 7.12](#) and are accessible to designated staff upon request.
- Verify testing logs include sufficient detail to enable independent verification of test execution, including test case identifiers, execution timestamps, pass/fail outcomes, and any error messages or exceptions encountered.
- Use integration testing to validate end-to-end pipeline execution from source to target and verify that data flows correctly through all pipeline stages, quality gates are triggered and function as designed, metadata is correctly captured and stored, and pipeline orchestration and error handling work correctly.
- Ensure all pipeline code is accompanied by automated tests that are executed as part of the deployment process. Pipelines **MUST NOT** be deployed to production without passing all automated tests.

OHIP reserves the right to access and audit all Lakehouse testing logs, pipeline test results, and quality validation outcomes at any time.

6.14.2 Data Validation Testing

The Contractor **MUST** implement data validation testing that verifies the accuracy and completeness of data produced by pipelines. The Contractor **MUST** also:

- Conduct reconciliation testing that compares record counts and key metrics between source and target systems.
- Investigate and resolve material discrepancies before production deployment.
- Use regression testing to verify that changes to pipeline code do not introduce unintended changes to output data.
- Include representative samples of historical data that can be reprocessed and compared against baseline outputs in regression test suites.

⁸⁰ For comprehensive testing methodology including UAT procedures, test case management, and defect tracking, refer to the OHIP Solution Delivery Domain Document, Test Management Standard.

- Verify, using semantic testing, that data values are meaningful and consistent with business expectations (e.g., dates fall within reasonable ranges, enumerated values are valid, calculated fields produce expected results).

Anomaly Testing

For batch data loads, the Contractor **MUST** implement anomaly testing that compares current batch metrics against historical baselines to detect unexpected variations. Anomaly testing **MUST** include:

- **Volume delta analysis:** Compare record counts against prior batches and historical averages. Significant deviations (e.g., greater than 20% variance from rolling average) **MUST** trigger alerts for investigation.
- **Distribution analysis:** Monitor statistical distributions of key fields (e.g., claim amounts, date ranges, category frequencies) and flag batches with unusual patterns.
- **Missing data detection:** Identify expected records or time periods that are absent from the current batch compared to source system expectations.
- **Late-arriving data handling:** Track and report records that arrive outside expected processing windows.

Anomaly thresholds **MUST** be defined in collaboration with the Data Steward for each data domain and documented in the data catalog. Anomaly alerts do not automatically quarantine data but **MUST** be investigated and dispositioned before data is promoted to the Curated Layer.

6.14.3 Performance Testing

While conducting performance testing, the Contractor **MUST**:

- Validate that pipelines and queries meet throughput and latency requirements.
- Establish baseline metrics for pipeline execution time, identify bottlenecks before they impact production operations, validate that auto-scaling configurations respond appropriately to load, and verify that SLA requirements for data freshness can be met.
- Simulate expected production volumes and, where applicable, peak load scenarios.
- Document and monitor performance baselines for regression over time.

6.14.4 Quality Gate Testing

The Contractor **MUST**:

- Validate the quality gates defined in [Section 7.6](#) through testing.
- Verify that each quality gate validation check correctly identifies conforming and non-conforming records.
- Include test cases for known-good records that **MUST** pass each gate, known-bad records that **MUST** fail specific validation checks, boundary cases that test threshold logic, and records with multiple quality issues to verify proper handling.
- Validate quality gate threshold values against historical data to confirm they are appropriately calibrated. Thresholds that are too tight **MAY** quarantine acceptable data; thresholds that are too loose **MAY** allow quality issues to propagate.

6.14.5 Test Environment Requirements

General testing methodology requirements are defined in the OHIP Solution Delivery Domain Document, Test Management Standard. This section addresses Lakehouse-specific testing requirements.

The Contractor **MUST**:

- Maintain test environments that enable thorough validation before deploying in a production environment.
- Logically isolate test environments from production to prevent test activities from affecting production data or operations.
- Ensure test data is representative of production data in terms of volume, variety, and complexity, while complying with requirements for protecting PHI, PII, and other sensitive data in non-production environments.
- Provision each environment for Snowflake implementations as a separate Snowflake database to:
 - o Ensure complete isolation.
 - o Prevent test activities from affecting production data.
 - o Enable environment-specific access controls.
 - o Support independent scaling and cost tracking per environment.
 - o Align with Snowflake best practices for environment management.
- Not use live production data in non-production environments.
 - o Exceptions **MUST** be formally justified, documented, and approved by OHIP prior to use.
- Use synthetic data generation that preserves data characteristics without containing real sensitive values for test cases.
 - o Where synthetic data is not feasible, data masking or de-identification is the required alternative, applied in accordance with the OHIP Security and Privacy Compliance Standard.
- Produce and maintain a Data Protection Plan (DPP) for the Lakehouse implementation that documents how production data will be protected in lower environments, including masking strategies, synthetic data approaches, FTI handling procedures, and access controls for any production data used in testing. The DPP **MUST** be submitted to OHIP for approval prior to any testing activities involving sensitive data.⁸¹
- Ensure test environment configurations mirror production configurations closely enough to provide confidence that tests predict production behavior.

6.15 Platform-Specific Implementation Guidance

While the architectural requirements defined in [Section 7.4](#) through [7.14](#) are mainly platform-agnostic and mandatory for all implementations, this section provides guidance for the current OHIP implementation on

⁸¹ Refer to the OHIP Security, Privacy, and Compliance Domain Document, Security and Privacy Compliance Standard for more information.

Snowflake and AWS. The Contractor utilizing these platforms **MUST** follow this implementation guidance in addition to the core requirements. The Contractor proposing alternative platforms **MUST** demonstrate equivalent capabilities and obtain OHIP approval before implementation.

6.15.1 Storage Architecture

The Contractor **MUST**:

- Implement the Raw Layer using OHIP-managed S3 storage.
- Store data files in S3 buckets under OHIP control.
- Define Snowflake External Tables over S3 locations to enable query access without copying data into Snowflake's managed storage.
- Use either Snowflake-managed tables or External Tables (based on performance, cost, and access requirements) for the Processed and Curated Layers.
- Use Snowflake-managed tables with Iceberg format for the Curated Layer where query performance and Time Travel capabilities are priorities.
 - When Snowflake-managed tables are used, the Contractor **MUST** ensure that data remains exportable and accessible for disaster recovery and portability purposes.
- Configure S3 buckets with server-side encryption according to OHIP security requirements.
- Enable bucket versioning to support recovery from accidental deletions or overwrites.
- Configure lifecycle policies to transition older data to appropriate storage tiers while maintaining accessibility for reprocessing scenarios.

6.15.2 Table Formats

The Contractor **MUST**:

- Use Snowflake Tables with Iceberg format where time travel and versioning requirements apply in the Processed and Curated Layers.
 - The format provides ACID transaction capabilities while maintaining an open table format that external compute engines can read without Snowflake-specific dependencies.
 - This approach ensures that Apache Spark engines running on AWS EMR or other platforms can access OHIP data for advanced analytics workloads without proprietary lock-in.
- Implement table partitioning strategies appropriate to query patterns.
 - Date-based partitioning is typically appropriate for transaction data, enabling efficient time-range queries for reporting.
- Document and obtain Data Architect approval for partition schemes before implementation.

6.15.3 Compute Resource Management

The Contractor **MUST**:

- Implement workload isolation using Snowflake Multi-Cluster Warehouses.
- Configure separate virtual warehouses for distinct workload types to ensure predictable performance and enable cost attribution.

- Maintain dedicated warehouses for at least three types of workloads: Extract, Transform, and Load (ETL) ingestion, business intelligence queries, and data science and machine learning.
- Configure warehouse sizing and auto-scaling policies based on workload characteristics and approved by OHIP.
- Implement resource monitors to prevent runaway costs and to generate alerts when consumption approaches budget thresholds.

6.15.4 Security and Privacy Governance

For Lakehouse implementations, the Contractor **MUST**:

- Comply with all requirements defined in the OHIP Security, Privacy, and Compliance Domain Document, Security and Privacy Compliance Standard. Key requirements applicable to Lakehouse platforms include:
 - o Zero Trust Architecture
 - o Encryption of data at rest and in transit
 - o Identity and Access Management via the state-approved IdP
 - o Audit logging and monitoring⁸²
- Implement row-level security policies to restrict data access based on user roles and data classifications.
- Apply dynamic data masking to sensitive fields including PHI and PII elements, with masking policies configured to reveal data only to users with appropriate authorization.
- Integrate authentication with a state-approved Identity Provider (IdP) and support single sign-on.
- Align role-based access control with the roles defined in [Section 7.3](#).

The Contractor **SHOULD**

- Integrate with OHIP's enterprise audit logging.
- Leverage Snowflake Horizon capabilities for data governance.
- Ensure all access grants and revocations are logged and auditable.

6.15.4.1 Lakehouse-Specific Security Considerations

The following security considerations are specific to Lakehouse architectures:

- Row-level and column-level security **MUST** be implemented for PHI/PII data in the Curated Layer.
- Data masking capabilities **MUST** be available for non-production environments.
- Cross-region replication **MUST** maintain encryption standards.

⁸² For comprehensive security and privacy requirements including vulnerability scanning, penetration testing, and certification requirements, refer to the OHIP Security, Privacy, and Compliance Domain Document, Security and Privacy Compliance Standard.

6.15.5 Quarantine Zone Implementation

The Contractor **MUST**:

- Implement the Quarantine Zone as a Snowflake Secure Schema with access restricted to authorized data quality personnel.⁸³
- Use ETL Streams and Tasks to automatically trigger remediation workflows when records are inserted into quarantine tables.
- Apply query tags to remediation workloads to enable tracking of compute costs associated with data quality remediation, supporting Contractor accountability and chargeback processes.

6.15.6 Database DevOps and Change Management

The Contractor **MUST** implement the following Database DevOps practices that enable automated, auditable, and repeatable deployments across environments:

- **Infrastructure as Code (IaC)**: Snowflake configuration (warehouses, databases, schemas, roles) **MUST** be defined in version-controlled IaC templates (e.g., Terraform, Pulumi).
- **Continuous Integration (CI)**: All database changes (DDL, stored procedures, policies) **MUST** be validated through automated testing before merge approval.
- **Continuous Deployment (CD)**: Approved changes **MUST** deploy automatically to lower environments; production deployments require approval gates per OHIP change management procedures.
- **Version Control**: All database objects, transformations, and configurations **MUST** be stored in an OHIP-approved version control system with complete change history.
- **Migration Management**: Schema migrations **MUST** use versioned migration scripts that can be applied and rolled back consistently across environments.
- **Monitoring and Alerts**: Deployment pipelines **MUST** include automated monitoring and alerting for deployment failures and anomalies.

6.15.7 Snowflake Data Loading Standards

The Contractor **MUST** implement data loading practices appropriate to the volume, frequency, and latency requirements of each data source. Snowflake supports multiple loading mechanisms, each suited to different ingestion patterns.

Bulk Loading (Batch)

Bulk loading is appropriate for scheduled batch ingestion of large file-based datasets. The Contractor **MUST** configure bulk loading processes to ensure reliability, error handling, and auditability and adhere to the following standards:

- External stages on S3 **MUST** be used for bulk file ingestion
- COPY INTO commands **MUST** include ON_ERROR handling (CONTINUE, SKIP_FILE, or ABORT_STATEMENT based on data criticality)

⁸³ For more information, refer to the OHIP Security, Privacy, and Compliance Domain Document, Security and Privacy Compliance Standard.

- File formats **MUST** be explicitly defined and version-controlled
- Loading history **MUST** be retained for audit and troubleshooting

Continuous Loading (Snowpipe)

Continuous loading enables near real-time data ingestion by automatically loading files as they arrive in cloud storage.

The Contractor **SHOULD**:

- Snowpipe **SHOULD** be used for event-driven ingestion patterns where low latency is required and for continuous ingestion from S3.

The Contractor **MUST** adhere to the following standards for continuous loading:

- Auto-ingest configuration with S3 event notifications is required for near real-time use cases.
- Snowpipe error handling **MUST** route failed files to a designated error location for remediation.
- Pipe status **MUST** be monitored via PIPE_USAGE_HISTORY.

Stage Management

Stages serve as intermediate storage locations that Snowflake uses to access source files during the loading process.

The Contractor **SHOULD** adhere to the following standards for stage management:

- Internal stages **MAY** be used for temporary data or small-volume loads.

The Contractor **MUST** adhere to the following standards for stage management:

- External stages (S3) are required for production data pipelines.
- Stage credentials **MUST** be managed via storage integrations, not embedded credentials.

6.15.8 Data Delivery and Consumption Standards

The Contractor **MUST** implement secure data delivery mechanisms that enforce access controls, support authorized consumption patterns, and maintain audit trails for all data access.

Row-Level Security for Data Filtering

The Contractor **MUST**:

- Implement Snowflake Row Access Policies to filter data based on the requesting user's Data Consuming Entity (DCE) membership and access tier.
- Replace the Virtual Private Database (VPD) functionality used in traditional Oracle-based systems with Row Access Policies.
- Apply Row Access Policies to all Curated Layer tables containing sensitive data and enforce the access scope documented in the DCE registration.

Approved Data Delivery Mechanisms

The following data delivery mechanisms are approved for Lakehouse data consumption:

Table 41 – Data Delivery Mechanisms

Mechanism	Use Case	Security Requirements
Direct Query (BI Tools)	Interactive analytics, dashboards, ad-hoc reporting	OAuth or SAML authentication; role-based warehouse access
File Export (COPY INTO)	Batch data delivery, regulatory submissions	Approved export locations only; encryption required; audit logging
Snowflake Data Sharing	Cross-account data access for authorized partners	Share definitions approved by Data Owner; no data movement
API Access (Snowpark/REST)	Programmatic integration, application embedding	API credentials managed via secrets manager; rate limiting enforced

Business Intelligence Integration

For Tableau and other business intelligence tool integrations, the Contractor **MUST**:

- Configure Snowflake OAuth or SAML authentication aligned with the state-approved Identity Provider (IdP).
- Use dedicated roles with least-privilege access for BI tool service accounts.
- Ensure query results respect row-level and column-level security policies without requiring additional filtering logic in the BI layer.

Data Delivery Audit Trail

The Contractor **MUST**:

- Ensure all data delivery activities are logged and auditable.
- Enable Snowflake Access History to capture query execution, data access patterns, and export activities.
- Retain audit logs for the periods specified in [Section 7.12](#) and provide them upon request.

6.15.9 Snowflake Naming Conventions

Snowflake object naming **MUST** comply with [Section 3.0](#). This section provides Snowflake-specific guidance to ensure consistency across Lakehouse implementations.

General Naming Principles

All Snowflake object names **MUST**:

- Use UPPER_SNAKE_CASE format for consistency and to avoid case-sensitivity issues.
- Be meaningful and business-aligned to support data discovery.
- Avoid abbreviations unless they are industry-standard (e.g., PHI, CIN, NPI) or defined in the enterprise glossary.

Database Naming

Databases **MUST** include an environment prefix to distinguish between environments:

Table 42 – Database Naming

Environment	Prefix	Example
Development	DEV_	DEV_MEDICAID_LAKEHOUSE
Test	TST_	TST_MEDICAID_LAKEHOUSE
Production	PRD_	PRD_MEDICAID_LAKEHOUSE

Schema Naming

Schemas **MUST** indicate the architectural layer and, where applicable, the data domain:

Table 43 – Schema Naming

Layer	Pattern	Example
Raw	RAW_[DOMAIN]	RAW_CLAIMS, RAW_ELIGIBILITY
Processed	PROC_[DOMAIN]	PROC_CLAIMS, PROC_PROVIDER
Curated	CUR_[DOMAIN]	CUR_ANALYTICS, CUR_REPORTING
Quarantine	QUARANTINE	QUARANTINE

Table and View Naming

Tables and views **MUST** follow the pattern: [ENTITY]_[QUALIFIER] where ENTITY describes the business object and QUALIFIER provides additional context (e.g., HIST for historical, DIM for dimension, FACT for fact tables).

Examples:

- MEMBER_ELIGIBILITY
- CLAIM_HEADER
- PROVIDER_DIM
- ENCOUNTER_FACT
- MEMBER_ELIGIBILITY_HIST

Warehouse Naming

Virtual warehouses **MUST** indicate their workload type:

Table 44 – Warehouse Naming

Workload	Pattern	Example
ETL/Ingestion	WH_ETL_[SIZE]	WH_ETL_MEDIUM
Business Intelligence	WH_BI_[SIZE]	WH_BI_LARGE
Data Science	WH_BI_LARGE	DS_XLARGE
Ad-Hoc Queries	DS_XLARGE	WH_ADHOC_SMALL

Other Objects

Table 45 – Other Object Naming

Object Type	Pattern	Example
Stage	STG_[SOURCE]_[TYPE]	STG_EMEDNY_EXTERNAL
Pipe	PIPE_[SOURCE]_[TARGET]	PIPE_S3_RAW_CLAIMS
Stream	STRM_[TABLE]	STRM_CLAIM_HEADER
Task	TASK_[FUNCTION]	TASK_DAILY_REFRESH
File Format	FF_[FORMAT]	FF_CSV_STANDARD, FF_PARQUET

For comprehensive naming requirements beyond Snowflake-specific objects, refer to [Section 3.0](#).

6.16 Performance Monitoring and Key Performance Indicators (KPIs)

Consistent with the OHIP ITSM Standard requirements for measurement and reporting, the Contractor **MUST** monitor Data Lakehouse operations and report on KPIs.

Table 46 – Performance Monitoring KPIs

KPI	Target	Description
Pipeline Success Rate	>99.9%	Percentage of scheduled ingestion jobs that complete successfully without manual intervention
Quality Fail Rate	<0.5%	Percentage of records routed to the Quarantine Zone due to quality gate failures
Data Freshness	Per SLA	Time elapsed between source system extraction and availability in the Curated Layer
Traceability Score	100%	Percentage of Curated Layer records that can be traced back to Raw Layer source files
Quarantine Resolution Time	<48 hours	Mean time to remediate quarantined records and restore them to normal processing
Cost Efficiency	<\$X/TB	Total compute and storage costs divided by total data volume managed
Disaster Recovery Readiness	100%	Percentage of critical datasets meeting replication and backup requirements
Catalog Completeness	100%	Percentage of production datasets registered in catalog with required attributes

The Contractor **MUST**:

- Report Performance Monitoring KPIs to OHIP monthly.
- Provide RCA and remediation plans for any significant deviations from targets.
- Review KPI trends quarterly to identify opportunities for process improvement.
- Submit final targets for review and approval by OHIP based on operational requirements and baseline performance data.

6.17 Compliance and Enforcement

6.17.1 Applicability

A ninety-day (90) transition period provides existing projects time to come into compliance with new requirements. A Contractor with existing implementations that do not conform to this standard **MUST** submit a Compliance Roadmap within thirty days (30) of standard publication.

6.17.2 Compliance Verification

OHIP will conduct periodic compliance reviews to verify adherence to this standard. Reviews **MAY** include examination of architecture documentation, inspection of implemented quality gates, audit of data lineage capabilities, verification of disaster recovery readiness, and assessment of KPI performance against targets. The Contractor **MUST** provide access to systems, documentation, and personnel as needed to support compliance verification activities.

7.0 Content and Document Management Standard

7.1 Introduction

This Content and Document Management Standard establishes mandatory requirements for the lifecycle management of unstructured and semi-structured content created, received, stored, retrieved, or disposed of by OHIP and its Contractors in support of the MES. Examples include scanned correspondence, member and provider attachments, contracts and Memoranda of Understanding (MOUs), audit responses, internal policy documents, and email or messaging records subject to records-retention obligations. Deliverables and artifacts associated with this Standard can be found in [Appendix A](#).

This Standard complements, and does not replace, the structured-data requirements defined elsewhere in this document. Where the same content has both a structured representation (e.g., a claim record) and an unstructured representation (e.g., a scanned attachment to that claim), both representations **MUST** be governed: the structured record under [Sections 2.0](#) through [7.0](#), and the unstructured content under this section.

 **Note:** Records management retention schedules and legal-hold obligations established by the New York State Archives and the New York State Department of Health Records Officer take precedence where they are stricter than the requirements of this Standard.

7.2 Objectives

The objectives of this Standard are to:

- Ensure unstructured and semi-structured content is captured, classified, retained, and disposed of in compliance with all applicable federal, State, and OHIP records-retention obligations.
- Establish authoritative repositories for OHIP business content so that records are discoverable, defensible, and resilient to loss.
- Apply the same data classification taxonomy (PHI, FTI, MCD, PII, INTERNAL, PUBLIC) to content that is applied to structured data and integrate content metadata with the Enterprise Data Catalog.
- Enable timely, complete, and defensible response to audit requests and CMS inquiries.
- Provide enforceable requirements that establish clear expectations for Contractors implementing or operating Enterprise Content Management (ECM) capabilities on behalf of OHIP.

7.3 Scope and Applicability

This Standard applies to all unstructured and semi-structured content created, received, or stored by Contractors in the course of MES delivery and operations, including but not limited to: scanned and electronic documents, images, faxes, audio and video recordings, transcripts, email and messaging records, contracts and amendments, project artifacts, and reports. Source code repositories are governed by the OHIP Solution Delivery Domain Document, Code Standard, and not by this Standard.

Applicability defines the content this Standard governs; it does not, by itself, designate a storage location. Where a given item resides is determined by its data classification and records status under [Section 7.4](#):

- **Operational and record content**—content received from external parties, content that constitutes an official record subject to a retention schedule, and any content classified as PHI,

PHI, MCD, or FTI—is maintained in an OHIP-approved Enterprise Content Management (ECM) system, which serves as the SOR (see Sections [7.4.1](#) and [7.4.3](#)).

- **Project working documents and interim deliverables** that Contractors produce in the course of project execution such as presentations, diagrams, mapping documents, and drafts need not reside in the ECM and may be maintained in the Department's approved, managed project workspace. Such content remains subject to the classification, retention, and metadata requirements of this Standard, and any item that later becomes an official record or that contains PHI, PII, MCD, or FTI is captured into the OHIP-approved ECM in accordance with [Section 7.4](#).

This Standard does not designate specific products. The approved ECM system and related tooling are identified in the OHIP Infrastructure and Operations Domain Document, Preferred Operational Tools Standard (see [Section 7.8](#)).

Relationship to Other OHIP Standards

This Standard operates in coordination with the data classification taxonomy ([Section 2.0](#)), data retention and purge requirements ([Section 2.6.1](#)), and metadata management ([Section 2.10](#)), and integrates content metadata with the Enterprise Data Catalog defined under the Data Lakehouse Architecture Standard ([Section 6.0](#)). Security and privacy requirements for content are governed by the OHIP Security, Privacy, and Compliance Domain Document. Source code repositories are governed by the OHIP Solution Delivery Domain Document, Code Standard, and are excluded from this Standard. For the complete list of companion standards, see [Section 1.6](#).

7.4 Content Lifecycle Management

The Contractor **MUST** implement and document a complete content lifecycle covering capture, classification, storage, access, retention, and disposition for all in-scope content. The lifecycle **MUST** be described in a Content Management Plan submitted for OHIP review and approval prior to ECM implementation activities.

7.4.1 Capture and Ingestion

- All content received from external parties **MUST** be captured into an OHIP-approved ECM system. The Contractor **MUST NOT** store content classified as PHI, PII, MCD, FTI, or INTERNAL in personal mailboxes, local workstation drives, or unmanaged shared drives.
- Scanned and imaged content **MUST** be captured at a resolution and color depth sufficient to preserve evidentiary quality (minimum 300 DPI for textual documents and 24-bit color for forms with graphical or signature elements).
- Optical Character Recognition (OCR) **MUST** be applied to all scanned textual content to enable full-text indexing and e-discovery searchability. OCR confidence scores **MUST** be retained as content metadata and content with confidence below an OHIP-approved threshold **MUST** be flagged for human review.
- Each captured item **MUST** receive a unique, immutable content identifier and **MUST** be associated with a capture event record (date, time, originating system or user, and channel).

7.4.2 Classification

- Every content item **MUST** be classified according to the OHIP data classification taxonomy (PHI, FTI, MCD, PII, INTERNAL, PUBLIC) defined in [Section 2.0](#) and the OHIP Security, Privacy, and Compliance Domain Document, Security and Privacy Compliance Standard.

- Classification **MUST** be applied at the time of capture, **SHOULD** be re-evaluated upon significant change, and **MUST** be reviewed at least annually or upon any reclassification trigger. When content sensitivity is ambiguous, content **SHOULD** be classified conservatively (most-restrictive) pending formal review, and the ambiguity **MUST** be resolved within [TBD: number of business days, to be confirmed with OHIP].
- Where content contains multiple sensitivity levels, the highest applicable classification **MUST** be assigned to the content item as a whole.

7.4.3 Storage and Repositories

- Content classified as PHI, PII, MCD, or FTI **MUST** be stored only in OHIP-approved ECM repositories that meet the encryption, access-control, and audit requirements of the OHIP Security, Privacy, and Compliance Domain Document, Security and Privacy Compliance Standard.
- Content **MUST** be stored with versioning enabled. All material modifications **MUST** create a new immutable version. Prior versions **SHOULD** be retained in accordance with the applicable records retention schedule, and versions subject to a legal hold **MUST** be retained for the duration of the hold.
- Content **SHOULD NOT** be duplicated across repositories without an approved, documented business justification. Where duplication is unavoidable, one repository **MUST** be designated the SOR for that content type.

7.4.4 Access and Use

- Access to content **MUST** be enforced through Role-Based Access Control (RBAC) integrated with the Department's enterprise Identity and Access Management (IAM) solution. Local repository accounts **MUST NOT** be used after initial ECM deployment.
- Content access events (read, download, print, export, share) for PHI, FTI, PII, and MCD content **MUST** be logged with sufficient detail to support audit and FOIL response, including user identity, content identifier, action performed, and timestamp; access events for INTERNAL and PUBLIC content **SHOULD** be logged to the same standard.
- Content **MUST NOT** be transmitted outside of OHIP-controlled environments except through OHIP-approved secure channels documented in the Content Management Plan.

7.4.5 Retention and Disposition

Content **MUST** be retained according to the retention schedule applicable to the content type, applying the longer of: the OHIP retention schedule established under [Section 2.8.1](#), the New York State Archives retention schedule for Department records, or any retention period imposed by litigation hold or regulatory mandate.

- Disposition (purge or archive) **MUST** be executed only through documented disposition workflows that produce an audit trail. Manual deletion by individual users **MUST NOT** be permitted for content classified at INTERNAL or higher.
- Content under active litigation hold, audit hold, or investigative hold **MUST NOT** be disposed of regardless of retention-schedule expiration. Hold status **MUST** be tracked as content metadata and **MUST** prevent automated disposition processes from acting on held content.

- Disposition events **MUST** be logged, and the disposition log **SHOULD** be retained in accordance with the applicable records retention schedule.

7.5 Metadata and Catalog Integration

The Contractor **MUST** capture and maintain a minimum metadata set for every content item and **MUST** integrate that metadata with the Enterprise Data Catalog defined in [Section 7.10](#). The minimum metadata set is defined in Table 49.

Table 47 – Mandatory Content Metadata

Metadata Element	Description	Required
Content Identifier	Unique, immutable system identifier assigned at capture.	MUST
Content Title	Human-readable title or document name.	MUST
Content Type	Classification of content (e.g., correspondence, contract, attachment, image, audio, video, email).	MUST
Origin / Source	Originating system, agency, individual, or channel.	MUST
Capture Event	Date, time, user or system, and capture method.	MUST
Classification	PHI, FTI, MCD, PII, INTERNAL, or PUBLIC (highest applicable).	MUST
Compliance Indicators	Applicable regulatory categories (HIPAA, IRS Pub 1075, 42 CFR Part 2, etc.).	MUST
Owner / Custodian	Designated business owner and records custodian.	MUST
Retention Class	Retention schedule code and minimum retention period.	MUST
Hold Status	Active legal, audit, or investigative holds, if any.	MUST
Linked Records	Identifiers of related structured records (claim, member, provider, case, etc.).	SHOULD
OCR Confidence	Average and minimum OCR confidence for scanned textual content.	SHOULD
Version	Version number and prior-version reference.	MUST
Disposition Date	Scheduled disposition date based on retention class and hold status.	MUST

7.6 E-Discovery Search and Production

- ECM repositories **MUST** support full-text search across the OCR'd content and metadata. Search results **MUST** be exportable in standard e-discovery production formats (e.g., load files compliant with EDRM XML, Concordance DAT, or Relativity-compatible exports).
- Redaction tooling **MUST** be available to mask PHI, PII, FTI, MCD, and other protected information prior to production. Redactions **SHOULD** be applied as a separate layer that

preserves the underlying content for in-house review, and redacted content **MUST NOT** be recoverable in production exports.

7.7 Content Quality and Information Governance

- The Contractor **MUST** define and apply quality controls for captured content, including duplicate detection, completeness checks (presence of required pages or signatures), and OCR-confidence thresholds. Quality issues **MUST** be tracked and remediated using the same processes defined in [Section 2.15](#).
- The Contractor **MUST** monitor and report on content proliferation indicators (e.g., growth rate per repository, duplicate rate, orphaned content rate) and **MUST** propose remediation when thresholds approved by OHIP are exceeded.
- The Contractor **MUST** obtain BEDG approval for all content classification changes that affect retention class, regulatory indicators, or sensitivity classification at the content-type level.

7.8 Tools and Standards

The Contractor **MUST** use OHIP-approved Enterprise Content Management (ECM) systems and **MUST** comply with the OHIP Infrastructure and Operations Domain Document, Preferred Operational Tools Standard for selection of ECM, e-discovery, and redaction tooling.

Standard markup and exchange formats (PDF/A for archival, ODF or DOCX for editable content, XML or JSON for structured exchange) **SHOULD** be preferred. Proprietary or unsupported formats **MUST** be migrated to standards-based formats during ingest where feasible.

7.9 Compliance and Enforcement

The Contractor **MUST** comply with this standard upon publication and with any amendments issued by OHIP. Where compliance is not feasible, the Contractor **MUST** formally document the exception with OHIP, develop a mitigation plan, and define a timeline for compliance per the process in [Section 1.8](#). Compliance is verified through the Compliance Tier Framework defined in [Section 1.7](#). Findings are addressed through the CAP process defined in [Section 1.8.5](#).

8.0 Contact Information

Submit all inquiries and requests for future enhancements to the policy owner at:

Division of Operations and Systems (DOS) and Chief Technology Office (CTO)
11 Corporate Woods Boulevard
Albany, NY 12211
Telephone: TBD
Email: TBD

Statewide technology policies, standards, and guidelines **MAY** be found at the following website:
<https://its.ny.gov/policies>

9.0 Revision History

The following table records the revision history of this Consolidated Domain Standard. The six component standards each maintained their own revision histories prior to consolidation. This table captures the lifecycle of the domain document from initial publication onward.

Date	Version	Author	Description of Change
06/25/26	1.0	NYS TAS	Initial publication of the Data and Information Management Domain Document, which consolidates the Data Governance Standard, Data Architecture and Naming Standard, Database Design Standard, Data Conversion Standard, and Data Lakehouse Architecture Standard, and Content and Document Management Standard and incorporates compliance tiers (MUST/SHOULD/MAY) for all requirements.
10/23/26	1.1	NYS TAS	Domain Document approved by DOH.

Appendices

Appendix A: Deliverables and Artifacts

The following tables identify the Deliverables and Artifacts associated with this Domain Document. Deliverables are contractually binding work products that the Contractor **must** submit to OHIP for formal review and approval. Artifacts are supporting work products (e.g. documentation, models, charts) that the Contractor **must** create, maintain, and retain, as determined by OHIP. The Contractor **must** provide Artifacts to OHIP upon request, or as otherwise directed, to support oversight, audits, and program needs.

The Contractor **must** produce and submit all applicable Deliverables and Artifacts identified in the format, structure, and cadence specified by OHIP or the Contract. OHIP may provide templates for the Contractor's use.

OHIP may modify the Deliverables and Artifacts required by this Appendix at any time to align with evolving program needs. Upon notice of changes, the Contractor **must** comply with all modifications and implement them promptly.

Table 48 – Data Governance Standard Deliverables and Artifacts

Type	Name	Description
Deliverable	Data Management Design Document	The Contractor must develop a Data Management Design Document, which is the detailed data management design for the solution, documenting data structures, flows, storage, retention, and governance decisions that implement the Data Management Plan for the solution.
Deliverable	Data Management Plan	<p>The Contractor must develop a Data Management Plan that provides a structured approach to handling data, which encompasses its collection, storage, analysis, sharing, and preservation. This plan must include, but is not limited to:</p> <ul style="list-style-type: none">• <i>Data Scope and Summary</i> – Provide the database to be used and clearly define the scope of data to be persisted (with types of data and anticipated volumes)• <i>Risk Assessment</i> – Perform a risk assessment covering both security and non-functional risks• <i>Standards, Guidelines, and Best Practices</i> – Provide a list of standards and guidelines to be used in the data management life cycle including database design. Database design must follow a Federal, State, and/or Industry best practice set of standards• <i>Tools</i> – A list of all the tools that the Contractor wishes to use must be populated• <i>Data Inventory, Catalog, Data Dictionary, and Glossary</i> – Provide details of the approach to implementation and use of a data catalog, data dictionary and glossary for data elements. The catalog must flag sensitive data (PHI/PII) and elements necessary for FHIR. The catalog must provide reference on associated Federal standards used to define the elements.

		<ul style="list-style-type: none"> • <i>Data Lineage (mapping)</i> – Provide details of the approach to implementation of technical and business lineage, including the approach to implementation of technical lineage (tracing the flow through systems) and business lineage (linking technical details to business processes) to provide a comprehensive view for both technical and non-technical stakeholders. • <i>Data Naming</i> – Provide the process to be used to maximize the use of Federal standards for data (entity and element) naming. • <i>Master data management (MDM) Keys</i> – Identify existing or establish master keys to link data across systems and provide them with their definition and any existing usage of the keys • <i>Data Quality Management</i> – A detailed strategy that outlines the processes, tools, and quality measures to ensure data integrity, accuracy, and completeness throughout the analytical life cycle This section must document the comprehensive measurement approach for all datasets and include: • <i>Data Quality Dimensions</i> – Documentation of quality dimensions with standardized calculation formulas <ul style="list-style-type: none"> o <i>Quality Thresholds by Criticality</i> – Specification of minimum quality thresholds by data criticality classification o <i>Assessment Frequency and Scheduling</i> – A defined schedule for data quality assessments based on criticality o <i>Validation Rules</i> – Documentation of all validation rules defining correct values for accuracy assessment (e.g., range checks, format validation) o <i>Monitoring and Correction Procedures</i> – Processes for routine data profiling, automated quality checks, timeliness monitoring, quality issue logging, and a formal process for identifying and remediating data quality issues, with preference for source corrections o <i>Quality KPIs and Scorecards</i> – Definition of Key Performance Indicators for data quality tracking with a regular reporting cadence o <i>Root Cause Analysis</i> – Process to conduct RCA for significant or recurring quality issues with documented outcomes • <i>Upgrades</i> – Schedule and downtime requirements for database upgrades maintaining major versions no older than N-1 • <i>Model Versioning</i> – Provide the approach to version the data model and all database changes (with Rollback) and the tools that support versions and Rollback
--	--	--

		<ul style="list-style-type: none"> • <i>Audit and Logging</i> – Provide the plan to audit all access, data, and database changes in an easily reportable manner • <i>Backup</i> – Provide the schedule and location for backups, how long backups will be retained, and how backups will be verified. Provide how long will take to restore from backups if the need arises • <i>Monitoring and Alerts</i> – Approach for the monitoring and alerts that will be implemented to assess for data anomalies • <i>Data Access Controls</i> – Provide the Identity Provider to be used for administrative and application access to the data. Provide the major security controls for access • <i>Assumptions, Constraints and Dependencies</i> – Provide a list of assumptions, constraints, and dependencies that may impact data management
--	--	---

Table 49 – Data Architecture and Naming Standard Deliverables and Artifacts

Type	Name	Description
Reference	Refer to the OHIP Solution Delivery Domain Document, Business and Technical Architecture Standard	The data architecture, naming, and modeling deliverables and artifacts previously listed here—including the Data Dictionary, Data Models, Functional Design Documents, Technical Design Documents, Roadmap—are produced and governed under the OHIP Solution Delivery Domain Document, Business and Technical Architecture Standard. Integration design is governed under the OHIP Infrastructure and Operations Domain Document, Application Integration Standard. This Standard's Section 3 requirements apply to those deliverables as they pertain to data architecture and naming.

Table 50 – Database Design Standard Deliverables and Artifacts

Type	Name	Description
Artifact	Data Fix Documentation	The Contractor must provide Data Fix Documentation that records all data fix events including business justification, records affected, testing results, business user sign-off, and associated problem/defect tickets.
Artifact	Database Change Request Log	The Contractor must provide a Database Change Request Log that records all database change requests including DDL changes, Data Architect approvals, testing results, and promotion status.
Artifact	Database Version History	The Contractor must provide a Database Version History that records all database versions (delta and named), including changes made, date, author, and release association.

Table 51 – Data Conversion Standard Deliverables and Artifacts

Type	Name	Description
Deliverable	Data Conversion and Migration Plan	<p>The Contractor must develop a Data Conversion and Migration Plan that outlines the strategy, steps, and considerations when transferring data from one system to another. The Data Conversion and Migration Plan must include, but not be limited to:</p> <ul style="list-style-type: none"> • <i>Synchronization Strategy</i> – This strategy outlines the approach to keeping legacy systems synchronized with new systems during the phased rollout • <i>Data Profiling Reports</i> – Statistical and distribution analyses of source data (patterns, outliers, nulls, referential issues) informing mapping and rule design • <i>Data Mapping Criteria</i> – Details the standards and rules for mapping source data to destination systems • <i>Data Lineage</i> – Involves documenting the movement, origins, and transformations of data through systems • <i>Conversion Environments</i> – Defines the infrastructure used for data conversion activities • <i>Conversion Schedule</i> – Specifies the timeline for each phase of the data conversion process, including milestones and deadlines • <i>Tools</i> – Lists the software and utilities used for data extraction, transformation, and loading (ETL), synchronization, and validation • <i>Stakeholder Identification and Engagement</i> – Identifies key stakeholders involved in the data conversion process and establishes engagement strategies • <i>Data Acceptance Criteria</i> – Defines the conditions required for accepting migrated data as valid and complete • <i>Testing and Validation Plan</i> – Outlines procedures to ensure data integrity, performance, and functionality post-conversion, including methods to compute and compare control totals, thresholds, exception handling, and approvals for reconciliation • <i>Backup and Recovery Plan</i> – Specifies strategies for backing up data, ensuring data preservation and recovery in case of faults. This must include secure, immutable snapshots of conversion data, scripts, and reports retained for audit, Rollback, and compliance • <i>Security Plan</i> – Details measures to protect data during conversion, handling sensitive information with encryption, access controls, and compliance with privacy regulations • <i>Roles and Responsibilities</i> – Defines roles involved in the conversion process, specifying responsibilities to ensure accountability • <i>Cutover Runbook</i> – Defines a step-by-step Cutover execution plan • <i>Security Plan</i> – Details conversion-specific security controls

		<ul style="list-style-type: none"> • <i>Risk Register</i> – Defines data conversion risks and their impact, likelihood, mitigation, owner, and status • <i>Assumptions, Constraints and Dependencies</i> – Lists assumed conditions, existing limitations, and interdependencies impacting the conversion plan
Deliverable	Data Mapping and Transformation Specification	<p>The Contractor must develop a Data Mapping and Transformation Specification that provides a timely and accurate report of data transformation. It must include, at a minimum, the following:</p> <ul style="list-style-type: none"> • <i>Data Mapping Specifications</i> – List of each data element in the source system and its corresponding target element; definitions and metadata for data elements, including data types and constraints; mapping rules and logic, detailing how each source element maps to the target system • <i>Transformation Rules</i> – Detailed descriptions of any transformations required for each data element; formulas, algorithms, or lookup tables used for data transformation; handling of exceptions or special cases in the transformation process; description of any business logic that must be applied to the data during transformation and version history • <i>Validation and Testing Procedures</i> – Testing strategy and procedures to validate data mapping and transformations; metrics for assessing the accuracy and completeness of the data transformations • <i>Transformation and Load Scripts</i> – Transformation and load scripts form the core of the extract, transfer, and load (ETL) process, enabling seamless transformation and loading of data from source system(s) to the destination database • <i>Reference Data Crosswalks</i> – Legacy-to-target code mappings with effective dates
Deliverable	Data Quality Assessment Report	<p>The Contractor must develop a Data Quality Assessment Report that evaluates, documents, and communicates the health of data against defined quality standards so risks are identified and addressed before the data is relied upon for operations, analytics, reporting, or transformation initiatives. The Data Quality Assessment Report must include, at a minimum, the following:</p> <ul style="list-style-type: none"> • <i>Data Profiling</i> – Source data profiling results by domain (completeness, uniqueness, validity, patterns, outliers, referential integrity) that show how the data behaves in practice, highlight systemic issues and anomalies, and quantify the extent of data quality risk • <i>Issue Ranking and Heatmaps</i> – Severity-ranked quality issues with heatmaps to visually prioritize what matters most, enabling stakeholders to focus remediation where it reduces the most risk. • <i>Baseline Quality Scorecards by Domain</i> – Starting-point measurement of data quality for each domain,

		<p>using scorecards that summarize performance against defined data rules and key indicators</p> <ul style="list-style-type: none"> • <i>Remediation Plan</i> – Actionable commitments that convert identified issues into defined remediation steps, with accountable owners and target completion dates • <i>Data Cleansing Strategy and Categorization Rationale</i> – Categorizes identified issues and explains the rationale for chosen cleansing approaches, documenting how each category will be addressed and why specific methods are selected
Deliverable	Dry Run Results	<p>The Contractor must develop Dry Run Results for each data conversion dry run conducted prior to production cutover. Dry Run Results must document the execution and outcomes of each conversion trial run in a production-like environment and must provide evidence that conversion readiness criteria were evaluated and met, or that corrective actions were identified. Dry Run Results must include, at a minimum, the following:</p> <ul style="list-style-type: none"> • <i>Dry Run Scope</i> – Conversion lot, data domains, source systems, target systems, and the data snapshot or extract used • <i>Execution Details</i> – Run date, environment, job identifiers, script and transformation-rule versions, and participating teams • <i>Reconciliation</i> – Record counts and reconciliation at each conversion stage, including extract, transform, load, exception handling, and target validation • <i>Validation Results</i> – Results against the approved Data Acceptance Criteria, including completeness, accuracy, conformity, uniqueness, referential integrity, validity, timeliness, auditability, and lineage • <i>Data Quality Metrics</i> – Exception counts, reject reasons, and unresolved data issues • <i>Defect Summary</i> – Defects by severity, including open defects, approved workarounds, and remediation actions • <i>Performance Results</i> – Whether the conversion completed within the approved cutover window • <i>Restartability and Recovery</i> – Evidence that conversion jobs can restart without data loss or duplication • <i>Rollback and Contingency</i> – Rollback or contingency observations, where applicable • <i>Converted-Data Identification</i> – Integrated validation results where converted data and natively-created target data coexist • <i>Corrective Actions</i> – Actions required before the next dry run or production cutover • <i>Approvals</i> – Data Owner, OHIP, and project approval or sign-off <p>Dry Run Results must be produced after each data conversion dry run; OHIP approval of the applicable Dry Run Results is required before production cutover may proceed.</p>

Deliverable	Final Conversion Report	The Contractor must develop a Final Conversion Report, which provides a comprehensive summary of conversion outcomes, lessons learned, final data quality metrics, unresolved items, and formal approvals.
Artifact	Reconciliation Reports	The Contractor must develop Reconciliation Reports, which are source-to-target reconciliation evidence for each conversion lot, including: record counts, control totals, variance analysis, root causes, and Data Owner signoffs.
Artifact	Unconverted Data Register	The Contractor must develop an Unconverted Data Register, which documents listings of Items not converted, items archived, and dates components were decommissioned, with rationales and owners.

Table 52 – Data Lakehouse Architecture Standard Deliverables and Artifacts

Type	Name	Description
Artifact	Purge Request Documentation	<p>The Contractor must develop Purge Request Documentation, consisting of a formal record of data deletion requests for Lakehouse datasets. This documentation supports compliance with data subject deletion requests (HIPAA right to amendment, potential future state privacy requirements) and ensures complete audit trails for data purge activities. The Purge Request Documentation must include, at a minimum, the following:</p> <ul style="list-style-type: none"> • <i>Request Source and Rationale</i> – Identification of the requestor and the legal or business justification for the purge (e.g., regulatory requirement, data subject request, retention policy expiration) • <i>Affected Records Identification</i> – Detailed identification of affected records across all Lakehouse layers (Raw, Processed, Curated), including record counts, date ranges, and data classifications • <i>Impact Analysis</i> – Assessment of impact on downstream datasets, reports, and consumers that depend on the data to be purged • <i>Approval Chain</i> – Documentation of required approvals from Data Owner and Legal (for PHI/PII), with signatures and approval dates • <i>Purge Execution Plan</i> – Detailed plan for executing the purge, including procedures for addressing time-travel snapshots and historical versions • <i>Verification Evidence</i> – Documentation confirming complete deletion across all layers, including validation queries and checksums • <i>Audit Trail</i> – Complete audit record of the purge execution for compliance purposes, including timestamps, executor identification, and system logs

Table 53 – Content and Document Management Standard Deliverables and Artifacts

Type	Name	Description
------	------	-------------

Deliverable	Content Management Plan	The Contractor must develop a Content Management Plan that defines the approach to content lifecycle management for unstructured and semi-structured content, including capture, classification, repository architecture, access controls, retention, and metadata.
-------------	-------------------------	---

Appendix B: Master Requirements Summary

The following table consolidates all 622 **MUST**, **SHOULD**, and **MAY** requirements across the seven component standards of this domain. Requirements have been re-sequenced with unified DATA-XXXX identifiers for direct reference in Request for Proposals (RFPs), contracts, and compliance audits. The original standard prefix and section reference are preserved in the final two columns for traceability back to each source standard.

Total Requirements: 622 MUST: 524 SHOULD: 56 MAY: 42

Table 54 – Master Requirements Summary Table

Req. ID	Requirement Statement	Tier	Standard	Std. Section
DATA-0001	All data MUST adhere to relevant federal standards to enable interstate consistency and comply with funding requirements, including CMS MITA framework and CMS Streamlined Certification Requirements.	MUST	Data Governance Standard	2.3.1 – Federal Compliance
DATA-0002	Data MUST adhere to Federal Interoperability standards identified by HHS and ONC, and MUST adopt USCDI v4+ as the baseline for certified health IT systems.	MUST	Data Governance Standard	2.3.1 – Federal Compliance
DATA-0003	Data SHOULD align with ONC Interoperability Standards such as USCDI for core data elements and data classes, where applicable, to ensure compliance with national definitions.	SHOULD	Data Governance Standard	2.3.1 – Federal Compliance
DATA-0004	Other federally recognized vocabulary standards (e.g., SNOMED CT, LOINC, RxNorm) and transport standards SHOULD be employed where appropriate.	SHOULD	Data Governance Standard	2.3.1 – Federal Compliance
DATA-0005	The State MUST retain ownership of data, even in Contractor-hosted solutions.	MUST	Data Governance Standard	2.4.1 – Data Ownership and Portability
DATA-0006	The Contractor MUST ensure provisions for regular data extracts or backups in a usable format for State-controlled environments for any SaaS/COTS solution.	MUST	Data Governance Standard	2.4.1 – Data Ownership and Portability
DATA-0007	The Contractor MUST adhere to the OHIP Data Conversion Standard, which governs data export processes, mapping, and transformation rules.	MUST	Data Governance Standard	2.4.1 – Data Ownership and Portability
DATA-0008	The Contractor MUST submit a SaaS Data Conversion Plan for OHIP approval before any conversion activities begin.	MUST	Data Governance Standard	2.4.1 – Data Ownership and Portability

DATA-0009	When configuring data exchange mappings, the Contractor SHOULD use tools or standard middleware instead of hard-coded logic deep in application code.	SHOULD	Data Governance Standard	2.4.2 – Configuration Over Customization
DATA-0010	The Contractor SHOULD utilize COTS configuration features for mapping to external data formats to keep conversion logic maintainable and avoid heavy custom code.	SHOULD	Data Governance Standard	2.4.2 – Configuration Over Customization
DATA-0011	The Contractor MUST establish a process with OHIP to evaluate how cloud solution updates MAY impact data interfaces.	MUST	Data Governance Standard	2.4.3 – SaaS Updates and Versioning
DATA-0012	The Contractor MUST enforce versioned APIs or backward compatibility for data exchange endpoints so that an update does not break the State's data flows.	MUST	Data Governance Standard	2.4.3 – SaaS Updates and Versioning
DATA-0013	The Contractor MUST conduct regression testing for data exchanges before major software upgrades occur.	MUST	Data Governance Standard	2.4.3 – SaaS Updates and Versioning
DATA-0014	Data MUST be strictly controlled when entering the OHIP Enterprise. A System of Record (SOR) MUST be identified for each data element.	MUST	Data Governance Standard	2.5 – Data Creation and Maintenance
DATA-0015	The Contractor MUST make changes to data elements in the SOR and communicate changes to relevant stakeholders.	MUST	Data Governance Standard	2.5 – Data Creation and Maintenance
DATA-0016	Data MAY be assigned to one or more of the defined categories (Master Data, Reference Data, Metadata, Transaction Data, Derived Data) as applicable to the data element.	MAY	Data Governance Standard	2.5.1 – Standard Definitions and Guidelines for SOR
DATA-0017	Data ownership and accountability MUST be assigned and executed per OHIP.	MUST	Data Governance Standard	2.5.1 – Standard Definitions and Guidelines for SOR
DATA-0018	Data corrections MUST be made in the SOR with business rules enforced, including data validation rules, default values, and referential integrity rules.	MUST	Data Governance Standard	2.5.1 – Standard Definitions and Guidelines for SOR
DATA-0019	The SOR MUST preserve for each record: the name of the staff or system that changed the data, and the date and time the change occurred.	MUST	Data Governance Standard	2.5.1 – Standard Definitions and Guidelines for SOR
DATA-0020	When a data value is corrected, changes SHOULD be communicated as appropriate and made available to all end-users.	SHOULD	Data Governance Standard	2.5.1 – Standard Definitions and Guidelines for SOR

DATA-0021	The SOR MUST be defined for each individual data element.	MUST	Data Governance Standard	2.5.1 – Standard Definitions and Guidelines for SOR
DATA-0022	The SOR MUST include functionality to correct data through a system interface rather than through manual or scripted database changes that might bypass security and validation rules.	MUST	Data Governance Standard	2.5.1 – Standard Definitions and Guidelines for SOR
DATA-0023	Derived data elements MUST be preserved. The solution that initially creates a derived data element MUST become the SOR for that element. Systems other than the SOR MUST NOT recreate intermediate results the SOR did not preserve.	MUST	Data Governance Standard	2.5.1 – Standard Definitions and Guidelines for SOR
DATA-0024	Business data that supports a business event MUST be preserved so that reference data values effective when the event occurred can be retrieved later.	MUST	Data Governance Standard	2.5.1 – Standard Definitions and Guidelines for SOR
DATA-0025	Data parameters that impact SOR data values MUST be preserved so that parameter values in effect at a specific time can be determined.	MUST	Data Governance Standard	2.5.1 – Standard Definitions and Guidelines for SOR
DATA-0026	Additive data MUST be changed using a cancel-and-correct approach (negating the previous amount and adding the new amount) to maintain a proper audit trail.	MUST	Data Governance Standard	2.5.1 – Standard Definitions and Guidelines for SOR
DATA-0027	Data MUST be created or modified within authoritative sources only.	MUST	Data Governance Standard	2.5.1 – Standard Definitions and Guidelines for SOR
DATA-0028	Data security and sensitivity classification of data sources MUST be published and easily accessible by users.	MUST	Data Governance Standard	2.5.1 – Standard Definitions and Guidelines for SOR
DATA-0029	Data users and business processes MUST use business information only from approved authoritative source(s).	MUST	Data Governance Standard	2.5.1 – Standard Definitions and Guidelines for SOR
DATA-0030	OHIP MUST monitor and review data sources annually to maintain security, sensitivity, access classification, and report compliance.	MUST	Data Governance Standard	2.5.1 – Standard Definitions and Guidelines for SOR
DATA-0031	Solutions MUST include the ability for appropriate users to make changes to data in a controlled manner. Changes MUST follow a defined change control process and notification.	MUST	Data Governance Standard	2.6 – Data Corrections

DATA-0032	A tracking tool MUST be used to log, track, and report requests, changes, and approvals of data corrections.	MUST	Data Governance Standard	2.6 – Data Corrections
DATA-0033	Corrections to data MUST be made at the SOR and propagated to all data users. Downstream solutions MUST reflect corrected data.	MUST	Data Governance Standard	2.6 – Data Corrections
DATA-0034	The Contractor MUST ensure a process for archiving and purging data is in place per all pertinent retention policies, regulatory and legal requirements, and non-destruct and retention policies issued by the State or OHIP.	MUST	Data Governance Standard	2.6.1 – Data Purge, Archive, Retention
DATA-0035	Data retirement, purge, and archiving rules MUST be fully documented, traced, and logged.	MUST	Data Governance Standard	2.6.1 – Data Purge, Archive, Retention
DATA-0036	Data archival MUST be based on a business unit of work and tied to a business event.	MUST	Data Governance Standard	2.6.1 – Data Purge, Archive, Retention
DATA-0037	Archive locations MUST be registered, dated, and maintained by the respective business units or records custodians.	MUST	Data Governance Standard	2.6.1 – Data Purge, Archive, Retention
DATA-0038	OHIP review and approval MUST be provided before retirement, purging, archival, and significant changes occur.	MUST	Data Governance Standard	2.6.1 – Data Purge, Archive, Retention
DATA-0039	The Contractor MUST develop and maintain a Data Management Plan providing a structured approach to handling data throughout its lifecycle.	MUST	Data Governance Standard	2.7 – Data Management Plan Requirements
DATA-0040	The Contractor MUST submit the Data Management Plan for Department review and approval prior to database design activities.	MUST	Data Governance Standard	2.7 – Data Management Plan Requirements
DATA-0041	The Contractor MUST categorize all data domains and data elements per Section 2.11 policies and standards.	MUST	Data Governance Standard	2.9 – Master Data and Reference Data Management Guidelines
DATA-0042	All master data domains and elements MUST have an assigned SOR as the Authoritative Source.	MUST	Data Governance Standard	2.9 – Master Data and Reference Data Management Guidelines
DATA-0043	Data categorized as master data MUST be consistently defined. The BEDG MUST approve any changes to definitions.	MUST	Data Governance Standard	2.9 – Master Data and Reference Data Management Guidelines
DATA-0044	All downstream solution consumers and data users MUST source and use master data from the SOR.	MUST	Data Governance Standard	2.9 – Master Data and Reference Data Management Guidelines

DATA-0045	Master data MUST have specific access rights, data quality rules, and data quality thresholds documented, published, and maintained.	MUST	Data Governance Standard	2.9 – Master Data and Reference Data Management Guidelines
DATA-0046	Data MUST be created, updated, purged, and maintained in the master SOR.	MUST	Data Governance Standard	2.9 – Master Data and Reference Data Management Guidelines
DATA-0047	Metadata MUST be supplied for each master data element as defined in Section 2.11.	MUST	Data Governance Standard	2.9 – Master Data and Reference Data Management Guidelines
DATA-0048	Master data domains and elements MUST be audited quarterly for data quality and OHIP Data Governance compliance. A Corrective Action Plan MUST be developed when exceptions are found.	MUST	Data Governance Standard	2.9 – Master Data and Reference Data Management Guidelines
DATA-0049	Hierarchies MUST be standardized and controlled, named consistently, utilized for regulatory reporting, and structured with at least one parent and one child.	MUST	Data Governance Standard	2.9 – Master Data and Reference Data Management Guidelines
DATA-0050	Alternate hierarchies MAY be identified where they exist; any alternate hierarchy MUST include the required exception from the State and the purpose for reporting.	MAY	Data Governance Standard	2.9 – Master Data and Reference Data Management Guidelines
DATA-0051	The Contractor MUST adopt and maintain enterprise-standard code sets as designated by OHIP (ICD, CPT, NDC, and federally/State mandated code sets).	MUST	Data Governance Standard	2.9 – Master Data and Reference Data Management Guidelines
DATA-0052	The Contractor MUST use enterprise-standard code sets where they exist and MUST NOT create custom code sets.	MUST	Data Governance Standard	2.9 – Master Data and Reference Data Management Guidelines
DATA-0053	All code set mappings and crosswalks MUST be documented, submitted to BEDG for review, and maintained in the Enterprise Data Catalog.	MUST	Data Governance Standard	2.9 – Master Data and Reference Data Management Guidelines
DATA-0054	Data exchange formats MUST comply with applicable federal and national standards including HL7, HIPAA EDI X12, and other standards as specified by OHIP.	MUST	Data Governance Standard	2.9 – Master Data and Reference Data Management Guidelines
DATA-0055	The Contractor MUST ensure that metadata is collected and standardized for all data domains and elements.	MUST	Data Governance Standard	2.10 – Metadata Management
DATA-0056	All metadata categories MUST be defined and managed in an OHIP-approved solution or tool.	MUST	Data Governance Standard	2.10 – Metadata Management

DATA-0057	Where a data element requires metadata spanning both Section 2.0 and Section 3.0, the Contractor MUST satisfy requirements of both. In any apparent conflict, the BEDG MUST issue a binding determination.	MUST	Data Governance Standard	2.10.1 – Categories of Metadata
DATA-0058	The following business metadata SHOULD be defined, documented, monitored, and reported as appropriate: business name, business definition, authoritative sources, data ownership roles and accountability, data ownership organization(s), data category, and data security classification.	SHOULD	Data Governance Standard	2.10.1.1 – Business Metadata
DATA-0059	Additional metadata MAY be defined within specific procurement-related documents beyond the categories established in Section 2.12.	MAY	Data Governance Standard	2.10.1.1 – Metadata Management
DATA-0060	The following data quality metadata SHOULD be defined, documented, monitored, and reported as appropriate: allowable values, data quality rules, data quality targets/objectives, data quality triggers, and data quality scores.	SHOULD	Data Governance Standard	2.10.1.2 – Data Quality Metadata
DATA-0061	The following technical metadata SHOULD be defined, documented, monitored, and reported as appropriate: DB name/IP/URL, schemas, tables and field relationships, data transformation and lineage, derivations and calculations, security settings, partitions and archiving settings, alias information, and volumetrics.	SHOULD	Data Governance Standard	2.10.1.3 – Technical Metadata
DATA-0062	Technical metadata MAY additionally include information on user roles, frequency of access, and transformations applied during data lineage tracing.	MAY	Data Governance Standard	2.10.1.3 – Technical Metadata
DATA-0063	The following operational metadata SHOULD be defined, documented, monitored, and reported as appropriate: data availability and refresh rates, job schedule information, expected completion times, list of jobs with run sequences and dependencies, and location of error logs.	SHOULD	Data Governance Standard	2.10.1.4 – Operational Metadata
DATA-0064	The Contractor SHOULD adhere to OHIP's standardized data naming conventions per the OHIP Data Architecture and Naming Standard.	SHOULD	Data Governance Standard	2.11 – Data Models
DATA-0065	The Contractor MUST create, update, and maintain data models per Section 3.0, which defines authoritative requirements for Conceptual, Logical, Physical, and Canonical Data Models.	MUST	Data Governance Standard	2.11 – Data Models

DATA-0066	Mapping and business rules, transformation scripts, source code, and technical logs for data movement between solutions MUST be documented and published in a central OHIP-managed repository.	MUST	Data Governance Standard	2.12 – Data Integration and Controls
DATA-0067	Data movement from one IT-supported data store to another SHOULD be automated and verified for completeness and accuracy through controls comparing source to target.	SHOULD	Data Governance Standard	2.12 – Data Integration and Controls
DATA-0068	Controls MUST include, at minimum, detection, notification, and management of exceptions when moving information from source to target.	MUST	Data Governance Standard	2.12 – Data Integration and Controls
DATA-0069	PHI and PII data MUST be encrypted at rest and in transit from source to destination per the OHIP Security and Privacy Compliance Standard.	MUST	Data Governance Standard	2.12 – Data Integration and Controls
DATA-0070	Proposed exceptions to data integration controls MUST be escalated to OHIP for final determination.	MUST	Data Governance Standard	2.12 – Data Integration and Controls
DATA-0071	The Contractor MUST apply the seven data quality dimensions (Accessibility, Accuracy, Completeness, Conformance and Validity, Consistency, Currency and Timeliness, and Uniqueness and Integrity) when defining data quality rules, thresholds, and metrics for all enterprise data elements.	MUST	Data Governance Standard	2.13 – Data Quality
DATA-0072	The data quality dimensions defined in Table 8 MUST be referenced in the data quality metadata required under Section 2.12 and in the quality gate thresholds defined under Section 7.6.	MUST	Data Governance Standard	2.13.1 – Data Quality Standards
DATA-0073	Solutions MUST include both preventative controls (blocking invalid data) and detective controls (identifying and supporting correction of errors).	MUST	Data Governance Standard	2.13.1 – Data Quality Standards
DATA-0074	Data quality issues MUST be monitored, logged, and tracked using OHIP-designated standard tools.	MUST	Data Governance Standard	2.13.1 – Data Quality Standards
DATA-0075	Data quality metrics, standards, targets, and acceptable thresholds SHOULD be defined for all Enterprise data elements.	SHOULD	Data Governance Standard	2.13.1 – Data Quality Standards
DATA-0076	Data quality performance SHOULD be reported monthly, or according to approved processes agreed with OHIP.	SHOULD	Data Governance Standard	2.13.1 – Data Quality Standards

DATA-0077	The Data Owner SHOULD request a data quality review with OHIP to ensure the integrity of core data elements is maintained throughout the data lineage.	SHOULD	Data Governance Standard	2.13.1 – Data Quality Standards
DATA-0078	Data quality rules SHOULD be collected, validated, reconciled, maintained, and reviewed quarterly by OHIP.	SHOULD	Data Governance Standard	2.13.1 – Data Quality Standards
DATA-0079	The Contractor MUST implement data quality controls spanning all three types (Preventive, Detective, and Corrective) for all enterprise data domains.	MUST	Data Governance Standard	2.13.1 – Data Quality Standards
DATA-0080	All corrective actions MUST be logged and traceable to the originating detective or preventive control finding that identified the issue.	MUST	Data Governance Standard	2.13.1 – Data Quality Standards
DATA-0081	All data quality issues identified at or above the High criticality threshold defined in Section 7.6 MUST be subject to a documented root cause analysis within ten (10) business days of identification. Where a data quality issue is also a logged incident, root cause analysis is performed within the OHIP ITSM Standard's incident and problem-management process.	MUST	Data Governance Standard	2.13.2 Root Cause Analysis and Continuous Improvement
DATA-0082	Root cause findings MUST be recorded in the OHIP-designated data quality tracking tool and MUST include: a description of the issue, the system or process where the defect originated, the contributing factor (technical, process, or human), and the corrective or preventive action taken.	MUST	Data Governance Standard	2.13.2 Root Cause Analysis and Continuous Improvement
DATA-0083	Where root cause analysis identifies a systemic defect in a business process or technical interface, the Contractor MUST submit a remediation plan to OHIP within fifteen (15) business days proposing process or system corrections to prevent recurrence. This data-governance remediation-plan commitment operates within, but is not duplicated by, the OHIP ITSM Standard's problem-management framework.	MUST	Data Governance Standard	2.13.2 Root Cause Analysis and Continuous Improvement
DATA-0084	Data quality improvement actions SHOULD be reviewed quarterly alongside the data quality rule review required under Section 2.15 to assess whether implemented corrections have reduced defect rates to within acceptable thresholds.	SHOULD	Data Governance Standard	2.13.2 Root Cause Analysis and Continuous Improvement

DATA-0085	The Contractor SHOULD apply the Plan-Do-Check-Act (PDCA) improvement cycle when managing recurring data quality issues: planning remediation based on root cause assessment, implementing corrections, monitoring outcomes against defined thresholds, and initiating a new cycle when quality falls below acceptable levels or when new data sets or business rules introduce fresh quality requirements.	SHOULD	Data Governance Standard	2.13.2 Root Cause Analysis and Continuous Improvement
DATA-0086	The frequency at which data quality monitoring is performed MUST be defined for each monitored data domain and documented in the Data Management Plan required under Section 2.9, based on all four factors: utilization frequency, operational importance, data volatility, and cost and operational impact of monitoring.	MUST	Data Governance Standard	2.13.3 Continuous Monitoring Frequency
DATA-0087	Data underpinning T-MSIS submissions, CMS-64 reporting, PERM audits, or eligibility determinations MUST be monitored no less than weekly, with automated alerts triggered on threshold breaches.	MUST	Data Governance Standard	2.13.3 Continuous Monitoring Frequency
DATA-0088	Data supporting routine internal operations SHOULD be monitored no less than monthly.	SHOULD	Data Governance Standard	2.13.3 Continuous Monitoring Frequency
DATA-0089	Where monitoring costs are significant, the Contractor MUST document the trade-off analysis and proposed monitoring approach for OHIP approval before reducing monitoring frequency below the minimums in Section 2.15.3.	MUST	Data Governance Standard	2.13.3 Continuous Monitoring Frequency
DATA-0090	Monitoring frequency decisions MUST be reviewed annually and updated whenever a material change occurs in data utilization patterns, system architecture, or federal reporting requirements.	MUST	Data Governance Standard	2.13.3 Continuous Monitoring Frequency
DATA-0091	When data history is required, the Contractor MUST capture original values as well as all changes.	MUST	Data Governance Standard	2.14 – Data History
DATA-0092	Data history MUST use an insertion-only approach: records cannot be updated or deleted. Any changes or corrections MUST create a new history record that is date- and time-stamped.	MUST	Data Governance Standard	2.14 – Data History
DATA-0093	No destructive changes to data history (such as updating business data in place) are permitted.	MUST	Data Governance Standard	2.14 – Data History

DATA-0094	When data is deleted in the SOR, versions in data history MUST NOT be physically deleted: they MUST be logically deleted.	MUST	Data Governance Standard	2.14 – Data History
DATA-0095	A delete flag indicator MAY be set on a history record to designate it as logically deleted, as one valid approach to implementing logical deletion.	MAY	Data Governance Standard	2.14 – Data History
DATA-0096	Data history MUST be traceable to the SOR(s) regardless of SOR location.	MUST	Data Governance Standard	2.14 – Data History
DATA-0097	The Contractor MUST ensure data requirements for new and modified business capabilities are defined precisely enough to verify solutions, interfaces, and databases satisfy OHIP's Data Standards.	MUST	Data Governance Standard	2.15 – Defining and Capturing Data Requirements
DATA-0098	Business data requirements for new projects MUST adhere to the governance deliverables specified in the applicable lifecycle (e.g., Waterfall, Agile, Hybrid) and MUST include detailed data requirements.	MUST	Data Governance Standard	2.15 – Defining and Capturing Data Requirements
DATA-0099	Process data requirements SHOULD describe how each process uses information, including governance of information, how current data needs to be, whether the process creates or updates data, and how long data SHOULD be retained.	SHOULD	Data Governance Standard	2.15 – Defining and Capturing Data Requirements
DATA-0100	OHIP approval SHOULD be obtained for business requirements that are traceable to physical implementation, including every physical database and data feed created or used by a project.	SHOULD	Data Governance Standard	2.15 – Defining and Capturing Data Requirements
DATA-0101	Data requirements SHOULD include master and reference data, metadata, and data quality requirements described in previous sections.	SHOULD	Data Governance Standard	2.15 – Defining and Capturing Data Requirements
DATA-0102	During the project requirements phase, data owners, data managers, data suppliers, and data consumers are identified to assist in defining and reviewing data requirements for master data, reference data, metadata, transaction data, and derived data.	SHOULD	Data Governance Standard	2.15 – Defining and Capturing Data Requirements
DATA-0103	A Test Management Plan MUST be submitted to OHIP prior to initiating any testing activities.	MUST	Data Governance Standard	2.16 – Test Data
DATA-0104	Approval from OHIP's Bureau of Security and Bureau of Privacy and Compliance MUST be obtained before any sensitive or production data is sent to the Contractor.	MUST	Data Governance Standard	2.16 – Test Data

DATA-0105	The Contractor MUST obtain OHIP approval of the Test Management Plan, addressing all data specifications, before extracting any production or sensitive data for testing.	MUST	Data Governance Standard	2.16 – Test Data
DATA-0106	The Contractor MUST identify the databases, tables, and fields to be extracted and MUST specify selection criteria and required referential integrity between tables.	MUST	Data Governance Standard	2.16 – Test Data
DATA-0107	The Contractor MUST collaborate with database administrators (DBAs) to extract, transform, and load data into appropriate test environments.	MUST	Data Governance Standard	2.16 – Test Data
DATA-0108	The Contractor SHOULD coordinate with OHIP or other Contractors whether test data transformation (cleansing, de-identifying, or other modifications) is required and SHOULD perform these activities as assigned.	SHOULD	Data Governance Standard	2.16 – Test Data
DATA-0109	The Contractor SHOULD evaluate whether cleansed data is required for the tests. When realistic data quality issues are needed to validate error handling, test data SHOULD appropriately reflect those conditions.	SHOULD	Data Governance Standard	2.16 – Test Data
DATA-0110	The Contractor MUST de-identify PII, PHI, FTI, and other sensitive data unless operating in a production environment. The Contractor MUST specify fields requiring de-identification and document methodology and rules in the Test Management Plan.	MUST	Data Governance Standard	2.16 – Test Data
DATA-0111	The Contractor MUST designate authorized personnel who MAY handle sensitive data during test data preparation, execution, or review.	MUST	Data Governance Standard	2.16 – Test Data
DATA-0112	Full security controls MUST be applied to any test environments, databases, and personnel involved in handling sensitive data.	MUST	Data Governance Standard	2.16 – Test Data
DATA-0113	The Contractor MUST develop a RACI chart for test data loading and management within shared test environments.	MAY	Data Governance Standard	2.16 – Test Data
DATA-0114	An OHIP-approved tool MUST be used to track all test datasets, including versions, updates, refreshes, and datasets created by other tests.	MUST	Data Governance Standard	2.16 – Test Data

DATA-0115	Test results MUST identify the specific test dataset version used to enable effective defect management and oversight of data fixes.	MUST	Data Governance Standard	2.16 – Test Data
DATA-0116	The Contractor MUST develop a Data Management Plan covering all required components defined in Appendix A, including data scope, risk assessment, standards, tools, data inventory, lineage, naming, MDM keys, data quality management, model versioning, audit and logging, backup, monitoring, and access controls.	MUST	Data Governance Standard	Appendix A
DATA-0117	The Contractor MUST produce and submit all Deliverables in the format, structure, and cadence specified by OHIP. Each Deliverable MUST satisfy all content requirements and acceptance criteria and is subject to formal OHIP review and approval.	MUST	Data Governance Standard	Appendix A
DATA-0118	Upon notice of modifications to Deliverables and Artifacts requirements, the Contractor MUST comply with all modifications and implement them promptly.	MUST	Data Governance Standard	Appendix A
DATA-0119	The Contractor MUST align all named data elements across three levels (Business Name, Logical Name, and Physical Name), ensuring each level is consistent with the others and that the physical name retains the business meaning of the element it represents.	MUST	Data Architecture and Naming Standard	3.1 Introduction
DATA-0120	The Contractor MUST develop conceptual and logical data models in sequence before creating a physical data model	MUST	Data Architecture and Naming Standard	3.1 Introduction
DATA-0121	The Contractor MUST develop a Data Name Abbreviations List including both the abbreviation and the complete name to standardize use of abbreviations.	MUST	Data Architecture and Naming Standard	3.4 – General Data Naming Principles and Standards
DATA-0122	The Contractor MUST adhere to OHIP data naming conventions and standards, including all principles defined in Section 3.4.	MUST	Data Architecture and Naming Standard	3.4.1 – Data Object Naming Principles
DATA-0123	Naming conventions MUST NOT result in conflict within the internal context of OHIP and its partner ecosystem.	MUST	Data Architecture and Naming Standard	3.4.1 – Data Object Naming Principles

DATA-0124	Common fields across different databases MUST have consistent naming, and business context MUST be considered when naming.	MUST	Data Architecture and Naming Standard	3.4.1 – Data Object Naming Principles
DATA-0125	Names SHOULD align with policy documents when possible.	SHOULD	Data Architecture and Naming Standard	3.4.1 – Data Object Naming Principles
DATA-0126	Names MUST be meaningful for a broad range of technical and non-technical staff while representing the business purpose.	MUST	Data Architecture and Naming Standard	3.4.1 – Data Object Naming Principles
DATA-0127	Names MUST be self-descriptive. The name MUST provide comprehension of what it represents without having to read the description, and MUST be extracted from the business usage and purpose of the element.	MUST	Data Architecture and Naming Standard	3.4.1 – Data Object Naming Principles
DATA-0128	Names MUST be unambiguous. People from different backgrounds reading the name at different times MUST have the same understanding of what it means.	MUST	Data Architecture and Naming Standard	3.4.1 – Data Object Naming Principles
DATA-0129	Any name conflict with an existing data element MUST be resolved immediately to allow development to continue.	MUST	Data Architecture and Naming Standard	3.4.1 – Data Object Naming Principles
DATA-0130	The Contractor SHOULD add descriptive in-line and data dictionary comments as necessary, regardless of how well the name describes the element.	SHOULD	Data Architecture and Naming Standard	3.4.1 – Data Object Naming Principles
DATA-0131	The Contractor MUST create data models that accurately represent business data usage and MUST ensure data model elements use full and descriptive English-language names that accurately convey the meaning of corresponding business concepts.	MUST	Data Architecture and Naming Standard	3.4.2 – Data Objects Naming Standards
DATA-0132	Data elements MUST be meaningful and derived from business use or purpose.	MUST	Data Architecture and Naming Standard	3.4.2 – Data Objects Naming Standards
DATA-0133	Data elements MUST use singular names reflecting a single instance of the object (e.g., 'PATIENT' not 'PATIENTS'), except in subject areas which MUST be expressed in plural form.	MUST	Data Architecture and Naming Standard	3.4.2 – Data Objects Naming Standards
DATA-0134	Standard qualifiers MAY be appended at the end of a name where useful, e.g., character type (CT), numeric value (NUM), description (DESC), identifier (ID),	MAY	Data Architecture and Naming Standard	3.4.2 – Data Objects Naming Standards

	currency date (CD), indicator (IND), amount (AMT).			
DATA-0135	Data elements MUST NOT use special characters (e.g., / \ * # \$) except an underscore, and MUST NOT begin with a numeric character.	MUST	Data Architecture and Naming Standard	3.4.2 – Data Objects Naming Standards
DATA-0136	Abbreviation algorithms MUST NOT be used except in rare cases where name-size limits necessitate their use.	MUST	Data Architecture and Naming Standard	3.4.2 – Data Objects Naming Standards
DATA-0137	Physical data element names MUST use approved class word abbreviations as the final term where applicable (e.g., DT for Date, CD for Code, AMT for Amount, NM for Name). Alternative abbreviations for approved class words MUST NOT be used without OHIP approval.	MUST	Data Architecture and Naming Standard	3.4.2 – Data Objects Naming Standards
DATA-0138	Class words not listed in the approved class word table MUST be submitted to OHIP and BEDG for approval before use and MUST be added to the Enterprise Glossary prior to adoption.	MUST	Data Architecture and Naming Standard	3.4.2 – Data Objects Naming Standards
DATA-0139	The Contractor MUST use well-documented or broadly understood abbreviations and acronyms rather than abbreviations relying on tribal knowledge.	MUST	Data Architecture and Naming Standard	3.4.3 – Abbreviations and Acronyms
DATA-0140	Abbreviations MUST be applied consistently across models, databases, and development artifacts.	MUST	Data Architecture and Naming Standard	3.4.3 – Abbreviations and Acronyms
DATA-0141	Words less than eight (8) characters in length SHOULD NOT be abbreviated.	SHOULD	Data Architecture and Naming Standard	3.4.3 – Abbreviations and Acronyms
DATA-0142	When abbreviating single word names, the Contractor MUST : check for existing standard acronyms in the Enterprise Glossary; remove all special characters; remove all vowels except the leading vowel(s); replace multiple consonants with a single consonant; and truncate after a sufficiently descriptive number of letters.	MUST	Data Architecture and Naming Standard	3.4.3.1 – Single Word Abbreviation Rules
DATA-0143	When abbreviating multiple word names, the Contractor MUST apply single-word abbreviation rules to each word.	MUST	Data Architecture and Naming Standard	3.4.3.2 – Multiple Word Abbreviation Rules
DATA-0144	For names with more than two words, the first letter of each word MAY be used as the abbreviation, if appropriate.	MAY	Data Architecture and Naming Standard	3.4.3.2 – Multiple Word Abbreviation Rules

DATA-0145	If an abbreviated word conflicts with a popular acronym, the Contractor MUST avoid the conflicting acronym and create a new abbreviation.	MUST	Data Architecture and Naming Standard	3.4.3.3 – Other Abbreviation Considerations
DATA-0146	When applying abbreviation rules, the Contractor MUST verify abbreviation results against the Enterprise Glossary and MUST resolve conflicts where two independently derived abbreviations produce the same result. Conflicting abbreviations MUST be documented and submitted to OHIP for approval before use.	MUST	Data Architecture and Naming Standard	3.4.3.3 – Other Abbreviation Considerations
DATA-0147	The Contractor MUST adhere to business-focused naming policies at the conceptual and logical data model levels.	MUST	Data Architecture and Naming Standard	3.5 – Conceptual and Logical Naming Standards
DATA-0148	The Contractor MUST ensure entity names are unique within the enterprise.	MUST	Data Architecture and Naming Standard	3.5 – Conceptual and Logical Naming Standards
DATA-0149	The Contractor SHOULD leverage any existing, applicable names from the Enterprise Data Model (EDM) and Enterprise Glossary when developing new conceptual and logical models.	SHOULD	Data Architecture and Naming Standard	3.5 – Conceptual and Logical Naming Standards
DATA-0150	Industry acronyms well understood in context (e.g., SSN, W-2) MAY be used; however, fully spelled-out names SHOULD be used where appropriate.	MAY	Data Architecture and Naming Standard	3.5 – Conceptual and Logical Naming Standards
DATA-0151	The Contractor MUST adhere to all entity naming requirements in Section 3.5.1, including conformance to ISO/IEC 11179-5:2015.	MUST	Data Architecture and Naming Standard	3.5.1 – Subject Areas and Entities
DATA-0152	Entity names MUST be unique across all OHIP models. If a chosen name already exists, the two entities MUST be combined or one MUST be renamed where OHIP has control.	MUST	Data Architecture and Naming Standard	3.5.1 – Subject Areas and Entities
DATA-0153	Entity names MUST NOT begin with a numeric character and MUST use business terminology.	MUST	Data Architecture and Naming Standard	3.5.1 – Subject Areas and Entities
DATA-0154	Entity names MUST use singular nouns (e.g., 'Provider') and subject areas MUST use the plural form.	MUST	Data Architecture and Naming Standard	3.5.1 – Subject Areas and Entities
DATA-0155	Entity names MUST use mixed case or uppercase characters with only an underscore or space as a delimiter between words.	MUST	Data Architecture and Naming Standard	3.5.1 – Subject Areas and Entities

DATA-0156	Entity names MUST use only alphabetic characters and an underscore. Numbers MUST only be included when they add essential clarity.	MUST	Data Architecture and Naming Standard	3.5.1 – Subject Areas and Entities
DATA-0157	Mixed case MAY be used for entity names; if mixed case is used, the first letter of each word MUST be capitalized (e.g., 'Status').	MAY	Data Architecture and Naming Standard	3.5.1 – Subject Areas and Entities
DATA-0158	The Contractor MUST adhere to all attribute naming requirements in Section 3.5.2, including compliance with general naming standards and use of acronyms or abbreviations that comply with Section 3.4.3.	MUST	Data Architecture and Naming Standard	3.5.2 – Attributes
DATA-0159	Attribute names MUST be unique within their parent entity and MUST use business terminology aligned to the EDM and/or glossary.	MUST	Data Architecture and Naming Standard	3.5.2 – Attributes
DATA-0160	Attribute names MUST contain only singular nouns, MUST use mixed case or uppercase, and MUST use a space as the delimiter between words.	MUST	Data Architecture and Naming Standard	3.5.2 – Attributes
DATA-0161	Attribute names MUST use only alphabetic characters, spaces, and underscores. Numbers MUST only be included when they add essential clarity.	MUST	Data Architecture and Naming Standard	3.5.2 – Attributes
DATA-0162	The Contractor MUST name both directions of each relationship separately.	MUST	Data Architecture and Naming Standard	3.5.3 – Relationships
DATA-0163	Relationship names MUST be meaningful and convey the business rule between entities. Generic names such as 'related to' or 'associated with' MUST NOT be used.	MUST	Data Architecture and Naming Standard	3.5.3 – Relationships
DATA-0164	Relationship names MUST use lowercase characters with a space as the delimiter, and MUST use only alphabetic characters, spaces, and underscores.	MUST	Data Architecture and Naming Standard	3.5.3 – Relationships
DATA-0165	Relationship names MUST NOT exceed thirty (30) characters in length.	MUST	Data Architecture and Naming Standard	3.5.3 – Relationships
DATA-0166	Naming supertype/subtype relationships is optional: this MAY be omitted where not necessary.	MAY	Data Architecture and Naming Standard	3.5.3 – Relationships
DATA-0167	The Contractor MUST comply with all schema naming requirements in Section	MUST	Data Architecture	3.6.1 – Database Schemas

	3.6.1, including use of general naming standards and consistent use of prefixes.		and Naming Standard	
DATA-0168	Schema names MUST be descriptive and meaningful, MUST avoid reserved keywords or unsupported special characters, and MUST use alphanumeric characters with underscore.	MUST	Data Architecture and Naming Standard	3.6.1 – Database Schemas
DATA-0169	Schema names MUST avoid spaces, special characters, or punctuation marks that complicate schema referencing.	MUST	Data Architecture and Naming Standard	3.6.1 – Database Schemas
DATA-0170	Schema names SHOULD be kept reasonably short and concise, not exceeding database management solution limitations.	SHOULD	Data Architecture and Naming Standard	3.6.1 – Database Schemas
DATA-0171	The Contractor MUST comply with all table naming requirements in Section 3.6.2, including compliance with general naming standards and use of unique names within the database schema.	MUST	Data Architecture and Naming Standard	3.6.2 – Tables
DATA-0172	Table names MUST be unique within the database schema; enterprise data table names MUST be unique within OHIP.	MUST	Data Architecture and Naming Standard	3.6.2 – Tables
DATA-0173	Table names MUST use singular form in uppercase or mixed case, capitalizing the first letter of each word per database management software capabilities.	MUST	Data Architecture and Naming Standard	3.6.2 – Tables
DATA-0174	The Contractor SHOULD leverage data modeling tools to derive physical table names from logical names using the glossary of abbreviations when possible.	SHOULD	Data Architecture and Naming Standard	3.6.2 – Tables
DATA-0175	Naming supertype/subtype constraints is optional: this MAY be omitted where not necessary for the physical model.	MAY	Data Architecture and Naming Standard	3.6.2 – Tables
DATA-0176	Physical data models MAY include denormalization of data to improve performance (e.g., adding columns to eliminate joins). Physical object names are subject to limitations of the storage environment.	MAY	Data Architecture and Naming Standard	3.6.2 – Tables
DATA-0177	View and materialized view names MUST be descriptive, reflecting the purpose, content, or functionality of the view. A suffix indicating type MUST be included (e.g., '_V' for views, '_MV' for materialized views).	MUST	Data Architecture and Naming Standard	3.6.2.1 – Views/Materialized Views

DATA-0178	View and materialized view names MUST use consistent naming conventions. A suffix indicating type MUST be applied uniformly across all views in the system.	MUST	Data Architecture and Naming Standard	3.6.2.1 – Views/Materialized Views
DATA-0179	Where a view draws from multiple tables, the view name MUST reflect the combination of primary source tables (e.g., CLAIM_MEMBER_V). Generic or non-descriptive view names such as RPT_V or TEMP_V are NOT permitted.	MUST	Data Architecture and Naming Standard	3.6.2.1 – Views/Materialized Views
DATA-0180	Where a public synonym is created for a view, the synonym MUST use the exact view name including its type suffix.	MUST	Data Architecture and Naming Standard	3.6.2.1 – Views/Materialized Views
DATA-0181	The Contractor MUST comply with all column naming requirements in Section 3.6.3, including compliance with general naming standards and use of unique, singular names within the parent table.	MUST	Data Architecture and Naming Standard	3.6.3 – Columns
DATA-0182	Column names MUST be unique and singular within the parent table.	MUST	Data Architecture and Naming Standard	3.6.3 – Columns
DATA-0183	Multiple words in column names MUST be separated using underscore as a delimiter (e.g., date_of_birth).	MUST	Data Architecture and Naming Standard	3.6.3 – Columns
DATA-0184	The Contractor MUST comply with all foreign key column naming requirements in Section 3.6.3.1.	MUST	Data Architecture and Naming Standard	3.6.3.1 – Foreign Key Column Names
DATA-0185	Foreign key columns MUST use the same name and data type as the column in the referenced table unless multiple relationships exist between the two entities.	MUST	Data Architecture and Naming Standard	3.6.3.1 – Foreign Key Column Names
DATA-0186	A foreign key constraint MUST be defined on the column to ensure data integrity.	MUST	Data Architecture and Naming Standard	3.6.3.1 – Foreign Key Column Names
DATA-0187	When the foreign key is a system-generated primary key ('ID'), the column name MUST follow the format: (parent table name) ID (e.g., 'ORGANIZATION ID', 'Person ID').	MUST	Data Architecture and Naming Standard	3.6.3.1 – Foreign Key Column Names
DATA-0188	Duplicate column names from multiple foreign key relationships between two tables MUST be resolved by role naming.	MUST	Data Architecture and Naming Standard	3.6.3.1 – Foreign Key Column Names

DATA-0189	System-generated surrogate primary keys MUST use the suffix KEY (e.g., CLAIM_KEY). Externally assigned identifiers MUST use the suffix ID (e.g., MEMBER_ID). These suffixes MUST NOT be used interchangeably.	MUST	Data Architecture and Naming Standard	3.6.3.2 – Surrogate Key Naming
DATA-0190	Slowly changing dimension surrogate keys in the data warehouse or Lakehouse environment MUST use the suffix HIST_KEY to distinguish them from operational surrogate keys (e.g., MEMBER_HIST_KEY).	MUST	Data Architecture and Naming Standard	3.6.3.2 – Surrogate Key Naming
DATA-0191	Data mart dimension surrogate keys MUST use the suffix DIM_KEY. DIM_KEY columns MUST NOT appear in warehouse-layer tables.	MUST	Data Architecture and Naming Standard	3.6.3.2 – Surrogate Key Naming
DATA-0192	Where a table contains both a surrogate key (KEY) and an externally assigned identifier (ID), both columns MUST be present and distinctly named. The surrogate key serves as the physical primary key; the external identifier MUST have a unique constraint applied at the database layer.	MUST	Data Architecture and Naming Standard	3.6.3.2 – Surrogate Key Naming
DATA-0193	Develop architectural models that allow OHIP to assess a solution's compliance with OHIP business requirements, CMS/MITA/SMC requirements, and any new or complex integrations.	MUST	Data Architecture and Naming Standard	3.7.1 – Assessment
DATA-0194	Develop and maintain all architecture models (including conceptual data models) in OHIP's Enterprise Architecture Repository. Ensure all models are kept current.	MUST	Data Architecture and Naming Standard	3.7.2 – Modeling Tools
DATA-0195	Register all modeled and implemented APIs in OHIP's API catalog.	MUST	Data Architecture and Naming Standard	3.7.2 – Modeling Tools
DATA-0196	Capture required metadata elements (unique identifier, name, description, domain/category, operational support, system owner, core system, data sensitivity, criticality, disposition) for all systems, applications, and components.	MUST	Data Architecture and Naming Standard	3.7.2 – Modeling Tools
DATA-0197	Develop logical and physical data models in a format compatible with OHIP's designated data modelling tool.	MUST	Data Architecture and Naming Standard	3.7.2 – Modeling Tools

DATA-0198	Evaluate existing Authoritative Reference Models and incorporate where appropriate. Adopt prescribed models where federal funding mandates specific reference architectures. Align modeling efforts with MITA guidance.	MUST	Data Architecture and Naming Standard	3.7.3 – Reference Architecture Models
DATA-0199	Collaborate with OHIP to select a relevant maturity model and associated modeling patterns before beginning any modeling activity. Patterns MUST provide cloud-native and microservices guidance.	MUST	Data Architecture and Naming Standard	3.7.4 – Model Maturity and Patterns
DATA-0200	Apply consistent entity, attribute, component, and interface naming for all models in accordance with the Section 3.4 and applicable federal conventions.	MUST	Data Architecture and Naming Standard	3.7.4 – Model Maturity and Patterns
DATA-0201	Adhere to the MITA Maturity Model (MMM) and its associated capability matrices.	MUST	Data Architecture and Naming Standard	3.7.4 – Model Maturity and Patterns
DATA-0202	Conduct a formal review process for all models, including peer review by at least one architect or analyst, validation against use cases and source documentation, approval from business owners, storage with proper metadata, and validation against quality dimensions.	MUST	Data Architecture and Naming Standard	3.7.5 – Model Quality Standards
DATA-0203	Submit all models to the Architecture Review Board for review and approval.	MUST	Data Architecture and Naming Standard	3.7.5 – Model Quality Standards
DATA-0204	Develop an accompanying architecture description for each model, including stakeholders, concerns, rationale, correspondence, mapping between views, element descriptions, variability, constraints, and directory. Comply with ISO/IEC/IEEE 42010:2022 and MITA.	MUST	Data Architecture and Naming Standard	3.7.6 – Architecture Description
DATA-0205	Maintain all Enterprise Architecture in OHIP's Enterprise Architecture Repository.	MUST	Data Architecture and Naming Standard	3.7.7 – Architecture Repository and Catalog
DATA-0206	Document all services and architecture elements within the proper solution context. Not duplicate components unless explicitly justified. Extend solutions from prescribed reference models.	MUST	Data Architecture and Naming Standard	3.7.7 – Architecture Repository and Catalog
DATA-0207	Document rationale explaining why reuse was not possible for every new element. Ensure components are traceable to	MUST	Data Architecture and Naming Standard	3.7.7 – Architecture Repository and Catalog

	business drivers with no orphaned elements.			
DATA-0208	Verify that architecture metadata is complete, well-formed, and structured for discoverability and reusability.	MUST	Data Architecture and Naming Standard	3.7.7 – Architecture Repository and Catalog
DATA-0209	Minimize service sprawl through appropriate reuse. Synchronize all architectural views. Document exceptions with justification and OHIP approval.	MUST	Data Architecture and Naming Standard	3.7.7 Architecture Repository and Catalog
DATA-0210	Identify and contribute reusable components back to the enterprise knowledge base. Capture MITA-prescribed service metadata for the Business and Technical Service Definition Packages.	MUST	Data Architecture and Naming Standard	3.7.7 – Architecture Repository and Catalog
DATA-0211	Utilize MITA Part II Data Standards as the basis of data models and dictionaries before adding State customizations.	MUST	Data Architecture and Naming Standard	3.8 – Data Architecture Artifacts
DATA-0212	Cross-reference subject areas handled by each data model against the MITA Information Architecture and produce a mapping document. Ensure entity names are consistently utilized across the Enterprise.	MUST	Data Architecture and Naming Standard	3.8 – Data Architecture Artifacts
DATA-0213	Reverse engineer existing databases for applications without existing data models prior to any database updates or modifications.	MUST	Data Architecture and Naming Standard	3.8.1 – Data Model Completeness Requirements
DATA-0214	Include all tables in the model (including unused tables), subject areas (color-coded), all key constraints showing relationships, a single comprehensive diagram, other database objects if generating DDL, and use the model to generate data dictionary information.	MUST	Data Architecture and Naming Standard	3.8.1 – Data Model Completeness Requirements
DATA-0215	Produce a Conceptual Data Model covering all information in scope, with a one-page summary diagram, multiple detailed diagrams for large CDMs, entity-relationship or UML notation, and a Data Dictionary.	MUST	Data Architecture and Naming Standard	3.8.2 – Conceptual Data Model
DATA-0216	Produce a Logical Data Model with a Data Dictionary. Ensure the LDM is traceable, entities are grouped into CDM subject areas, each entity has a primary key, and all relationships follow the OHIP Database Design Standard.	MUST	Data Architecture and Naming Standard	3.8.3 – Logical Data Model
DATA-0217	Produce a Physical Data Model showing tables, fields, relationships, database	MUST	Data Architecture	3.8.4 – Physical Data Model

	descriptors, diagrams, data dictionary, and specifications. Include all pre-packaged database data elements.		and Naming Standard	
DATA-0218	Include required PDM data elements for custom-developed databases: table names, column details (name, data type, length, nullability, defaults), primary/foreign keys, indexes, views, stored procedures, triggers, sequences, partitioning, tablespace, and constraints.	MUST	Data Architecture and Naming Standard	3.8.4 – Physical Data Model
DATA-0219	Produce a Canonical Data Model defining a unified, Contractor-neutral representation of enterprise data. Ensure alignment with federal reference models and industry standards.	MUST	Data Architecture and Naming Standard	3.8.5 – Canonical Data Model
DATA-0220	Represent the Canonical Data Model in graphical form readable by humans and implemented in API clients of the interfaces.	MUST	Data Architecture and Naming Standard	3.8.5 – Canonical Data Model
DATA-0221	Include in each canonical entity: entity name, definition, source domain/system, attributes (name, data type, format, optionality, domain values), relationships, mapping to source/target systems, applicable data standards, and data sensitivity.	MUST	Data Architecture and Naming Standard	3.8.5 – Canonical Data Model
DATA-0222	Produce a Metadata Model covering structural, syntactic, semantic, integration, business, and operational metadata. Adhere to standards such as HL7/FHIR, DCAT, and ISO/IEC 11179.	MUST	Data Architecture and Naming Standard	3.8.6 – Metadata Model
DATA-0223	The Data Architecture and Naming Standard is a living artifact. The Contractor MUST support the annual review and maintenance of this standard by identifying proposed revisions based on evolving business needs, technology changes, and lessons learned during implementation.	MUST	Data Architecture and Naming Standard	3.9 – Maintaining the Data Naming Standard
DATA-0224	The Enterprise Glossary SHOULD be kept current and accessible with a defined process for approving, adding, updating, or deleting terms in a timely manner.	SHOULD	Data Architecture and Naming Standard	3.10.1 – Enterprise Glossary
DATA-0225	Automated data schema and modeling tools MAY be used to assist with and document data descriptions. Note: information extracted by these tools MAY lack meaningful content without proper enforcement of inline data documentation.	MAY	Data Architecture and Naming Standard	3.10.2 – Data Descriptions
DATA-0226	Entity definitions MUST begin with the entity name followed by 'is a' or 'is an',	MUST	Data Architecture	3.10.2 – Data Descriptions

	MUST define the entity as a thing not a process, and MUST NOT contain circular definitions or restate the entity name as its own explanation.		and Naming Standard	
DATA-0227	Attribute definitions MUST begin with the attribute name followed by 'is the', MUST define the attribute as a specific fact, and MUST include permissible values or value sets where the attribute is a coded element.	MUST	Data Architecture and Naming Standard	3.10.2 – Data Descriptions
DATA-0228	The Contractor MUST collaborate with OHIP to update the Enterprise Glossary, establishing a common vocabulary and ensuring consistent understanding of standard terminology.	MUST	Data Architecture and Naming Standard	3.10 – Related Data Maintenance Topics
DATA-0229	The Contractor MUST produce and submit all Deliverables in the format, structure, and cadence specified by OHIP, satisfying all content requirements and acceptance criteria.	MUST	Data Architecture and Naming Standard	Appendix A
DATA-0230	Upon notice of modifications to Deliverables and Artifacts requirements, the Contractor MUST comply with all modifications and implement them promptly.	MUST	Data Architecture and Naming Standard	Appendix A
DATA-0231	Organizations MAY use different job titles for the functional roles defined in this standard; however, each role defined MUST be assigned.	MAY	Database Design Standard	4.1 – Introduction
DATA-0232	The functional roles MUST be assigned to specific individuals or teams. The Contractor MUST collaborate with OHIP on database governance requirements.	MUST	Database Design Standard	4.1 – Introduction
DATA-0233	For database types not specifically addressed by OLTP or OLAP requirements, the Contractor MUST follow the MITA Information Architecture, an industry or Contractor design standard approved by the Department, and a Data Management Plan per Section 2.0.	MUST	Database Design Standard	4.2.1 – Database Type Applicability
DATA-0234	Additional database types (graph databases, document stores, vector databases, NoSQL) MAY come into scope. Where requirements in this standard do not specify OLTP or OLAP, they apply to all database types.	MAY	Database Design Standard	4.2.1 – Database Type Applicability
DATA-0235	The Contractor MUST demonstrate how the requirements of this standard will be met within their specific system and database technology context.	MUST	Database Design Standard	4.2.2 – Multi-System Design Applicability

DATA-0236	All database designs MUST have a clearly defined purpose matched to specific business needs. The Department reserves the right to require alterations in database implementations that do not align with design best practices.	MUST	Database Design Standard	4.4 – Database Implementation Standards
DATA-0237	The normalization or denormalization approach MUST be documented in the Data Management Plan and MUST follow an industry standard approved by the Department (e.g., DAMA DMBOK or Contractor-provided guidelines).	MUST	Database Design Standard	4.4.1 – Normalization and Schema Standards
DATA-0238	OLTP database schemas MUST adhere to normalization principles and MUST conform to at least Third Normal Form (3NF) to minimize redundancy, prevent update anomalies, and ensure referential integrity.	MUST	Database Design Standard	4.4.1.1 – OLTP Normalization Requirements
DATA-0239	For 3NF compliance, every table MUST have a clearly defined primary key; all non-key attributes MUST be fully functionally dependent on the entire primary key; the table MUST NOT contain transitive dependencies; all columns MUST contain atomic values; and the table MUST NOT contain repeating groups.	MUST	Database Design Standard	4.4.1.1 – OLTP Normalization Requirements
DATA-0240	Selective denormalization MAY be permitted for OLTP when justified by performance, scalability, or operational considerations. Any denormalization MUST be explicitly documented, reviewed, and approved by the Department, and MUST NOT compromise data accuracy, auditability, or maintainability.	MAY	Database Design Standard	4.4.1.1 – OLTP Normalization Requirements
DATA-0241	OLAP database schemas MUST be implemented using dimensional modeling to support efficient querying, aggregation, and analytical reporting.	MUST	Database Design Standard	4.4.1.2 – OLAP Dimensional Modeling Requirements
DATA-0242	OLAP databases MAY employ controlled denormalization through dimensional modeling techniques rather than fully normalized relational structures.	MAY	Database Design Standard	4.4.1.2 – OLAP Dimensional Modeling Requirements
DATA-0243	An OLAP star schema MUST meet: measurable data elements stored in a central fact table; descriptive data stored in separate dimension tables; dimension tables denormalized to reduce join complexity; each dimension related to the fact table through a single primary-foreign key relationship; and the schema MUST	MUST	Database Design Standard	4.4.1.2 – OLAP Dimensional Modeling Requirements

	support predictable query performance and drill-down/roll-up operations.			
DATA-0244	Dimensional models MUST align with established business processes and reporting requirements and MUST be validated to ensure consistency, accuracy, and scalability.	MUST	Database Design Standard	4.4.1.2 – OLAP Dimensional Modeling Requirements
DATA-0245	The Contractor MUST utilize database constraints in addition to primary and unique keys. Handling data quality controls exclusively in the application layer is NOT acceptable.	MUST	Database Design Standard	4.4.2 – Referential Integrity and Constraint Standards
DATA-0246	Referential integrity MUST be managed via inherent relational database capabilities and MUST be clearly documented in the data model via relationships and constraints.	MUST	Database Design Standard	4.4.2 – Referential Integrity and Constraint Standards
DATA-0247	Every table MUST have a natural key: the minimum set of business attributes that uniquely identifies a row without relying on system-generated values. The natural key MUST be documented in the data dictionary and enforced through a unique constraint at the database layer, even when a surrogate key serves as the primary key	MUST	Database Design Standard	4.4.2 – Referential Integrity and Constraint Standards
DATA-0248	Tables MAY use a surrogate key as the primary key. When a surrogate key is used, the natural key MUST still be identified, documented, and enforced with a unique constraint. Surrogate key naming MUST follow the conventions defined in Section 3.6.3.2.	MAY	Database Design Standard	4.4.2 – Referential Integrity and Constraint Standards
DATA-0249	The Contractor MAY define foreign key constraints for all parent-child relationships between tables. Foreign keys MUST enforce referential integrity at the database layer, not solely through application logic	MAY	Database Design Standard	4.4.2 – Referential Integrity and Constraint Standards
DATA-0250	Every table MUST have a unique primary key with a governed definition. Any table without a primary key MUST provide an explanation in the data dictionary and receive specific Department approval.	MUST	Database Design Standard	4.4.2 – Referential Integrity and Constraint Standards
DATA-0251	All primary keys MUST follow the naming convention defined in the OHIP Data Architecture and Naming Standard, including table name reference as opposed to simply 'ID'.	MUST	Database Design Standard	4.4.2 – Referential Integrity and Constraint Standards

DATA-0252	Foreign keys MUST use the same key name as the primary key in the referenced table to ensure clarity and traceability across the data model.	MUST	Database Design Standard	4.4.2 – Referential Integrity and Constraint Standards
DATA-0253	Designers MUST establish primary key–foreign key relations between main and secondary tables to prevent orphan records. Where such a relationship does not exist, reasons MUST be provided in the data dictionary and approved by the Department.	MUST	Database Design Standard	4.4.2 – Referential Integrity and Constraint Standards
DATA-0254	Designers MUST use default value constraints, check constraints, and unique, primary, and foreign key constraints at the database layer to ensure data accuracy. Developers and Data Architects MUST handle errors generated at the database layer in their code.	MUST	Database Design Standard	4.4.2 – Referential Integrity and Constraint Standards
DATA-0255	All database elements (e.g., tables, views, indexes, stored procedures, functions, triggers, constraints, sequences, identity columns, schemas, synonyms, materialized views, and partitions) MUST have usage, optimization, and maintenance standards documented in the Data Management Plan and approved by the Department.	MUST	Database Design Standard	4.4.3 – Database Elements and Partitioning
DATA-0256	Database partitioning strategies, when applicable, MUST be documented in the Data Management Plan with justification for the chosen approach.	MUST	Database Design Standard	4.4.3 – Database Elements and Partitioning
DATA-0257	Partitioning strategy MUST account for transactional consistency, cross-partition querying, query performance, and operational complexity. The selected approach MUST be approved by the Department before implementation.	MUST	Database Design Standard	4.4.3 – Database Elements and Partitioning
DATA-0258	The Contractor MUST establish database performance baselines, monitor query execution times, and track resource utilization trends. Performance degradation exceeding agreed thresholds MUST be reported and addressed per the OHIP incident management process.	MUST	Database Design Standard	4.4.3 – Database Elements and Partitioning
DATA-0259	Data in the database MUST be stored in a user-friendly, readable format where no application layer translation is required to understand or view the data.	MUST	Database Design Standard	4.4.4 – Data Storage Standards
DATA-0260	Special characters MUST be stored as they are expected to be presented to the user, not in ASCII format in the database	MUST	Database Design Standard	4.4.4 – Data Storage Standards

	(e.g., an asterisk MUST be stored as an asterisk character, not its ASCII equivalent).			
DATA-0261	All database object names, entity names, attribute names, and index names MUST comply with the OHIP Data Architecture and Naming Standard. Index names MUST use the IDX_ prefix and include table and column references (e.g., IDX_Customers_LastName).	MUST	Database Design Standard	4.4.4 – Data Storage Standards
DATA-0262	All database tables MUST include standard audit columns: created_date, created_by, last_updated_date, last_updated_by, and logical_delete_flag. Any exception MUST be documented in the data dictionary and approved by the Department.	MUST	Database Design Standard	4.4.5 – Audit Column Standards
DATA-0263	created_date and last_updated_date MUST be system-generated and not modifiable by application users. last_updated_date MUST be updated automatically on any record change.	MUST	Database Design Standard	4.4.5 – Audit Column Standards
DATA-0264	Physical deletion of records is PROHIBITED unless explicitly approved by the Department and documented per the data fix governance process in Section 4.5.4. The logical_delete_flag default value MUST be 'N' or FALSE.	MUST	Database Design Standard	4.4.5 – Audit Column Standards
DATA-0265	Audit column naming MUST comply with the OHIP Data Architecture and Naming Standard.	MUST	Database Design Standard	4.4.5 – Audit Column Standards
DATA-0266	Alternative audit column implementations (e.g., temporal tables, change data capture, or history tables) MAY be proposed, provided they are documented in the Data Management Plan, provide equivalent traceability, and are approved by the Department prior to implementation.	MAY	Database Design Standard	4.4.5 – Audit Column Standards
DATA-0267	Audit columns MUST NOT be included in application-facing data entry screens or made editable by end users. Audit column values MUST be populated and maintained exclusively by the database layer or application middleware.	MUST	Database Design Standard	4.4.5 – Audit Column Standards
DATA-0268	Every index MUST have a documented business or technical justification in the data dictionary. Indexes MUST NOT be created speculatively.	MUST	Database Design Standard	4.4.6 – Index Management Standards

DATA-0269	Each index MUST be associated with a specific query pattern, report requirement, or application function. Indexes created solely for exploratory or one-time queries MUST be removed after use.	MUST	Database Design Standard	4.4.6 – Index Management Standards
DATA-0270	All indexes MUST comply with the OHIP Data Architecture and Naming Standard using the IDX_ prefix and referencing the table and column(s) they support (e.g., IDX_Claims_ServiceDate).	MUST	Database Design Standard	4.4.6 – Index Management Standards
DATA-0271	The Contractor MUST monitor index usage on a regular cadence. Unused indexes (not referenced by any query workload for a documented period) MUST be flagged for review and removed following the change control process.	MUST	Database Design Standard	4.4.6 – Index Management Standards
DATA-0272	Index fragmentation MUST be monitored and addressed through rebuild or reorganization schedules documented in the Data Management Plan.	MUST	Database Design Standard	4.4.6 – Index Management Standards
DATA-0273	Removing an index MUST follow the same change control process as any DDL change (Section 4.5.3). Before removal, the Contractor MUST verify that no active queries depend on the index.	MUST	Database Design Standard	4.4.6 – Index Management Standards
DATA-0274	Composite indexes MUST document the column order rationale and the query patterns that justify the specific column sequence.	MUST	Database Design Standard	4.4.6 – Index Management Standards
DATA-0275	The Contractor MUST provide data model versioning. Prior to any release, the designer MUST create delta and named versions to track all changes. Versioning MUST include data model and data dictionary updates.	MUST	Database Design Standard	4.5.1 – Model and Database Versioning
DATA-0276	Database versions MUST be updated on a regular cadence. The Contractor MUST provide a regular schedule of updates to OHIP for every database in scope.	MUST	Database Design Standard	4.5.1 – Model and Database Versioning
DATA-0277	Tests MUST include any application that uses the databases being updated.	MUST	Database Design Standard	4.5.1 – Model and Database Versioning
DATA-0278	Updated data dictionaries and data models (in the Department's preferred modeling tool) MUST be provided for review and validation with each major application release.	MUST	Database Design Standard	4.5.1 – Model and Database Versioning
DATA-0279	Database designers MUST identify tables or objects no longer in active use. These MUST be marked as obsolete in the data	MUST	Database Design Standard	4.5.2 – Database Maintenance and Cleanup

	model with a date and reason, and the Data Steward MUST be notified.			
DATA-0280	Legacy tables or objects causing unnecessary storage use MUST be removed. Data MUST be archived before removal per Section 2.0 retention requirements.	MUST	Database Design Standard	4.5.2 – Database Maintenance and Cleanup
DATA-0281	Obsolete tables and objects MUST be removed within two (2) major revisions. An obsolete object is one that is not referenced by any other object or process.	MUST	Database Design Standard	4.5.2 – Database Maintenance and Cleanup
DATA-0282	Temporary tables MUST be removed regularly as part of the process that created them. Any table without an identified owner MUST be flagged for review within the next release cycle.	MUST	Database Design Standard	4.5.2 – Database Maintenance and Cleanup
DATA-0283	The data dictionary MUST reflect all data elements including those marked as obsolete. The data dictionary MUST be updated to reflect the deletion of objects before removal.	MUST	Database Design Standard	4.5.2 – Database Maintenance and Cleanup
DATA-0284	The Contractor MUST ensure database designs accommodate retention-based partitioning, archival extraction, and data purge processes required by Section 2.0.	MUST	Database Design Standard	4.5.2 – Database Maintenance and Cleanup
DATA-0285	All database models and structural changes MUST undergo review and approval by OHIP within the general change management framework defined in the OHIP IT Service Management (ITSM) Standard.	MUST	Database Design Standard	4.5.3 – Database Change Control
DATA-0286	Database structure or design changes MUST be updated in line with application changes and MUST be given the same level of attention as application development in terms of planning, testing, and release management.	MUST	Database Design Standard	4.5.3 – Database Change Control
DATA-0287	An updated data dictionary and data model MUST be provided for review and validation with each major application release involving database changes.	MUST	Database Design Standard	4.5.3 – Database Change Control
DATA-0288	All DDL changes MUST be approved by a Data Architect. Data Architects MUST ensure proposed designs do not create duplicate or conflicting data in the overall data model.	MUST	Database Design Standard	4.5.3 – Database Change Control
DATA-0289	Database structural changes MUST be documented in separate tickets from application changes in the Department's designated issue tracking system.	MUST	Database Design Standard	4.5.3 – Database Change Control

DATA-0290	The Contractor MUST provide reports on the volume of database change requests to the Department on a cadence defined in the applicable contract.	MUST	Database Design Standard	4.5.3 – Database Change Control
DATA-0291	DDL scripts MUST NOT be run manually. DDL MUST be implemented via automated processes for promotion across environments.	MUST	Database Design Standard	4.5.3 – Database Change Control
DATA-0292	Errors from executing DDL MUST be reported and addressed prior to production deployment.	MUST	Database Design Standard	4.5.3 – Database Change Control
DATA-0293	Schema changes SHOULD be non-breaking where possible; existing consumers SHOULD continue to function without modification after the change is applied.	SHOULD	Database Design Standard	4.5.3.1 – Backward Compatibility and Schema Migration
DATA-0294	Non-breaking changes still MUST require Data Architect approval and change control documentation per Section 4.5.3.	MUST	Database Design Standard	4.5.3.1 – Backward Compatibility and Schema Migration
DATA-0295	Breaking changes MUST have a documented impact analysis before approval. The impact analysis MUST identify all downstream consumers affected by the change.	MUST	Database Design Standard	4.5.3.1 – Backward Compatibility and Schema Migration
DATA-0296	Deprecation protocol: the object MUST be marked as deprecated in the data dictionary; all known consumers MUST be notified with a defined transition window; and the deprecated object MUST remain functional until the transition window expires.	MUST	Database Design Standard	4.5.3.1 – Backward Compatibility and Schema Migration
DATA-0297	All schema migrations MUST be implemented through versioned, idempotent migration scripts. Scripts MUST be stored in version control and MUST be executable repeatedly without error or unintended side effects.	MUST	Database Design Standard	4.5.3.1 – Backward Compatibility and Schema Migration
DATA-0298	Rollback procedures MUST be tested in a non-production environment before the migration is approved for production.	MUST	Database Design Standard	4.5.3.1 – Backward Compatibility and Schema Migration
DATA-0299	All database changes MUST be tested per the OHIP Test Management Standard, including DDL validation, referential integrity verification, and application integration testing.	MUST	Database Design Standard	4.5.3.1 – Backward Compatibility and Schema Migration
DATA-0300	Data fixes MUST NOT be regular practice. Data fixes MUST only occur when critical issues in the application or interface	MUST	Database Design Standard	4.5.4 – Data Fix Governance

	cannot be resolved in a timely manner through normal application processes.			
DATA-0301	Data MUST only be updated through the application GUI or automated application processes. Direct data manipulation via scripts is PROHIBITED except in approved data fix scenarios.	MUST	Database Design Standard	4.5.4 – Data Fix Governance
DATA-0302	Any situation requiring a data fix MUST generate a problem ticket and MUST lead to changes in application logic or interface design to prevent recurrence.	MUST	Database Design Standard	4.5.4 – Data Fix Governance
DATA-0303	The data fix MUST track the number of records it adjusts via a logging mechanism.	MUST	Database Design Standard	4.5.4 – Data Fix Governance
DATA-0304	The fix SHOULD be executed in a non-production environment with live data if possible, to ensure the expected impact on production is known before execution.	SHOULD	Database Design Standard	4.5.4 – Data Fix Governance
DATA-0305	The fix MUST be thoroughly validated by business users and approved before running in production to mitigate unexpected impacts.	MUST	Database Design Standard	4.5.4 – Data Fix Governance
DATA-0306	All data fixes MUST be documented, including business justification, the records affected, testing results, and business user sign-off.	MUST	Database Design Standard	4.5.4 – Data Fix Governance
DATA-0307	Recurring data fixes addressing the same root cause MUST trigger an application defect ticket with a required remediation timeline.	MUST	Database Design Standard	4.5.4 – Data Fix Governance
DATA-0308	Database schemas MUST be consistent across all environments (development, test, staging, production). Schema drift, where lower environments diverge from production, is PROHIBITED.	MUST	Database Design Standard	4.5.5 – Database Environment Management
DATA-0309	The Contractor MUST implement automated schema comparison tools or processes to detect and remediate schema drift on a scheduled basis.	MUST	Database Design Standard	4.5.5 – Database Environment Management
DATA-0310	Production data containing PHI, PII, FTI, or MCD MUST NOT be copied to non-production environments without masking or de-identification approved by the Department and the OHIP Privacy Officer.	MUST	Database Design Standard	4.5.5 – Database Environment Management
DATA-0311	Any exception requiring testing with unmasked live data MUST be approved by the Department and the OHIP Privacy and Security Officers with documented justification and access controls.	MUST	Database Design Standard	4.5.5 – Database Environment Management

DATA-0312	The Contractor MUST document and maintain a schedule for refreshing lower environments from production (with appropriate masking). Refresh schedules MUST be approved by the Department.	MUST	Database Design Standard	4.5.5 – Database Environment Management
DATA-0313	After each environment refresh, the Contractor MUST verify that all application-specific configurations and connection strings are correctly updated for the non-production environment.	MUST	Database Design Standard	4.5.5 – Database Environment Management
DATA-0314	Developers MAY have elevated object-level permissions (e.g., CREATE, ALTER) in development environments only. Direct DDL execution by developers in production is PROHIBITED.	MAY	Database Design Standard	4.5.5 – Database Environment Management
DATA-0315	Direct DDL execution by developers in production is PROHIBITED.	MUST	Database Design Standard	4.5.5 – Database Environment Management
DATA-0316	All database activities MUST comply with the OHIP Security and Privacy Compliance Standard, including IAM integration, role-based access, encryption at rest and in transit, and audit logging.	MUST	Database Design Standard	4.6 – Database Security
DATA-0317	The Contractor MUST ensure database security implementations are coordinated with the Department's enterprise Identity Access Management (IAM) solution. After IAM integration, local database accounts MUST NOT be used.	MUST	Database Design Standard	4.6 – Database Security
DATA-0318	After initial database deployments and IAM integration, local database accounts MUST NOT be used.	MUST	Database Design Standard	4.6 – Database Security
DATA-0319	Compliance with this standard is REQUIRED throughout the term of the awarded contract. Compliance with amended policies and standards is required within the timeframe specified in the amendment notice, or immediately if no timeframe is specified.	MUST	Database Design Standard	1.7 – Compliance Tier Framework
DATA-0320	For existing databases, the Contractor MUST develop a compliance roadmap to bring current implementations into alignment with this standard. The roadmap MUST be submitted to OHIP within sixty (60) days of contract award.	MUST	Database Design Standard	1.8 – Compliance and Enforcement
DATA-0321	The compliance roadmap MUST prioritize referential integrity enforcement, data dictionary currency, audit column implementation, and elimination of direct DDL execution in production.	MUST	Database Design Standard	1.8 – Compliance and Enforcement

DATA-0322	The Contractor MUST provide Data Fix Documentation recording all data fix events, including business justification, records affected, testing results, approval sign-off, and root cause analysis.	MUST	Database Design Standard	Appendix A
DATA-0323	The Contractor MUST provide a Database Change Request Log recording all database change requests, including request date, description, impact analysis, approval status, and implementation date.	MUST	Database Design Standard	Appendix A
DATA-0324	The Contractor MUST provide a Database Version History Document recording all database versions (delta and named), changes per version, and the data dictionary state at each version.	MUST	Database Design Standard	Appendix A
DATA-0325	The Contractor MUST produce and submit all Deliverables in the format, structure, and cadence specified by OHIP. Each Deliverable MUST satisfy all content requirements and is subject to formal OHIP review and approval.	MUST	Database Design Standard	Appendix A
DATA-0326	Upon notice of modifications to Deliverables and Artifacts requirements, the Contractor MUST comply with all modifications and implement them promptly.	MUST	Database Design Standard	Appendix A
DATA-0327	Organizations MAY use different job titles for the functional roles defined in this Standard; however, each role MUST be assigned to specific individuals or teams to ensure clear accountability throughout the conversion lifecycle.	MAY	Data Conversion Standard	5.1 – Introduction
DATA-0328	Real-time transfer MUST be used where source and target solutions support real-time or near real-time data exchange to minimize data lag and ensure operational continuity.	MUST	Data Conversion Standard	5.4 – Guiding Principles
DATA-0329	Scheduled exchanges MAY be used only for legacy system decommissioning or when the target system cannot support real-time transfer, with documented justification approved by OHIP.	MAY	Data Conversion Standard	5.4 – Guiding Principles
DATA-0330	All data exchanges and conversions MUST comply with relevant federal standards (CMS MITA, CEF, ARC-A, FHIR, and X12) and applicable NYS standards.	MUST	Data Conversion Standard	5.4 – Guiding Principles
DATA-0331	Conversion activities MUST maintain or improve data quality. Source data MUST be profiled and cleansed before	MUST	Data Conversion Standard	5.4 – Guiding Principles

	conversion begins, and quality outcomes MUST be measurably verified against thresholds approved by OHIP.			
DATA-0332	Data Owners MUST approve all conversion decisions including source-to-target mappings, cleansing rules, default value strategies, and exception handling approaches before development begins.	MUST	Data Conversion Standard	5.4 – Guiding Principles
DATA-0333	Conversion MUST use the appropriate data exchange standard for each transaction type (FHIR for clinical data, X12 for administrative transactions, NIEM for cross-agency information exchange, etc.).	MUST	Data Conversion Standard	5.4 – Guiding Principles
DATA-0334	Converted data MUST adhere to the OHIP Data Conversion Standard and all OHIP data governance requirements, including naming conventions, data dictionary registration, and metadata completeness.	MUST	Data Conversion Standard	5.4 – Guiding Principles
DATA-0335	All conversion activities MUST maintain full traceability from source through transformations to target. Every transformation step MUST be documented in a source-to-target mapping specification.	MUST	Data Conversion Standard	5.4 – Guiding Principles
DATA-0336	100% of target records MUST carry audit metadata: source system, extract timestamp, load timestamp, and transformation version.	MUST	Data Conversion Standard	5.4 – Guiding Principles
DATA-0337	Column-level lineage MUST be queryable for all converted attributes.	MUST	Data Conversion Standard	5.4 – Guiding Principles
DATA-0338	The Contractor MUST be able to reproduce any conversion run with its exact rule set, code baseline, and source data snapshot. Reproducibility MUST be demonstrated during dry runs.	MUST	Data Conversion Standard	5.4 – Guiding Principles
DATA-0339	At OHIP's request, all conversion activities, data, environments, artifacts, and materials MUST be provided and accessible to authorized OHIP staff and auditors.	MUST	Data Conversion Standard	5.4 – Guiding Principles
DATA-0340	The Contractor MUST profile 100% of in-scope source data at the attribute level before conversion design or development begins.	MUST	Data Conversion Standard	5.5.1 – Data Profiling
DATA-0341	The profiling assessment MUST include Statistical Profiling, Completeness Analysis, Uniqueness Analysis, Pattern and Format Analysis, Referential Integrity	MUST	Data Conversion Standard	5.5.1 – Data Profiling

	Analysis, and Temporal Consistency Analysis.			
DATA-0342	Attributes below 95% completeness MUST be flagged for remediation or documented exception before conversion design proceeds.	MUST	Data Conversion Standard	5.5.1 – Data Profiling
DATA-0343	The Contractor MUST produce a Data Quality Assessment Report with severity-ranked issues, heatmaps by domain, and remediation assignments. Baseline data quality scores MUST be published before conversion design begins.	MUST	Data Conversion Standard	5.5.1 – Data Profiling
DATA-0344	No conversion design or development MAY begin for a domain without an approved Corrective Action Plan (CAP) for all Severity 1 and Severity 2 profiling findings.	MUST	Data Conversion Standard	5.5.1 – Data Profiling
DATA-0345	The Contractor MUST develop and execute a Data Cleansing Strategy defining how source data quality issues will be remediated before or during conversion.	MUST	Data Conversion Standard	5.5.2 – Data Cleansing Strategy
DATA-0346	Each identified quality issue MUST be categorized as: (a) MUST -fix at source before extraction, (b) MUST -fix via ETL transformation, or (c) acceptable with documented exception approved by the Data Owner.	MUST	Data Conversion Standard	5.5.2 – Data Cleansing Strategy
DATA-0347	The Contractor MUST define match rules with confidence thresholds for deduplication. Survivorship rules MUST be documented and approved by the Data Owner before execution.	MUST	Data Conversion Standard	5.5.2 – Data Cleansing Strategy
DATA-0348	For issues not resolvable programmatically, a ticketing-based workflow SHOULD be established with owners and resolution deadlines prior to conversion execution.	SHOULD	Data Conversion Standard	5.5.2 – Data Cleansing Strategy
DATA-0349	Data Stewards MUST approve all manual remediation actions before they are executed.	MUST	Data Conversion Standard	5.5.2 – Data Cleansing Strategy
DATA-0350	The Contractor MUST publish quality scorecards per domain showing pre-cleansing baseline, post-cleansing improvement, and remaining exceptions. Scorecards MUST be provided to OHIP before each conversion phase milestone.	MUST	Data Conversion Standard	5.5.2 – Data Cleansing Strategy
DATA-0351	Every reference data domain MUST have a documented crosswalk mapping legacy	MUST	Data Conversion Standard	5.5.3 – Reference and Master Data Mapping

	to target values, versioned and linked to the source-to-target mapping specification.			
DATA-0352	Where applicable, the Contractor SHOULD define the approach for creating golden records, including survivorship rules and source system priority rankings, for master data entities.	SHOULD	Data Conversion Standard	5.5.3 – Reference and Master Data Mapping
DATA-0353	Reference data changes MUST be governed with versioning and effective dating. The conversion MUST correctly handle point-in-time lookups where business events depend on reference data values that were effective at a specific historical date.	MUST	Data Conversion Standard	5.5.3 – Reference and Master Data Mapping
DATA-0354	The Contractor MUST produce comprehensive source-to-target mapping documentation before conversion development begins. OHIP sign-off is REQUIRED before development starts.	MUST	Data Conversion Standard	5.5.4 – Source-to-Target Mapping
DATA-0355	100% of target attributes MUST be mapped with business rules, transformation logic, defaulting rules, survivorship rules, and reference data crosswalks. Unmapped target attributes are NOT permitted without documented exception approved by OHIP.	MUST	Data Conversion Standard	5.5.4 – Source-to-Target Mapping
DATA-0356	100% of mappings MUST include full source-to-target lineage: source table/column, transformation version, business rule description, and target table/column.	MUST	Data Conversion Standard	5.5.4 – Source-to-Target Mapping
DATA-0357	OHIP sign-off is REQUIRED on mapping specifications before development begins. Post-sign-off changes to mappings MUST go through a formal change control process with OHIP re-approval.	MUST	Data Conversion Standard	5.5.4 – Source-to-Target Mapping
DATA-0358	Each transformation rule MUST include explicit handling for nulls, truncation, time zone conversions, encoding differences, and other edge cases relevant to the data domain.	MUST	Data Conversion Standard	5.5.4 – Source-to-Target Mapping
DATA-0359	Each transformation rule MUST be tagged with a version number and effective date. A change history MUST be maintained for all mapping changes made after initial OHIP approval.	MUST	Data Conversion Standard	5.5.4 – Source-to-Target Mapping
DATA-0360	The Contractor MUST develop a Data Conversion and Migration Plan and obtain OHIP approval before work begins. The plan MUST cover all phases, milestones,	MUST	Data Conversion Standard	5.5.5 – Data Conversion and Migration Plan

	resources, risk mitigation, and rollback procedures.			
DATA-0361	Conversions SHOULD leverage SaaS or COTS tools implementing federal standards such as HL7/FHIR where available and applicable, to reduce custom development and improve standards compliance.	SHOULD	Data Conversion Standard	5.5.6 – SaaS/COTS for Data Exchange
DATA-0362	Contractors SHOULD examine every avenue for using supported products before proposing custom-built conversion components. Custom development MUST be justified with documented evidence that no suitable standard product exists.	SHOULD	Data Conversion Standard	5.5.6 – SaaS/COTS for Data Exchange
DATA-0363	All Pre-Conversion activities MUST be completed and documented per the approved Data Conversion and Migration Plan before Conversion Phase activities begin.	MUST	Data Conversion Standard	5.6.1 – Conversion Phases
DATA-0364	Any deviation from the approved plan MUST be documented, justified, and approved by OHIP before proceeding to the next phase.	MUST	Data Conversion Standard	5.6.1 – Conversion Phases
DATA-0365	All transformation and load processes MUST be rerunnable (idempotent) without side effects. Identical input MUST always produce identical output regardless of how many times the process is executed.	MUST	Data Conversion Standard	5.6.2 – Transformation Standards
DATA-0366	All loads MUST support restart without data loss or duplication. Proof of restartability MUST be provided and verified during dry runs.	MUST	Data Conversion Standard	5.6.2 – Transformation Standards
DATA-0367	Non-deterministic functions MUST be avoided in conversion transformation logic. Where technically unavoidable, the non-deterministic behavior MUST be documented, bounded, and approved by OHIP.	MUST	Data Conversion Standard	5.6.2 – Transformation Standards
DATA-0368	Conversion jobs SHOULD be modular with parameterized connectors. Retry and backoff logic SHOULD be implemented for all external system calls.	SHOULD	Data Conversion Standard	5.6.2 – Transformation Standards
DATA-0369	Every conversion job MUST log: job ID, run ID, start/end timestamps, record counts by entity, reject reasons and counts, and exception details. Logs MUST be retained per OHIP records retention requirements.	MUST	Data Conversion Standard	5.6.2 – Transformation Standards

DATA-0370	Every data conversion MUST have measurable acceptance criteria defined and approved by OHIP before conversion development begins. Acceptance criteria MUST be traceable to specific business requirements.	MUST	Data Conversion Standard	5.6.3 – Data Acceptance Criteria
DATA-0371	Completeness thresholds: mandatory fields MUST be 100% non-null; conditionally required fields MUST be ≥99.5% non-null when the condition is met; optional fields MUST be ≥90% non-null where data is expected.	MUST	Data Conversion Standard	5.6.3 – Data Acceptance Criteria
DATA-0372	Accuracy thresholds: critical elements MUST achieve ≥99.90% accuracy match; important elements MUST achieve ≥99.50% match against authoritative source values.	MUST	Data Conversion Standard	5.6.3 – Data Acceptance Criteria
DATA-0373	Conformance thresholds: critical elements MUST be 100% conformant to data type, format, domain values, and code sets; important elements MUST achieve ≥99.50% conformance.	MUST	Data Conversion Standard	5.6.3 – Data Acceptance Criteria
DATA-0374	Uniqueness thresholds: duplicate rate MUST be ≤0.05% for critical entities and ≤0.10% for all other entities.	MUST	Data Conversion Standard	5.6.3 – Data Acceptance Criteria
DATA-0375	Referential integrity: parent-child referential integrity MUST be 100% enforced with a 0% orphan record rate.	MUST	Data Conversion Standard	5.6.3 – Data Acceptance Criteria
DATA-0376	Code/lookup validity: ≥99.90% for critical entities; ≥99.50% for all others.	MUST	Data Conversion Standard	5.6.3 – Data Acceptance Criteria
DATA-0377	Record counts MUST match 100% between source extract and target load. Financial balances MUST differ by ≤0.001% from source totals.	MUST	Data Conversion Standard	5.6.3 – Data Acceptance Criteria
DATA-0378	100% of rejects MUST be routed to an exception repository. Zero silent failures are permitted: any job that drops records without logging them is a critical defect.	MUST	Data Conversion Standard	5.6.3 – Data Acceptance Criteria
DATA-0379	All production conversion jobs MUST complete within the approved cutover window defined in the Cutover Runbook.	MUST	Data Conversion Standard	5.6.3 – Data Acceptance Criteria
DATA-0380	100% of target records MUST carry audit metadata. Column-level lineage MUST be captured and queryable for all converted attributes.	MUST	Data Conversion Standard	5.6.3 – Data Acceptance Criteria
DATA-0381	Record counts MUST be verified by entity and key subgroups at each conversion stage (extract, transform, load). Any	MUST	Data Conversion Standard	5.6.4 – Reconciliation and Balancing

	variance MUST be immediately investigated and resolved before proceeding.			
DATA-0382	Numeric sums (e.g., financial balances, benefit amounts) MUST be calculated at source and verified at target.	MUST	Data Conversion Standard	5.6.4 – Reconciliation and Balancing
DATA-0383	Hash totals MAY be used for non-financial attributes where appropriate to supplement numeric sum controls.	MAY	Data Conversion Standard	5.6.4 – Reconciliation and Balancing
DATA-0384	Statistically valid samples per domain SHOULD be selected for manual verification to complement automated reconciliation checks.	SHOULD	Data Conversion Standard	5.6.4 – Reconciliation and Balancing
DATA-0385	Automated reconciliation reports MUST be produced for each conversion lot, including source counts, target counts, variance analysis, and exception summaries.	MUST	Data Conversion Standard	5.6.4 – Reconciliation and Balancing
DATA-0386	Data Owners MUST validate and approve reconciliation results for each conversion lot before the lot is considered complete. Sign-off MUST be documented in writing.	MUST	Data Conversion Standard	5.6.4 – Reconciliation and Balancing
DATA-0387	OHIP MAY require additional reconciliation controls beyond the minimums in Section 5.6.4 based on the risk profile of the specific conversion.	MAY	Data Conversion Standard	5.6.4 – Reconciliation and Balancing
DATA-0388	The Contractor MUST perform a minimum of two (2) successful dry runs for standard conversions; three (3) for high-volume or high-complexity conversions. Each dry run MUST use a full production-representative dataset.	MUST	Data Conversion Standard	5.7.1 – Conversion Trial Runs
DATA-0389	OHIP MUST approve the dry run plan before the first dry run is executed.	MUST	Data Conversion Standard	5.7.1 – Conversion Trial Runs
DATA-0390	Per dry run exit criteria: all Section 5.6.3 thresholds MUST be met; zero Severity 1 defects MAY remain unresolved; timing results MUST demonstrate the conversion can complete within the approved cutover window.	MUST	Data Conversion Standard	5.7.1 – Conversion Trial Runs
DATA-0391	The Contractor MUST maintain a data validation checklist covering all Section 5.6.3 criteria. Only data that passes all validation checks MAY proceed to the next conversion phase.	MUST	Data Conversion Standard	5.7.3 – Data Validation Process
DATA-0392	Converted data MUST be validated through all downstream endpoints: reports, screens, interfaces, and analytical queries	MUST	Data Conversion Standard	5.7.4 – End-to-End Verification

	that consume the converted data in the target system.			
DATA-0393	End-to-end verification MUST include both technical staff and end-users confirming converted data supports business workflows correctly in the target environment.	MUST	Data Conversion Standard	5.7.4 – End-to-End Verification
DATA-0394	The Contractor SHOULD use automated tools, when available, to check compliance with applicable standards (FHIR, X12, NIEM) at the field and transaction level during conversion testing.	SHOULD	Data Conversion Standard	5.7.5 – Automated Compliance Checking
DATA-0395	CI/CD deployment pipelines SHOULD be used for conversion deployments. Automated tools SHOULD be used for configuration drift detection and environment parity validation.	SHOULD	Data Conversion Standard	5.7.5 – Automated Compliance Checking
DATA-0396	All records loaded through conversion MUST be identifiable as converted data via a tagging or flagging mechanism that persists in the target system.	MUST	Data Conversion Standard	5.7.6 – Converted Data Identification and Integrated Validation
DATA-0397	The tagging approach MUST be approved by OHIP prior to the first dry run and MUST persist in the target system for the duration defined in the Data Conversion and Migration Plan.	MUST	Data Conversion Standard	5.7.6 – Converted Data Identification and Integrated Validation
DATA-0398	The Contractor MUST develop a Deployment Plan encompassing the full cutover strategy including communication plan, go/no-go decision criteria, rollback triggers, and stakeholder responsibilities.	MUST	Data Conversion Standard	5.8.1 – Cutover
DATA-0399	The Contractor MUST provide a detailed Cutover Runbook with step-by-step tasks, owners, checkpoints, timing, and escalation procedures. The Runbook MUST be approved by OHIP before production cutover is initiated.	MUST	Data Conversion Standard	5.8.1 – Cutover
DATA-0400	Before production cutover: all Section 5.6.3 thresholds MUST be met; no Severity 1 defects MAY remain unresolved; all dry run exit criteria MUST be satisfied; and OHIP sign-off MUST be obtained.	MUST	Data Conversion Standard	5.8.1 – Cutover
DATA-0401	The Contractor MUST develop and test a Rollback Plan including rollback triggers, pre/post-conversion backups, tested restore procedures, and bridge reactivation procedures for phased deployments.	MUST	Data Conversion Standard	5.8.2 – Rollback and Contingency

DATA-0402	Recovery Time Objective (RTO) and Recovery Point Objective (RPO) MUST be defined and tested prior to production cutover (e.g., RTO ≤4 hours; RPO = 0 during the conversion window).	MUST	Data Conversion Standard	5.8.2 – Rollback and Contingency
DATA-0403	If rollback is not viable, a fall-forward plan with compensating controls and a remediation timeline MAY be defined as an alternative, subject to OHIP approval.	MAY	Data Conversion Standard	5.8.2 – Rollback and Contingency
DATA-0404	The Contractor MUST conduct daily data quality and operational health checks during the initial Hypercare period as defined in the applicable contract or SOW.	MUST	Data Conversion Standard	5.8.3 – Hypercare and Post-Go-Live Stabilization
DATA-0405	After the initial Hypercare period, the Contractor MAY transition to a reduced monitoring cadence, subject to OHIP approval and confirmation that quality thresholds are being maintained.	MAY	Data Conversion Standard	5.8.3 – Hypercare and Post-Go-Live Stabilization
DATA-0406	The Contractor MUST develop data quality monitoring rules including dashboards, alerts, and automated data quality checks to monitor converted data in the production environment.	MUST	Data Conversion Standard	5.8.3 – Hypercare and Post-Go-Live Stabilization
DATA-0407	Alerts MUST be configured for quality threshold breaches, job failures, and performance degradation, and MUST be integrated with the MES Incident Management System.	MUST	Data Conversion Standard	5.8.3 – Hypercare and Post-Go-Live Stabilization
DATA-0408	The Contractor MUST correct data errors introduced by the conversion process at no additional cost during the Hypercare period.	MUST	Data Conversion Standard	5.8.3 – Hypercare and Post-Go-Live Stabilization
DATA-0409	Weekly Hypercare status reports MUST be provided to OHIP including data quality metrics, defect counts by severity, resolution status, and outstanding risks.	MUST	Data Conversion Standard	5.8.3 – Hypercare and Post-Go-Live Stabilization
DATA-0410	The Contractor MUST complete knowledge transfer and deliver operational runbooks to OHIP prior to go-live. Knowledge transfer sessions MUST be completed with attendance confirmation before Hypercare begins.	MUST	Data Conversion Standard	5.8.3 – Hypercare and Post-Go-Live Stabilization
DATA-0411	Production Standard Operating Procedures (SOPs), on-call schedules, alert thresholds, and escalation paths MUST be in place before cutover is initiated.	MUST	Data Conversion Standard	5.8.3 – Hypercare and Post-Go-Live Stabilization

DATA-0412	The Contractor MUST keep the legacy solution accessible for the period defined by the Department following production cutover to support audit, inquiry, and dispute resolution requirements.	MUST	Data Conversion Standard	5.8.4 – Post-Conversion Data Retention
DATA-0413	If maintaining the legacy solution is impractical, the legacy data MAY be exported to an archival format approved by OHIP, provided the archive supports the same access and query capabilities required for audit purposes.	MAY	Data Conversion Standard	5.8.4 – Post-Conversion Data Retention
DATA-0414	The retention period for legacy data MUST be documented in the Data Conversion and Migration Plan and approved by OHIP before cutover.	MUST	Data Conversion Standard	5.8.4 – Post-Conversion Data Retention
DATA-0415	Temporary staging areas and intermediate conversion files MUST be purged within OHIP-approved timeframes unless retention is required for audit or reproducibility purposes.	MUST	Data Conversion Standard	5.8.4 – Post-Conversion Data Retention
DATA-0416	The Contractor MUST implement detailed audit logging documenting: operator identity, timestamps, affected data domains, transformation versions applied, and exception counts for every conversion run.	MUST	Data Conversion Standard	5.9.1 – Conversion Audit Trails
DATA-0417	Audit logs MUST be protected against unauthorized modification and retained per OHIP records retention requirements and applicable NYS Records Retention Schedules.	MUST	Data Conversion Standard	5.9.1 – Conversion Audit Trails
DATA-0418	The Contractor MUST provide the following conversion deliverables: Data Quality Reports, Issues Log, Reconciliation Reports, and a Final Conversion Summary Report signed off by the Data Owner.	MUST	Data Conversion Standard	5.9.2 – Reporting Requirements
DATA-0419	The Contractor MUST provide a Post-Production Compliance Dashboard flagging interface outages and quality threshold breaches for the duration of the Hypercare period.	MUST	Data Conversion Standard	5.9.2 – Reporting Requirements
DATA-0420	All vendor-provided conversion tools MUST be available and licensed to OHIP staff. The Contractor MUST provide full access to conversion code, scripts, configurations, and documentation to OHIP upon request.	MUST	Data Conversion Standard	5.9.3 – Conversion Transparency
DATA-0421	The Contractor MUST maintain documentation sufficient for CMS SMC certification reviews, including conversion	MUST	Data Conversion Standard	5.9.4 – Compliance Audits

	methodology documentation, data quality evidence, reconciliation records, and audit logs for all conversion activities.			
DATA-0422	All data conversion activities MUST comply with the OHIP Security and Privacy Compliance Standard, including all applicable PHI, PII, FTI, and MCD handling requirements.	MUST	Data Conversion Standard	5.10 – Security and Privacy
DATA-0423	The Contractor MUST convert and exchange only the minimum necessary data required for the specific business purpose. Data minimization MUST be applied at the extraction stage.	MUST	Data Conversion Standard	5.10 – Security and Privacy
DATA-0424	Non-production environments MUST have 100% masking or de-identification of PII, PHI, PCI, FTI, and MCD data before any data is loaded into lower environments.	MUST	Data Conversion Standard	5.10 – Security and Privacy
DATA-0425	Intermediate files and staging databases SHOULD be secured using equivalent or risk-appropriate controls commensurate with data sensitivity	SHOULD	Data Conversion Standard	5.10 – Security and Privacy
DATA-0426	When feasible and appropriate, PHI SHOULD be de-identified or anonymized per HIPAA Safe Harbor or Expert Determination standards for development and testing use cases.	SHOULD	Data Conversion Standard	5.10 – Security and Privacy
DATA-0427	Compliance with this standard is required upon publication. Compliance with the version of this standard provided at the time of RFP response is required throughout the contract term.	MUST	Data Conversion Standard	1.7 – Compliance Tier Framework
DATA-0428	Best effort SHOULD be made to comply with all amendments to this standard issued after bid submission. Where full compliance is not immediately achievable, the Contractor MUST notify OHIP and propose a compliance timeline.	SHOULD	Data Conversion Standard	1.8 – Compliance and Enforcement
DATA-0429	The Contractor MUST produce and submit all Deliverables and Artifacts defined in Appendix A (including the Data Conversion and Migration Plan, Data Quality Assessment Report, Reconciliation Reports, Final Conversion Report, and an Unconverted Data Register) in the format, structure, and cadence specified by OHIP.	MUST	Data Conversion Standard	Appendix A
DATA-0430	Upon notice of modifications to Deliverables and Artifacts requirements, the Contractor MUST comply with all modifications and implement them promptly.	MUST	Data Conversion Standard	Appendix A

DATA-0431	The Contractor MUST adhere to these architectural patterns for any solution that ingests, processes, stores, or delivers OHIP Medicaid data within the Lakehouse environment.	MUST	Data Lakehouse Architecture Standard	6.1 – Introduction
DATA-0432	The Contractor MUST implement all standard governance requirements: data ingestion pipelines, tiered quality gates, data lineage tracking, and data catalog registration for all datasets.	MUST	Data Lakehouse Architecture Standard	6.1 – Introduction
DATA-0433	The Contractor MAY use different job titles but MUST assign the functional roles defined in this standard to specific individuals or teams with clear accountability.	MAY	Data Lakehouse Architecture Standard	6.3 – Functional Responsibilities
DATA-0434	The Contractor MUST register each Data Consuming Entity (DCE) in the Data Catalog with designated Data Owner, Data Steward, and intended use case.	MUST	Data Lakehouse Architecture Standard	6.3 – Functional Responsibilities
DATA-0435	The Contractor MUST document the business justification, specific datasets required, intended use, and access tier for each DCE before granting access.	MUST	Data Lakehouse Architecture Standard	6.3 – Functional Responsibilities
DATA-0436	Access tier assignments MUST be approved by the Data Owner and enforced through row-level and column-level security controls in the platform.	MUST	Data Lakehouse Architecture Standard	6.3 – Functional Responsibilities
DATA-0437	The Contractor MUST implement the three-layer Lakehouse architecture (Raw, Processed, Curated) as defined in this standard. Data MUST flow unidirectionally through the layers unless explicitly approved by OHIP.	MUST	Data Lakehouse Architecture Standard	6.4 – Layered Data Architecture
DATA-0438	The Contractor MUST store all incoming data in its native format without modification. No business transformations, cleansing, or enrichment MAY be applied in the Raw Layer.	MUST	Data Lakehouse Architecture Standard	6.4.1 – Raw Layer
DATA-0439	Format conversion for storage optimization (e.g., to Parquet) MAY be performed in the Raw Layer provided the original data content and values are preserved exactly and the conversion is documented.	MAY	Data Lakehouse Architecture Standard	6.4.1 – Raw Layer
DATA-0440	Each data load into the Raw Layer MUST include ingestion metadata documenting the source system identifier, extraction timestamp, load timestamp, and record count.	MUST	Data Lakehouse Architecture Standard	6.4.1 – Raw Layer

DATA-0441	The Contractor MUST comply with OHIP data retention policies for Raw Layer data as defined in Section 7.12.	MUST	Data Lakehouse Architecture Standard	6.4.1 – Raw Layer
DATA-0442	The Contractor MUST convert Processed Layer data into an open table format supporting ACID transactions. The format MUST support time travel and versioning capabilities.	MUST	Data Lakehouse Architecture Standard	6.4.2 – Processed Layer
DATA-0443	Apache Iceberg SHOULD be used as the preferred open table format for the current OHIP implementation. Alternative formats require OHIP approval.	SHOULD	Data Lakehouse Architecture Standard	6.4.2 – Processed Layer
DATA-0444	The Contractor MUST apply standardization transformations consistently: convert dates to ISO 8601 format, standardize and validate addresses against postal standards, normalize currency values to consistent precision, and map code values to enterprise-standard reference data.	MUST	Data Lakehouse Architecture Standard	6.4.2 – Processed Layer
DATA-0445	The Contractor MUST resolve and apply MDM identifiers. Every member record MUST be assigned applicable OHIP enterprise identifiers (HMID and Snowflake-managed cross-system ID linkages).	MUST	Data Lakehouse Architecture Standard	6.4.2 – Processed Layer
DATA-0446	The Contractor MUST document the record linkage methodology for each data domain, including match rules, confidence thresholds, and survivorship logic, before the first production load.	MUST	Data Lakehouse Architecture Standard	6.4.2 – Processed Layer
DATA-0447	The Contractor MUST maintain linkage audit logs capturing match decisions, confidence scores, and any manual overrides applied during MDM processing.	MUST	Data Lakehouse Architecture Standard	6.4.2 – Processed Layer
DATA-0448	The Contractor MUST model Curated Layer data into structures appropriate for its intended use: star or snowflake schemas for BI/analytics, flat denormalized tables for ML, and FHIR-conformant structures for clinical APIs.	MUST	Data Lakehouse Architecture Standard	6.4.3 – Curated Layer
DATA-0449	For machine learning and advanced analytics workloads, flat denormalized tables SHOULD be used to minimize join complexity and optimize query performance.	SHOULD	Data Lakehouse Architecture Standard	6.4.3 – Curated Layer
DATA-0450	The Contractor MUST restrict Curated Layer write access to technical personnel responsible for approved ETL/ELT	MUST	Data Lakehouse	6.4.3 – Curated Layer

	pipelines. Direct manual writes to Curated Layer tables are PROHIBITED.		Architecture Standard	
DATA-0451	The Contractor MUST maintain complete traceability for curated datasets back to their source records in the Processed and Raw Layers through documented lineage.	MUST	Data Lakehouse Architecture Standard	6.4.3 – Curated Layer
DATA-0452	The Contractor MUST include all mandatory metadata fields (Source_System_ID, Extraction_Timestamp, Load_Timestamp, Record_Count, Pipeline_Run_ID, Data_Classification, Batch_ID) for every data load.	MUST	Data Lakehouse Architecture Standard	6.5.1 – Mandatory Ingestion Metadata
DATA-0453	The Contractor MUST capture metadata at the micro-batch or transaction batch level (not the file level) for streaming data sources to ensure fine-grained lineage and auditability.	MUST	Data Lakehouse Architecture Standard	6.5.1 – Mandatory Ingestion Metadata
DATA-0454	The Contractor MUST design all ingestion pipelines for reliability, observability, and recoverability. Every pipeline MUST support restart from the point of failure without data loss or duplication.	MUST	Data Lakehouse Architecture Standard	6.5.2 – Ingestion Pipeline Requirements
DATA-0455	All ingestion pipelines MUST be idempotent: re-running the same pipeline with the same input MUST produce the same output with no side effects.	MUST	Data Lakehouse Architecture Standard	6.5.2 – Ingestion Pipeline Requirements
DATA-0456	For streaming ingestion where at-least-once delivery MAY create duplicates in the Raw Layer, deduplication MUST be applied before promotion to the Processed Layer.	MUST	Data Lakehouse Architecture Standard	6.5.2 – Ingestion Pipeline Requirements
DATA-0457	Pipelines MUST include comprehensive logging capturing start/end times, record counts at each transformation stage, reject counts and reasons, and exception details.	MUST	Data Lakehouse Architecture Standard	6.5.2 – Ingestion Pipeline Requirements
DATA-0458	The Contractor MUST implement automated monitoring and alerting for ingestion pipelines. Alerts MUST be triggered for pipeline failures, SLA breaches, and anomalous record count deviations.	MUST	Data Lakehouse Architecture Standard	6.5.2 – Ingestion Pipeline Requirements
DATA-0459	Alert notifications MUST be routed to appropriate personnel and integrated with the MES Incident Management System as defined in the Infrastructure and Operations Domain Document OHIP Information Technology Service Management Standard.	MUST	Data Lakehouse Architecture Standard	6.5.2 – Ingestion Pipeline Requirements

DATA-0460	For PostgreSQL and other databases, AWS Database Migration Service (DMS) with S3 targets SHOULD be used as the preferred CDC mechanism.	SHOULD	Data Lakehouse Architecture Standard	6.5.2.1 – Ingestion Pattern Requirements
DATA-0461	For Oracle sources, Oracle GoldenGate or equivalent CDC tools MAY be used for near real-time ingestion, subject to OHIP approval.	MAY	Data Lakehouse Architecture Standard	6.5.2.1 – Ingestion Pattern Requirements
DATA-0462	Low-code platforms (e.g., Salesforce) with native Snowflake connectors MAY use those connectors directly, subject to metadata and idempotency requirements being satisfied.	MAY	Data Lakehouse Architecture Standard	6.5.2.1 – Ingestion Pattern Requirements
DATA-0463	Snowpipe SHOULD be used as the preferred mechanism for batch file ingestion from S3 staging areas.	SHOULD	Data Lakehouse Architecture Standard	6.5.2.1 – Ingestion Pattern Requirements
DATA-0464	For event-driven architectures, AWS Kinesis or equivalent streaming services MAY be used with appropriate de-duplication applied before Processed Layer promotion.	MAY	Data Lakehouse Architecture Standard	6.5.2.1 – Ingestion Pattern Requirements
DATA-0465	The Contractor MUST implement all quality gates defined in this standard and MUST NOT promote data between layers unless quality gate checks have passed or a documented exception has been approved by OHIP.	MUST	Data Lakehouse Architecture Standard	6.6 – Data Quality Gates
DATA-0466	Quality Gate 1 (Raw to Processed) MUST enforce: schema validation at 100%, completeness of primary key fields at 100%, no null values in mandatory fields, data type conformance at 100%, and format validation per applicable standards.	MUST	Data Lakehouse Architecture Standard	6.6.1 – Quality Gate 1: Raw to Processed Layer
DATA-0467	Quality Gate 2 (Processed to Curated) MUST enforce: referential integrity at ≥99%, semantic accuracy of critical business elements at ≥99.9%, cross-domain consistency checks, and business rule validation as defined in the Data Management Plan.	MUST	Data Lakehouse Architecture Standard	6.6.2 – Quality Gate 2: Processed to Curated Layer
DATA-0468	For a Fail result: the pipeline MUST halt promotion of the affected batch, failing records MUST be routed to the Quarantine Zone, and an alert MUST be triggered for immediate investigation.	MUST	Data Lakehouse Architecture Standard	6.6.3 – Promotion Logic
DATA-0469	For a Warning result: the pipeline MUST promote the data but MUST flag affected records with a data quality warning	MUST	Data Lakehouse Architecture Standard	6.6.3 – Promotion Logic

	indicator and notify the Data Steward for review within the defined SLA.			
DATA-0470	The Contractor MUST assess data quality across all dimensions defined in the Data Quality Dimensions table and MUST adhere to the threshold requirements specified therein for each data classification tier.	MUST	Data Lakehouse Architecture Standard	6.6.4 – Data Quality Dimensions and Thresholds
DATA-0471	The quality gate validation logic MUST enforce thresholds programmatically. Records failing threshold checks MUST NOT be silently dropped: all failures MUST be logged with reason codes in the Quarantine Zone.	MUST	Data Lakehouse Architecture Standard	6.6.4 – Data Quality Dimensions and Thresholds
DATA-0472	The Contractor MUST implement scheduled data quality assessments on incoming data. At minimum, critical data MUST be assessed at each ingestion cycle.	MUST	Data Lakehouse Architecture Standard	6.6.5 – Data Quality Management Requirements
DATA-0473	The Contractor MUST maintain a database table of data quality issue history that can be queried to show trend analysis and recurring patterns over time.	MUST	Data Lakehouse Architecture Standard	6.6.5 – Data Quality Management Requirements
DATA-0474	The Contractor MUST adhere to a formalized process for identifying, tracking, and remediating data quality issues, integrated with the MES incident management system.	MUST	Data Lakehouse Architecture Standard	6.6.5 – Data Quality Management Requirements
DATA-0475	The Contractor MUST document all missing data patterns. Missing data MUST be classified as MCAR, MAR, or MNAR with appropriate handling strategies documented and approved by the Data Steward.	MUST	Data Lakehouse Architecture Standard	6.6.5 – Data Quality Management Requirements
DATA-0476	The Contractor MUST monitor and enforce timeliness in data availability. Data missing or delayed beyond defined SLAs MUST trigger automated alerts within the ingestion pipeline monitoring system.	MUST	Data Lakehouse Architecture Standard	6.6.5 – Data Quality Management Requirements
DATA-0477	The Contractor MUST use a data quality profiling and validation tool approved by OHIP to automate complex checks, calculate dimension scores, and generate quality reports.	MUST	Data Lakehouse Architecture Standard	6.6.5 – Data Quality Management Requirements
DATA-0478	The Contractor MUST develop KPIs for data quality, track metrics over time, and report on KPIs to OHIP on a monthly basis or as otherwise specified in the applicable contract.	MUST	Data Lakehouse Architecture Standard	6.6.5 – Data Quality Management Requirements

DATA-0479	For significant or recurring quality issues, the Contractor MUST conduct a root cause analysis and implement preventive measures. Root cause findings MUST be documented and shared with OHIP.	MUST	Data Lakehouse Architecture Standard	6.6.5 – Data Quality Management Requirements
DATA-0480	The Contractor MUST calculate each data quality dimension using standardized formulas and MUST report results using the OHIP-defined scoring framework.	MUST	Data Lakehouse Architecture Standard	6.6.6 – Data Quality Measurement Methodology
DATA-0481	The Contractor MUST implement automated data quality measurement processes that execute on a defined schedule and store results in a queryable quality metrics repository.	MUST	Data Lakehouse Architecture Standard	6.6.6 – Data Quality Measurement Methodology
DATA-0482	Critical data MUST be measured weekly with automated alerts triggered immediately when thresholds are breached. Standard data MUST be measured monthly.	MUST	Data Lakehouse Architecture Standard	6.6.6 – Data Quality Measurement Methodology
DATA-0483	When an individual measurement period falls below threshold, the Contractor MUST document and investigate the root cause, and implement corrective action within the defined SLA.	MUST	Data Lakehouse Architecture Standard	6.6.6 – Data Quality Measurement Methodology
DATA-0484	Non-compliant dataset instances from external or periodic sources MUST be rejected and returned to the provider with a documented quality report identifying specific failures.	MUST	Data Lakehouse Architecture Standard	6.6.6 – Data Quality Measurement Methodology
DATA-0485	The Contractor MUST document and maintain the Lakehouse data quality management approach within the Data Management Plan (Data Quality Management section), covering all required components. Once OHIP approval is obtained, the plan MUST be implemented and maintained throughout the contract term.	MUST	Data Lakehouse Architecture Standard	6.6.6 – Data Quality Measurement Methodology
DATA-0486	When quality measurements indicate thresholds are not met, formal remediation MUST be initiated: issue logged in incident management system, Data Steward notified within 4 hours, root cause analysis completed, and remediation plan submitted to OHIP.	MUST	Data Lakehouse Architecture Standard	6.6.6.5 – Remediation Process
DATA-0487	Remediation activities MUST be completed within 30 days of the threshold violation unless an alternative timeline is approved in writing by OHIP.	MUST	Data Lakehouse Architecture Standard	6.6.6.5 – Remediation Process

DATA-0488	The Contractor MUST implement a Quarantine Zone as part of their Lakehouse architecture. The Quarantine Zone MUST retain the original raw state of quarantined records without modification.	MUST	Data Lakehouse Architecture Standard	6.7.1 – Quarantine Zone Requirements
DATA-0489	The Quarantine Zone MAY be implemented as a dedicated cloud storage prefix, a separate database schema, or another isolation mechanism approved by OHIP.	MAY	Data Lakehouse Architecture Standard	6.7.1 – Quarantine Zone Requirements
DATA-0490	For each quarantined record, the Contractor MUST include a Rejection_Reason code that identifies which quality gate check failed, a Rejection_Timestamp, and the Pipeline_Run_ID of the failing run.	MUST	Data Lakehouse Architecture Standard	6.7.1 – Quarantine Zone Requirements
DATA-0491	Access to the Quarantine Zone MUST be restricted to authorized personnel responsible for data quality remediation. Business users and analysts MUST NOT have access to quarantined data.	MUST	Data Lakehouse Architecture Standard	6.7.1 – Quarantine Zone Requirements
DATA-0492	The Contractor MUST execute the structured remediation workflow defined in Section 7.7.2 when a data quality violation occurs, integrating with the MES incident management processes defined in the Infrastructure and Operations Domain Document OHIP ITSM Standard.	MUST	Data Lakehouse Architecture Standard	6.7.2 – Remediation Workflow
DATA-0493	Critical incidents (data corruption, PHI exposure risk, widespread pipeline failure) MUST be responded to within 2 hours, root cause completed within 4 hours, and remediation completed within 24 hours.	MUST	Data Lakehouse Architecture Standard	6.7.2 – Remediation Workflow
DATA-0494	Corrected data MUST be reprocessed through the standard pipeline including all quality gate validations before being promoted to the Processed or Curated Layer.	MUST	Data Lakehouse Architecture Standard	6.7.2 – Remediation Workflow
DATA-0495	The incident record MUST capture root cause analysis findings, the remediation approach taken, any code or configuration changes made, and the re-ingestion results for full audit closure.	MUST	Data Lakehouse Architecture Standard	6.7.2 – Remediation Workflow
DATA-0496	The Contractor MUST document lineage across the Data Lakehouse architecture for critical data pipelines and transformations, covering column-level lineage from source to each consumption point.	MUST	Data Lakehouse Architecture Standard	6.8 – Data Lineage

DATA-0497	The Contractor MUST capture for each dataset: the original source and system of record, the legal authority for data collection, the data owner and steward, and the applicable retention and sensitivity classification.	MUST	Data Lakehouse Architecture Standard	6.8.1 – Data Provenance
DATA-0498	Provenance documentation MUST be maintained in the data catalog and made accessible to OHIP and authorized auditors upon request.	MUST	Data Lakehouse Architecture Standard	6.8.1 – Data Provenance
DATA-0499	The Contractor MUST develop and maintain an enterprise data flow diagram showing major systems, data movement paths, transformation points, and consumption endpoints, updated whenever architectural changes are made.	MUST	Data Lakehouse Architecture Standard	6.8.2 – Lineage Documentation Requirements
DATA-0500	The Contractor MUST document the logic and input/output for each ETL job or data transformation process as part of the overall lineage documentation.	MUST	Data Lakehouse Architecture Standard	6.8.2 – Lineage Documentation Requirements
DATA-0501	All scripts, queries, or code performing data transformations MUST be under version control with a complete history of changes, authors, and deployment dates maintained in an OHIP-approved repository.	MUST	Data Lakehouse Architecture Standard	6.8.3 – Version Control and Audit Trail
DATA-0502	The Contractor MUST be able to trace from any reported value back through each transformation step to the original source record. This traceability MUST be demonstrable to OHIP upon request.	MUST	Data Lakehouse Architecture Standard	6.8.3 – Version Control and Audit Trail
DATA-0503	The Contractor MUST employ an OHIP-approved data lineage tracking tool that integrates with the Lakehouse platform and automatically captures column-level lineage for all pipeline executions.	MUST	Data Lakehouse Architecture Standard	6.8.4 – Lineage Tooling and Integration
DATA-0504	The Contractor MUST perform end-to-end reconciliation tests when major data transformations are changed, verifying that lineage documentation accurately reflects the updated transformation logic.	MUST	Data Lakehouse Architecture Standard	6.8.4 – Lineage Tooling and Integration
DATA-0505	The Contractor MUST consult lineage documentation and review and document the impact on all downstream datasets, reports, and interfaces before changing any upstream data source, schema, or transformation logic.	MUST	Data Lakehouse Architecture Standard	6.8.5 – Impact Analysis
DATA-0506	The Contractor MUST design and implement database structures that support data versioning enabling point-in-	MUST	Data Lakehouse	6.9.1 – Versioning Requirements

	time data recovery, historical state queries, and audit trail capabilities.		Architecture Standard	
DATA-0507	Versioning implementation MUST support: querying historical snapshots by timestamp, comparing data states across different time periods, and recovering specific historical versions of datasets.	MUST	Data Lakehouse Architecture Standard	6.9.1 – Versioning Requirements
DATA-0508	The Contractor MUST retain ninety (90) days of version history for the Processed Layer and Curated Layer to support operational troubleshooting and short-term audit requirements.	MUST	Data Lakehouse Architecture Standard	6.9.2 – Version Retention
DATA-0509	The Contractor MUST document versioning capabilities available to data consumers, including exact syntax for point-in-time queries, retention limits, and any cost or performance implications.	MUST	Data Lakehouse Architecture Standard	6.9.3 – Version Access and Documentation
DATA-0510	Access to historical versions MUST be governed using the same access controls as current data. Historical version access MUST be logged and auditable.	MUST	Data Lakehouse Architecture Standard	6.9.3 – Version Access and Documentation
DATA-0511	The Contractor MUST maintain a centralized Data Catalog approved by OHIP that provides visibility into all datasets, their ownership, quality status, lineage, and access controls across the Lakehouse.	MUST	Data Lakehouse Architecture Standard	6.10.1 – Data Catalog Requirements
DATA-0512	The Data Catalog MUST support registration of all tables, views, and data products; enable search and discovery by business users; display data quality scores and lineage; and enforce access control policies.	MUST	Data Lakehouse Architecture Standard	6.10.1 – Data Catalog Requirements
DATA-0513	For Snowflake-based implementations, the Contractor MUST leverage the Snowflake Horizon Catalog as the primary catalog solution, configured per OHIP specifications.	MUST	Data Lakehouse Architecture Standard	6.10.1 – Data Catalog Requirements
DATA-0514	Where business requirements exceed native Horizon Catalog capabilities, the Contractor MAY propose a supplementary catalog solution subject to OHIP approval and integration requirements.	MAY	Data Lakehouse Architecture Standard	6.10.1 – Data Catalog Requirements
DATA-0515	The Contractor MUST ensure all mandatory catalog attributes defined in the Mandatory Catalog Attributes table are populated and maintained for every registered dataset, with no attributes left blank without documented justification.	MUST	Data Lakehouse Architecture Standard	6.10.2 – Mandatory Catalog Attributes

DATA-0516	The Contractor MUST maintain an enterprise glossary within the Data Catalog defining key business terms, data elements, and their authoritative definitions, aligned with OHIP program terminology and MITA standards.	MUST	Data Lakehouse Architecture Standard	6.10.3 – Enterprise Glossary
DATA-0517	The Contractor MUST govern reference data through the data catalog, maintain a master list of all reference data domains, and ensure all reference data values are registered with their business definition, effective dates, and authoritative source.	MUST	Data Lakehouse Architecture Standard	6.10.4 – Reference Data Management
DATA-0518	Changes to reference data MUST follow a controlled process including impact analysis, reference data steward approval, and version history maintenance.	MUST	Data Lakehouse Architecture Standard	6.10.4 – Reference Data Management
DATA-0519	The Contractor MUST obtain OHIP approval before modification of reference data values used for federal reporting (T-MSIS, CMS-64) or cross-agency data exchange.	MUST	Data Lakehouse Architecture Standard	6.10.4 – Reference Data Management
DATA-0520	The Contractor MUST evaluate each proposed schema change and apply the appropriate classification (Non-Breaking, Potentially Breaking, or Breaking) using the classification criteria defined in this standard before initiating the change process.	MUST	Data Lakehouse Architecture Standard	6.11.1 – Schema Change Classification
DATA-0521	The Contractor MUST follow a documented change control process for all schema changes, with timelines and approval requirements determined by the change classification.	MUST	Data Lakehouse Architecture Standard	6.11.2 – Schema Change Process
DATA-0522	Non-breaking schema changes MAY proceed with Data Architect approval and documentation in the data catalog within 5 business days, without requiring broader stakeholder notification.	MAY	Data Lakehouse Architecture Standard	6.11.2 – Schema Change Process
DATA-0523	Potentially breaking changes MUST require: impact analysis identifying all affected consumers, a 10-business-day notification window for consumer teams, and Data Owner approval before implementation.	MUST	Data Lakehouse Architecture Standard	6.11.2 – Schema Change Process
DATA-0524	Breaking changes MUST require: impact analysis with explicit acknowledgment from all affected consumer teams, a 20-business-day notification window, OHIP approval, and a tested migration path before implementation.	MUST	Data Lakehouse Architecture Standard	6.11.2 – Schema Change Process

DATA-0525	Emergency schema changes that bypass normal timelines MUST obtain OHIP approval before implementation and MUST complete post-implementation impact documentation and affected consumer notification within 24 hours.	MUST	Data Lakehouse Architecture Standard	6.11.2 – Schema Change Process
DATA-0526	The Contractor SHOULD design schemas to support backward compatibility whenever feasible, adding new columns rather than modifying or removing existing ones.	SHOULD	Data Lakehouse Architecture Standard	6.11.3 – Backward Compatibility Requirements
DATA-0527	When breaking changes are unavoidable, the Contractor MAY develop versioned schemas or views that maintain the old interface for a defined transition period approved by OHIP.	MAY	Data Lakehouse Architecture Standard	6.11.3 – Backward Compatibility Requirements
DATA-0528	The Contractor MUST implement retention policies based on data classification as defined in the Retention Requirements table, applying the longest applicable retention period for datasets containing multiple data classifications.	MUST	Data Lakehouse Architecture Standard	6.12.1 – Retention Requirements by Data Classification
DATA-0529	The Contractor MUST document the retention period for each dataset in the data catalog and MUST implement automated controls that prevent deletion before the retention period expires.	MUST	Data Lakehouse Architecture Standard	6.12.1 – Retention Requirements by Data Classification
DATA-0530	Raw Layer data MUST be retained for the full retention period applicable to its classification. Data MAY only be deleted when a legal hold has been lifted or there is explicit OHIP authorization.	MUST	Data Lakehouse Architecture Standard	6.12.2 – Retention by Architectural Layer
DATA-0531	Processed Layer data MUST be retained for the required operational period plus ninety (90) days of version history. Longer retention periods required by data classification govern where applicable.	MUST	Data Lakehouse Architecture Standard	6.12.2 – Retention by Architectural Layer
DATA-0532	The Contractor MUST implement automated storage lifecycle policies to transition data to cost-appropriate storage tiers as it ages, in accordance with OHIP-approved transition guidelines.	MUST	Data Lakehouse Architecture Standard	6.12.3 – Storage Tier Transitions
DATA-0533	Data older than 90 days accessed infrequently MAY transition to S3 Standard-IA or S3 Glacier Instant Retrieval, subject to retrieval time requirements being met.	MAY	Data Lakehouse Architecture Standard	6.12.3 – Storage Tier Transitions
DATA-0534	Data older than one year retained primarily for compliance MAY transition to S3 Glacier Flexible Retrieval or S3 Glacier	MAY	Data Lakehouse	6.12.3 – Storage Tier Transitions

	Deep Archive, subject to OHIP retrieval time requirements.		Architecture Standard	
DATA-0535	PHI, FTI, and MCD data SHOULD be retrievable within timeframes agreed with OHIP based on data criticality and access frequency. Storage tier transitions for regulated data MUST be approved by OHIP before implementation.	SHOULD	Data Lakehouse Architecture Standard	6.12.3 – Storage Tier Transitions
DATA-0536	The Contractor MUST execute a controlled purge process including: verifying no legal hold applies, obtaining Data Owner approval, documenting the purge scope and justification, and retaining purge audit logs permanently.	MUST	Data Lakehouse Architecture Standard	6.12.4 – Data Purge Procedures
DATA-0537	The Contractor MUST implement legal hold flags and automated controls that prevent deletion of any dataset subject to a legal hold, regardless of retention period expiration.	MUST	Data Lakehouse Architecture Standard	6.12.4 – Data Purge Procedures
DATA-0538	The Contractor MUST maintain the capability to restore archived data to an accessible state within the timeframe defined for its classification tier, and MUST test this capability annually.	MUST	Data Lakehouse Architecture Standard	6.12.5 – Archival and Historical Access
DATA-0539	The Contractor MUST design and implement disaster recovery capabilities meeting: Tier 1 (Critical) RTO ≤4 hours and RPO ≤1 hour; Tier 2 (Important) RTO ≤24 hours and RPO ≤4 hours; Tier 3 (Standard) RTO ≤72 hours and RPO ≤24 hours.	MUST	Data Lakehouse Architecture Standard	6.13.1 – Recovery Objectives
DATA-0540	The Contractor MUST document dataset classification into tiers based on the Data Owner's Business Impact Analysis (BIA). Tier assignments MUST be reviewed and approved by OHIP annually.	MUST	Data Lakehouse Architecture Standard	6.13.1 – Recovery Objectives
DATA-0541	The Contractor MUST implement comprehensive backup procedures for all three architectural layers covering automated daily incremental backups, automated weekly full backups, and Snowflake Time Travel configuration.	MUST	Data Lakehouse Architecture Standard	6.13.2 – Backup Requirements
DATA-0542	The Contractor MUST configure Tier 1 Point-In-Time Recovery (PITR) backup with a minimum 30-day retention window for all Processed and Curated Layer tables containing PHI, FTI, or MCD.	MUST	Data Lakehouse Architecture Standard	6.13.2 – Backup Requirements
DATA-0543	The Contractor MUST generate and store Tier 2 monthly full database snapshots in	MUST	Data Lakehouse	6.13.2 – Backup Requirements

	S3 Glacier Deep Archive with a minimum 7-year retention for compliance purposes.		Architecture Standard	
DATA-0544	All Raw Layer objects MUST be replicated to the secondary region (US-East-2, Ohio) within the approved replication lag SLA. Replication status MUST be monitored and alerting configured for replication failures.	MUST	Data Lakehouse Architecture Standard	6.13.3 – Cross-Region Replication
DATA-0545	The Contractor MUST develop and maintain documented disaster recovery procedures covering: declaration criteria, activation sequence, role responsibilities, communication protocol, and validation steps for each recovery scenario.	MUST	Data Lakehouse Architecture Standard	6.13.4 – Disaster Recovery Procedures
DATA-0546	The Contractor MUST review and update disaster recovery procedures annually and whenever significant changes are made to the Lakehouse architecture or data classification.	MUST	Data Lakehouse Architecture Standard	6.13.4 – Disaster Recovery Procedures
DATA-0547	The Contractor MUST conduct: an annual full failover test validating complete Tier 1 RTO/RPO objectives, a quarterly recovery drill testing specific component recovery, and monthly backup restoration verification for a sample of critical datasets.	MUST	Data Lakehouse Architecture Standard	6.13.5 – Disaster Recovery Testing Requirements
DATA-0548	Test results MUST be documented including actual recovery times, issues encountered, and remediation actions taken. Test reports MUST be provided to OHIP within 10 business days of each test.	MUST	Data Lakehouse Architecture Standard	6.13.5 – Disaster Recovery Testing Requirements
DATA-0549	The Contractor MUST implement comprehensive testing for all data ingestion and transformation pipelines including unit tests for individual transformation functions, integration tests for complete pipeline flows, and regression tests.	MUST	Data Lakehouse Architecture Standard	6.14.1 – Pipeline Testing Requirements
DATA-0550	All pipeline code MUST be accompanied by automated tests executed as part of the deployment process. No pipeline code MAY be deployed to production without passing automated test suites.	MUST	Data Lakehouse Architecture Standard	6.14.1 – Pipeline Testing Requirements
DATA-0551	Testing logs MUST be retained per Section 7.12 retention periods and MUST be accessible to designated OHIP staff and auditors.	MUST	Data Lakehouse Architecture Standard	6.14.1 – Pipeline Testing Requirements
DATA-0552	The Contractor MUST conduct reconciliation testing comparing record counts and key metrics between source systems and each Lakehouse layer after	MUST	Data Lakehouse Architecture Standard	6.14.2 – Data Validation Testing

	every major pipeline deployment or data migration.			
DATA-0553	For batch data loads, the Contractor MUST implement anomaly testing including volume delta analysis, statistical outlier detection on key metrics, and referential integrity validation.	MUST	Data Lakehouse Architecture Standard	6.14.2 – Data Validation Testing
DATA-0554	Anomaly thresholds MUST be defined in collaboration with the Data Steward for each data domain and MUST be approved by OHIP before the pipeline goes to production.	MUST	Data Lakehouse Architecture Standard	6.14.2 – Data Validation Testing
DATA-0555	The Contractor MUST validate that pipelines and queries meet throughput and latency requirements specified in the applicable contract SLAs before production deployment and after significant architecture changes.	MUST	Data Lakehouse Architecture Standard	6.14.3 – Performance Testing
DATA-0556	The Contractor MUST validate quality gates through testing. Test cases MUST include known-good data (expected pass), data with known violations (expected fail), and edge cases for each threshold.	MUST	Data Lakehouse Architecture Standard	6.14.4 – Quality Gate Testing
DATA-0557	Quality gate threshold values MUST be validated against historical data to confirm they are appropriately calibrated before production deployment.	MUST	Data Lakehouse Architecture Standard	6.14.4 – Quality Gate Testing
DATA-0558	The Contractor MUST maintain test environments logically isolated from production. Test data MUST be representative of production data volume, variety, and complexity without exposing real PHI, PII, or regulated data.	MUST	Data Lakehouse Architecture Standard	6.14.5 – Test Environment Requirements
DATA-0559	Non-production environments MUST NOT use live production data. Exceptions MUST be formally justified, approved by the OHIP Privacy Officer, and time-limited with enhanced access controls.	MUST	Data Lakehouse Architecture Standard	6.14.5 – Test Environment Requirements
DATA-0560	The Contractor MUST use synthetic data generation or data masking/de-identification for test cases requiring realistic data characteristics.	MUST	Data Lakehouse Architecture Standard	6.14.5 – Test Environment Requirements
DATA-0561	The Contractor MUST produce and maintain a Data Protection Plan (DPP) documenting how production-representative test data is generated, controlled, and disposed of after test activities.	MUST	Data Lakehouse Architecture Standard	6.14.5 – Test Environment Requirements
DATA-0562	The Contractor MUST implement the Raw Layer using OHIP-managed S3 storage	MUST	Data Lakehouse	6.15.1 – Storage Architecture

	with data files in S3 buckets under the OHIP-designated AWS account, following the approved bucket naming and folder hierarchy conventions.		Architecture Standard	
DATA-0563	For the Processed and Curated Layers, the Contractor MAY use either Snowflake-managed tables or external tables on S3 (Iceberg format), based on workload requirements and with OHIP approval.	MAY	Data Lakehouse Architecture Standard	6.15.1 – Storage Architecture
DATA-0564	S3 buckets MUST be configured with server-side encryption, bucket versioning enabled, and lifecycle policies aligned with Section 7.12 retention requirements.	MUST	Data Lakehouse Architecture Standard	6.15.1 – Storage Architecture
DATA-0565	The Contractor MUST use Snowflake Tables with Iceberg format where time travel and versioning requirements apply, ensuring ACID compliance and support for point-in-time queries.	MUST	Data Lakehouse Architecture Standard	6.15.2 – Table Formats
DATA-0566	The Contractor MUST implement workload isolation using Snowflake Multi-Cluster Warehouses with separate virtual warehouses for ingestion, transformation, and consumption workloads.	MUST	Data Lakehouse Architecture Standard	6.15.3 – Compute Resource Management
DATA-0567	Warehouse sizing and auto-scaling policies MUST be configured based on workload characteristics and validated during performance testing. Credit consumption MUST be monitored and reported to OHIP monthly.	MUST	Data Lakehouse Architecture Standard	6.15.3 – Compute Resource Management
DATA-0568	The Contractor MUST comply with all requirements in the Security, Privacy, and Compliance Domain Document OHIP Security and Privacy Compliance Standard and implement Snowflake-specific security controls defined in this section.	MUST	Data Lakehouse Architecture Standard	6.15.4 – Security and Privacy Governance
DATA-0569	The Contractor MUST implement row-level security policies, apply dynamic data masking to PHI/PII columns for non-privileged users, and configure network policies restricting Snowflake access to approved IP ranges.	MUST	Data Lakehouse Architecture Standard	6.15.4 – Security and Privacy Governance
DATA-0570	The Contractor SHOULD Ensure all access grants and revocations are integrated with OHIP's enterprise audit logging	SHOULD	Data Lakehouse Architecture Standard	6.15.4 – Security and Privacy Governance
DATA-0571	The Contractor SHOULD leverage Snowflake Horizon or equivalent governance capabilities approved by OHIP.	SHOULD	Data Lakehouse Architecture Standard	6.15.4 – Security and Privacy Governance

DATA-0572	Row-level and column-level security MUST be implemented for PHI/PII data in the Curated Layer. Data masking policies MUST be applied to prevent unauthorized access to sensitive attributes.	MUST	Data Lakehouse Architecture Standard	6.15.4.1 – Lakehouse-Specific Security
DATA-0573	The Contractor MUST implement the Quarantine Zone as a Snowflake Secure Schema with access restricted to the data quality remediation team role, preventing unauthorized read or write access.	MUST	Data Lakehouse Architecture Standard	6.15.5 – Quarantine Zone Implementation
DATA-0574	Snowflake configuration (warehouses, databases, schemas, roles) MUST be defined in version-controlled Infrastructure-as-Code using an OHIP-approved tool (e.g., Terraform). Manual configuration changes to production are PROHIBITED.	MUST	Data Lakehouse Architecture Standard	6.15.6 – Database DevOps and Change Management
DATA-0575	All database objects, transformations, and configurations MUST be stored in an OHIP-approved version control repository with CI/CD pipelines enforcing automated testing before any deployment to production.	MUST	Data Lakehouse Architecture Standard	6.15.6 – Database DevOps and Change Management
DATA-0576	For bulk loading: external stages on S3 MUST be used for bulk file ingestion; COPY INTO commands MUST include ON_ERROR handling; file formats MUST be explicitly defined and version-controlled; and loading history MUST be retained for audit purposes.	MUST	Data Lakehouse Architecture Standard	6.15.7 – Snowflake Data Loading Standards
DATA-0577	Snowpipe SHOULD be used for event-driven ingestion patterns where low latency is required and for continuous ingestion from S3.	SHOULD	Data Lakehouse Architecture Standard	6.15.7 – Snowflake Data Loading Standards
DATA-0578	Snowpipe error handling MUST route failed files to a designated error location for remediation. Pipe status MUST be monitored via PIPE_USAGE_HISTORY with alerts configured for pipe failures.	MUST	Data Lakehouse Architecture Standard	6.15.7 – Snowflake Data Loading Standards
DATA-0579	External stages (S3) are REQUIRED for production data pipelines. Stage credentials MUST be managed via storage integrations; embedded credentials in stage definitions are PROHIBITED.	MUST	Data Lakehouse Architecture Standard	6.15.7 – Snowflake Data Loading Standards
DATA-0580	Internal stages MAY be used for temporary data or small-volume loads only. Internal stages MUST NOT be used for production data pipelines.	MAY	Data Lakehouse Architecture Standard	6.15.7 – Snowflake Data Loading Standards
DATA-0581	The Contractor MUST implement Snowflake Row Access Policies to filter data based on the requesting role's access	MUST	Data Lakehouse	6.15.8 – Data Delivery and

	tier, enforcing data minimization at the query level.		Architecture Standard	Consumption Standards
DATA-0582	All data delivery activities MUST be logged and auditable. Snowflake Access History MUST be enabled and query logs MUST be retained per Section 7.12 retention requirements.	MUST	Data Lakehouse Architecture Standard	6.15.8 – Data Delivery and Consumption Standards
DATA-0583	All Snowflake object names MUST use UPPER_SNAKE_CASE format. Names MUST be meaningful and business-oriented, complying with Section 3.4.	MUST	Data Lakehouse Architecture Standard	6.15.9 – Snowflake Naming Conventions
DATA-0584	Schema names MUST indicate the architectural layer and domain (RAW_[DOMAIN], PROC_[DOMAIN], CUR_[DOMAIN]) to make data location immediately identifiable without requiring additional documentation.	MUST	Data Lakehouse Architecture Standard	6.15.9 – Snowflake Naming Conventions
DATA-0585	The Contractor MUST report Performance Monitoring KPIs to OHIP monthly, provide root cause analysis for any KPI breaches, and implement corrective actions within the timeframes defined in the applicable contract SLAs.	MUST	Data Lakehouse Architecture Standard	6.16 – Performance Monitoring and KPIs
DATA-0586	Pipeline Success Rate MUST exceed 99.9%. Quality Fail Rate MUST remain below 0.5%. Traceability Coverage MUST be 100% for critical datasets. Catalog Currency MUST be 100% for all active datasets.	MUST	Data Lakehouse Architecture Standard	6.16 – Performance Monitoring and KPIs
DATA-0587	Compliance with this standard is REQUIRED upon publication. Compliance with amended policies and standards is required within the timeframe specified in the amendment notice, or immediately if no timeframe is specified.	MUST	Data Lakehouse Architecture Standard	6.17 – Compliance and Enforcement
DATA-0588	A Contractor with existing implementations not conforming to this standard MUST submit a Compliance Roadmap to OHIP within 60 days of contract award identifying remediation steps, timelines, and responsible parties.	MUST	Data Lakehouse Architecture Standard	6.17.1 – Applicability
DATA-0589	The Contractor MUST provide access to systems, documentation, and personnel as needed to support OHIP compliance verification reviews, CMS SMC certification activities, and authorized audits.	MUST	Data Lakehouse Architecture Standard	6.17.2 – Compliance Verification
DATA-0590	The Contractor MUST document the Lakehouse data quality management approach within the Data Management Plan (Data Quality Management section),	MUST	Data Lakehouse Architecture Standard	6.6 – Data Quality Gates

	covering all required components, submitted to OHIP for approval before the first production data load.			
DATA-0591	The Contractor MUST develop Purge Request Documentation for all data deletion requests, including legal hold verification, Data Owner approval, purge scope, and post-purge audit confirmation.	MUST	Data Lakehouse Architecture Standard	6.12 – Data Retention and Lifecycle Management
DATA-0592	The Contractor MUST produce and submit all Deliverables and Artifacts defined in Appendix A (including the Purge Request Documentation) in the format, structure, and cadence specified by OHIP.	MUST	Data Lakehouse Architecture Standard	Appendix A
DATA-0593	Upon notice of modifications to Deliverables and Artifacts requirements, the Contractor MUST comply with all modifications and implement them promptly.	MUST	Data Lakehouse Architecture Standard	Appendix A
DATA-0594	The Contractor MUST develop and submit a Content Management Plan for OHIP review and approval prior to Enterprise Content Management (ECM) implementation activities.	MUST	Content and Document Management Standard	7.4 – Content Lifecycle Management
DATA-0595	All content received from external parties MUST be captured into an OHIP-approved ECM system; storage of content classified as PHI, PII, MCD, FTI, or INTERNAL in personal mailboxes, local workstation drives, or unmanaged shared drives is PROHIBITED .	MUST	Content and Document Management Standard	7.4.1 – Capture and Ingestion
DATA-0596	Scanned and imaged content MUST be captured at a minimum of 300 DPI for textual documents and 24-bit color for forms with graphical or signature elements.	MUST	Content and Document Management Standard	7.4.1 – Capture and Ingestion
DATA-0597	Optical Character Recognition (OCR) MUST be applied to all scanned textual content and OCR confidence scores MUST be retained as content metadata.	MUST	Content and Document Management Standard	7.4.1 – Capture and Ingestion
DATA-0598	Content with OCR confidence below an OHIP-approved threshold MUST be flagged for human review.	MUST	Content and Document Management Standard	7.4.1 – Capture and Ingestion
DATA-0599	Each captured item MUST receive a unique, immutable content identifier and MUST be associated with a capture event record (date, time, originating system or user, and channel).	MUST	Content and Document Management Standard	7.4.1 – Capture and Ingestion
DATA-0600	Every content item MUST be classified according to the OHIP data classification	MUST	Content and Document	7.4.2 – Classification

	taxonomy (PHI, FTI, MCD, PII, INTERNAL, PUBLIC).		Management Standard	
DATA-0601	When content sensitivity is ambiguous, content SHOULD be classified conservatively (most-restrictive) pending formal review, and the ambiguity MUST be resolved within [TBD: number of business days, to be confirmed with OHIP].	SHOULD	Content and Document Management Standard	7.4.2 – Classification
DATA-0602	Where content contains multiple sensitivity levels, the highest applicable classification MUST be assigned to the content item as a whole.	MUST	Content and Document Management Standard	7.4.2 – Classification
DATA-0603	Content classified as PHI, PII, MCD, or FTI MUST be stored only in OHIP-approved ECM repositories meeting the encryption, access-control, and audit requirements of the OHIP Security, Privacy, and Compliance Domain Document, Security and Privacy Compliance Standard.	MUST	Content and Document Management Standard	7.4.3 – Storage and Repositories
DATA-0604	Content MUST be stored with versioning enabled. All material modifications MUST create a new immutable version. Prior versions SHOULD be retained in accordance with the applicable records retention schedule, and versions subject to a legal hold MUST be retained for the duration of the hold	MUST	Content and Document Management Standard	7.4.3 – Storage and Repositories
DATA-0605	Content SHOULD NOT be duplicated across repositories without an approved, documented business justification; where duplication is unavoidable, one repository MUST be designated the System of Record for that content type.	SHOULD	Content and Document Management Standard	7.4.3 – Storage and Repositories
DATA-0606	Access to content MUST be enforced through Role-Based Access Control (RBAC) integrated with the Department's enterprise Identity and Access Management (IAM) solution; local repository accounts MUST NOT be used after initial ECM deployment.	MUST	Content and Document Management Standard	7.4.4 – Access and Use
DATA-0607	Content access events (read, download, print, export, share) for PHI, FTI, PII, and MCD content MUST be logged with sufficient detail to support audit and FOIL response; access events for INTERNAL and PUBLIC content SHOULD be logged to the same standard.	MUST	Content and Document Management Standard	7.4.4 – Access and Use
DATA-0608	Content MUST NOT be transmitted outside OHIP-controlled environments except through OHIP-approved secure	MUST	Content and Document Management Standard	7.4.4 – Access and Use

	channels documented in the Content Management Plan.			
DATA-0609	Content MUST be retained according to the longer of: the OHIP retention schedule, the New York State Archives retention schedule for Department records, or any retention period imposed by litigation hold or regulatory mandate.	MUST	Content and Document Management Standard	7.4.5 – Retention and Disposition
DATA-0610	Disposition (purge or archive) MUST be executed only through documented disposition workflows that produce an audit trail; manual deletion by individual users MUST NOT be permitted for content classified at INTERNAL or higher.	MUST	Content and Document Management Standard	7.4.5 – Retention and Disposition
DATA-0611	Content under active litigation, audit, or investigative hold MUST NOT be disposed of regardless of retention-schedule expiration; hold status MUST be tracked as content metadata.	MUST	Content and Document Management Standard	7.4.5 – Retention and Disposition
DATA-0612	Disposition events MUST be logged, and the disposition log SHOULD be retained in accordance with the applicable records retention schedule.	MUST	Content and Document Management Standard	7.4.5 – Retention and Disposition
DATA-0613	The Contractor MUST capture and maintain the mandatory content metadata set defined in Table 49 for every content item.	MUST	Content and Document Management Standard	7.5 – Metadata and Catalog Integration
DATA-0614	The Contractor MUST integrate content metadata with the Enterprise Data Catalog defined in Section 7.10.	MUST	Content and Document Management Standard	7.5 – Metadata and Catalog Integration
DATA-0615	The Contractor SHOULD maintain links between content items and related structured records (claim, member, provider, case, etc.) in the Enterprise Data Catalog.	SHOULD	Content and Document Management Standard	7.5 – Metadata and Catalog Integration
DATA-0616	ECM repositories MUST support full-text search across OCR'd content and metadata, with results exportable in standard e-discovery production formats (EDRM XML, Concordance DAT, or Relativity-compatible).	MUST	Content and Document Management Standard	7.6 – E-Discovery Search and Production
DATA-0617	Redaction tooling MUST be available to mask PHI, PII, FTI, MCD, and other protected information prior to production; redactions SHOULD be applied as a separate layer that preserves the underlying content for in-house review, and redacted content MUST NOT be recoverable in production exports.	MUST	Content and Document Management Standard	7.6 – E-Discovery Search and Production

DATA-0618	The Contractor MUST define and apply quality controls for captured content including duplicate detection, completeness checks, and OCR-confidence thresholds, with quality issues tracked and remediated under Section 2.15.	MUST	Content and Document Management Standard	7.7 – Content Quality and Information Governance
DATA-0619	The Contractor MUST monitor and report on content proliferation indicators (growth rate per repository, duplicate rate, orphaned content rate) and MUST propose remediation when OHIP-approved thresholds are exceeded.	MUST	Content and Document Management Standard	7.7 – Content Quality and Information Governance
DATA-0620	The Contractor MUST obtain Bureau of Enterprise Data Governance (BEDG) approval for all content classification changes that affect retention class, regulatory indicators, or sensitivity classification at the content-type level.	MUST	Content and Document Management Standard	7.7 – Content Quality and Information Governance
DATA-0621	The Contractor MUST use OHIP-approved Enterprise Content Management (ECM) systems and MUST comply with the OHIP Infrastructure and Operations Domain Document, Preferred Operational Tools Standard for selection of ECM, e-discovery, and redaction tooling.	MUST	Content and Document Management Standard	7.8 Tools and Standards
DATA-0622	Proprietary or unsupported content formats MUST be migrated to standards-based formats (PDF/A for archival, ODF or DOCX for editable content, XML or JSON for structured exchange) during ingest where feasible.	MUST	Content and Document Management Standard	7.8 Tools and Standards

Appendix C: MITA Service Definition Packages

MITA prescribes as part of the Business Service Definition Package and Technical Service Definition Package in Part III Chapter's 3 and 4.⁸⁴

The following is a recap of the MITA Service Definition Package that must be accounted for when documenting Business Services in OHIP's Enterprise Architecture Repository:

Business Service Parts

- **Service Name:** The title of the service.
- **Business Service Definition Package (BSDP):** Defined set of metadata describing the service.
- **Service Contract:** Describes the expected behavior of the interface as well as the security and privacy constraints on the service. The following are examples of interface behavior patterns:
 1. One-way only receives or outputs data (e.g., report generator);
 2. Involved parties negotiate a specific Service Level Agreement (SLA); and,
 3. Two-way receives and sends data.

Two-way traffic has two (2) other attributes:

1. Initiator defines who initiates the interface, either the service (e.g., request for information) or the outside client (e.g., inquiry); and,
 2. Processing characteristic defines the relationship between the input and the output:
 - Point-of-Sale (POS) transaction: A real-time transaction (e.g., a pharmacy POS) that features a very constrained response time and high reliability.
 - Online transaction: An inquiry on a provider that features a more relaxed response time while still allowing for conversation-type human interaction.
 - Batch: Typical batch processing constraints.
 - Asynchronous: No constraints on processing times but response and coordinating data are required.
- **Purpose:** Developed from the business process definition and associated BCM entry.
 - **Business Logic:** Describes what is occurring within the business service. It documents the underlying logic, functionality, and capability provided by the service. Initially, the business logic is free-form text or template driven (e.g., MITA Business Process Template), that an application developer will define and code in order to incorporate into a business process orchestration step and possibly integrate to a rules engine.
 - **Constraints:** List of any constraints of the service. The MITA business process triggers, results, and the MITA Logical Data Model (LDM) provide a formal interface definition. It also documents the interfaces and operations used by the service. Initially, the MITA Framework provides an

⁸⁴ Refer to [Medicaid CHIP Part III Zip File](#) for more guidance.

informal textual description (i.e., template) of the services interface. As groups populate MITA Services, they can then define and develop interfaces using WSDL interface descriptions.

- **Use Case:** Documents the main success path and critical failure conditions of the service. Initially, the TA writes a use case in free-form text and then models it with OMG UML prior to the development of the business services.
- **Solution Set:** Maps the specific detail developed to execute a service.
- **Structure Diagram:** Graphically depicts the business logic performed by the service and interconnects to the solution sets.
- **Performance Standards:** Define the anticipated performance metrics of the service so all stakeholders measure the same criteria in the same manner.
- **Test Scenarios and Test Cases:** Documentation to validate compliance to the service contract.
- **Map to MITA Data Models:** A trace of data used by the service (e.g., data in motion and shared business data only) to the MITA Conceptual Data Model (CDM) and LDM. This is an enhancement of the mapping done for the business service. Both American National Standards Institute (ANSI) Accredited Standards Committee (ASC) X12 positional transactions and W3C XML Schema are examples of defined data in motion.

Technical Service Parts

- **Service Name:** The name of the service component invoked by the Technical Service Definition Package (TSD P) that is a MITA-defined set of metadata describing the service. The service contract information describes the expected behavior of the interface (e.g., whether the interface is real-time or an online interface), as well as the security and privacy constraints on the service (i.e., requires strong encryption).

The following are examples of interface behavior patterns:

- 1) A one-way interface only receives or outputs data (e.g., report generator);
- 2) A two-way interface receives and sends data.

Two-way traffic has two other attributes:

- (1) initiator defines who initiates the interface, either the service (e.g., request for information) or the outside client (e.g., inquiry)
 - (2) processing characteristic defines the relationship between the input and the output:
 - Point-of-sale (POS) transaction: A real-time transaction (e.g., a pharmacy POS) that features very constrained response time and high reliability.
 - Online transaction: An inquiry on a provider that features more relaxed response time while still allowing for conversation-type human interaction.
 - Batch: Typical batch processing constraints.
 - Asynchronous: No constraints on processing times but require response and data coordination.
- **Purpose:** The technical area's principles and TCM depict the purpose.
 - **Business Logic:** Initially, the business logic is free-form text or template driven (e.g., MITA Business Process Template), application developers will eventually define and code business logic in order to incorporate into business process orchestration steps and standardized business

rule definitions. States must ensure the use of business rules engines to separate business rules from core programming and must provide information about the change control process that will manage development and implementation of business rules. States must be able to accommodate changes to business rules on a regularized schedule and on an emergency basis.

- **Constraints:** Lists constraints of the service, e.g., loose authentication control.
- **Formal Interface Definition:** Defines the services triggers and results. It also documents the interfaces and operations the service uses while using Web Service Definition Language (WSDL). Initially, the MITA Framework provides an informal textual description (i.e., template) of the services interface. As stakeholders develop the MITA services, they also develop the WSDL interface descriptions.
- **Use Case:** The architect or designer writes the use case in free-form text and supports the text with OMG UML models prior to the development of the business services.
- **Solution Set:** Maps the solution set(s) currently available to execute this service. Solution sets are previously defined groupings of technical services that are operational.
- **Structure and Activity Diagrams:** Graphically depicts the business logic performed by the service and interconnects the solution sets. The architect or designer models the diagram in OMG Business Process Model and Notation (BPMN). Using a modeling tool based on UML notation produces an UML diagram as part of the set.
- **Performance Standards:** Defines the anticipated performance standards of the service so all stakeholders measure the same metrics in the same way.
- **Test Scenarios and Test Cases:** Validates documentation of compliance to the service contract.
- **Map to MITA Data Models:** Maps a trace of data used by the service (e.g., data in motion and shared business data only) to the MITA Conceptual Data Model (CDM) and Logical Data Model (LDM). This is an enhancement of the mapping done for the business service. The XML schema defines the data in motion. American National Standards Institute (ANSI) Accredited Standards Committee (ASC) X12 positional transactions and World Wide Web Consortium (W3C) XML Schemas are examples of defined data in motion.

Appendix D: Acronyms

The following acronyms are used throughout the OHIP Data and Information Management Domain Document.

API: Application Programming Interface
ARB: Architecture Review Board
BC/DR: Business Continuity/Disaster Recovery
CAP: Corrective Action Plan
CI/CD: Continuous Integration/Continuous Deployment
CMS: Centers for Medicare & Medicaid Services
COTS: Commercial Off-The-Shelf
DDI: Design, Development, and Implementation
DOH: Department of Health (New York State)
DOS: Division of Operations and Systems (New York State)
E&E: Eligibility and Enrollment
EDW: Enterprise Data Warehouse
eMedNY: Electronic Medicaid of New York
FFP: Federal Financial Participation
FHIR: Fast Healthcare Interoperability Resources
FTI: Federal Tax Information
IaC: Infrastructure as Code
KPI: Key Performance Indicator
MECM: Medicaid Eligibility and Client Management
MES: Medicaid Enterprise System
MITA: Medicaid Information Technology Architecture
M&O: Maintenance and Operations
OHIP: Office of Health Insurance Programs (New York State)
OMG: Object Management Group
OWASP: Open Web Application Security Project
PHI: Protected Health Information
PII: Personally Identifiable Information
RCA: Root Cause Analysis
RFP: Request for Proposal
SaaS: Software as a Service
SDLC: Solution Development Life Cycle
SDR: Standards Deviation Request
SLA: Service Level Agreement
SMC: Streamlined Modular Certification
SOR: System of Record
UAT: User Acceptance Testing
UML: Unified Modeling Language