Data Architecture Quality

> **Source File:** generated-documents\dmbok\data-architecture-quality.md
> **Generated:** 30/07/2025 at 06:57:25
> **Generated by:** Requirements Gathering Agent - PDF Converter

Data Architecture & Quality

Generated by adpa-enterprise-framework-automation v3.2.0
Category: dmbok
Generated: 2025-07-18T06:34:14.586Z
Description: Defines the standards, principles, and practices for data architecture and quality management.

Data Architecture & Quality Document

For: ADPA – Advanced Document Processing & Automation Framework
Aligned with DMBOK 2.0 Best Practices

---

1. Introduction

1.1 Purpose and Scope

This document defines the Data Architecture and Data Quality framework for the ADPA platform, aligning with the DAMA-DMBOK 2.0 and enterprise best practices. It establishes principles, standards, and approaches to ensure data is structured, integrated, and managed with high quality and compliance across all ADPA modules, including AI-powered document generation, analytics, and enterprise integrations.

1.2 Alignment with Enterprise Architecture

The ADPA data architecture is an integral part of the overall enterprise architecture, supporting business, application, and technology domains. It enables modularity, scalability, and compliance for document automation, analytics, and system interoperability.

1.3 Stakeholders

- **Data Architect**: Defines and maintains data architecture standards.
- **Business Analyst**: Ensures data supports business requirements.
- **Developers/Engineers**: Implement data models and integration logic.
- **Data Stewards**: Oversee data quality and governance.
- **Compliance Officer**: Ensures adherence to regulatory requirements.
- **Project Manager**: Coordinates implementation and resource allocation.

---

2. Data Architecture Principles

2.1 Core Principles

- **Data as an Asset**: Data is a key enterprise asset, governed and managed for value.
- **Standardization & Reuse**: Data models, metadata, and interfaces are standardized and reused across modules.
- **Interoperability**: Data structures facilitate integration with external systems (Confluence, SharePoint, Adobe, etc.).

- **Security & Privacy by Design**: Data architecture embeds security and privacy controls throughout.
- **Scalability & Flexibility**: Supports modular growth, extensibility, and performance at enterprise scale.

2.2 Reference Architectures

- **Layered Architecture**: Separation of concerns (Ingestion, Storage, Processing, Access, Integration).
- **API-First Design**: All data access and manipulation via secured REST APIs (OpenAPI/TypeSpec-driven).
- **Metadata-driven**: All data assets described with rich metadata for discovery and governance.

2.3 Standards and Guidelines

- **Naming Conventions**: Consistent, descriptive entity, attribute, and file naming.
- **Data Modeling**: Logical and physical models (JSON Schema, ERD, UML) for all core data domains.
- **Versioning**: API and schema versioning for backward compatibility.
- **Documentation**: All data structures, flows, and dictionaries are documented and accessible.

---

3. Current State Assessment

3.1 Existing Data Architecture

- **Core Storage**: JSON-based configurations and document outputs; extensible to SQL/NoSQL DBs.
- **Integration**: RESTful APIs, CLI, and direct connectors to SharePoint, Confluence, Adobe, and VCS.
- **Metadata Management**: Initial support via configuration and template registries.

- **Security**: JWT-based authentication, OAuth2/SAML integration, and role-based access control.

3.2 Data Flows and Integration Points

- **Ingestion**: User input, external API calls, and file uploads.
- **Processing**: AI-powered modules, document generators, and validation engines.
- **Distribution**: Direct publishing to enterprise systems and content repositories.

3.3 Gaps and Challenges

- Fully normalized data models not yet implemented for all document types.
- Metadata and data lineage capabilities in progress.
- Automated data quality checks and profiling require expansion.
- Broader support for structured relational storage and analytics planned.

---

4. Target Data Architecture

4.1 Future State Vision

A unified, scalable, standards-based data architecture supporting:

- Modular document/data model expansion
- End-to-end metadata and lineage tracking
- High-quality, governed, and secure data flows across all modules and integrations

4.2 Technology Stack

- **Backend**: Node.js/TypeScript, Express.js
- **APIs**: OpenAPI 3.0, TypeSpec

- **Storage**: JSON (current), extensible to PostgreSQL, MongoDB, or Azure CosmosDB
- **Integration**: OAuth2/SAML, Microsoft Graph, Adobe PDF Services
- **Analytics**: Built-in reporting, extensible to BI platforms

4.3 Data Models and Schemas

- **Logical Data Models**: Defined for all document types (requirements, project, data management, etc.).
- **Physical Schemas**: JSON Schema for API payloads; ERD for structured storage (planned).
- **Enterprise Data Dictionary**: Canonical definitions for all data elements, including business meaning, format, and usage.

4.4 Integration Architecture

- **API Layer**: All data exchange via authenticated RESTful endpoints.
- **Connectors**: Managed integrations with SharePoint, Confluence, Adobe, VCS.
- **Event-Driven Extensions**: Future support for webhooks and message queues (Kafka, Azure Event Grid).

---

5. Data Quality Management

5.1 Data Quality Dimensions

- **Accuracy**: Data reflects the real-world asset or event.
- **Completeness**: All required fields populated.
- **Consistency**: No conflicting values across modules.
- **Timeliness**: Data is up-to-date and processed promptly.
- **Uniqueness**: No unintended duplicates.
- **Validity**: Data conforms to defined formats and rules.

5.2 Data Profiling and Assessment

- **Profiling Tools**: Automated scanning of data for anomalies and patterns (via validation libraries like Joi, Zod, AJV).
- **Quality Reports**: Periodic reports on completeness, accuracy, and conformance.

5.3 Data Cleansing and Enrichment

- **Validation Routines**: Enforced at API and CLI input layers.
- **Cleansing Scripts**: Automated removal of duplicates, correction of known issues.
- **Enrichment**: Standardization (e.g., date formats, entity names), metadata tagging.

5.4 Data Quality Metrics and KPIs

- **Error Rate**: Number of validation errors per document type.
- **Completeness Score**: Percentage of mandatory fields filled.
- **Timeliness**: Processing time from ingestion to output.
- **User Feedback**: Rate of rejected/accepted documents.
- **Audit Logs**: Traceability of all changes and corrections.

---

6. Governance and Compliance

6.1 Data Governance Structure

- **Data Governance Council**: Cross-functional team overseeing data policies.
- **Data Stewards**: Assigned for each data domain/document type.
- **Roles & Responsibilities**: Clearly defined in the Data Stewardship & Roles document.

6.2 Policies and Standards

- **Data Lifecycle Management**: Defined retention, archival, and disposal rules.

- **Access Control**: RBAC, least-privilege policies, periodic access reviews.
- **Quality Standards**: Minimum thresholds for data quality enforced pre-publication.
- **Incident Management**: Procedures for data breaches and quality failures.

6.3 Regulatory and Compliance Requirements

- **GDPR**: Data subject rights, consent management, data minimization.
- **SOX/PCI DSS**: Audit trails, segregation of duties, integrity controls.
- **Industry-Specific**: Basel III, MiFID II, HIPAA, etc., as required by deployments.

---

7. Implementation Roadmap

7.1 Phased Approach

Phase 1: Define logical data models and implement data quality validations (in progress).
Phase 2: Extend metadata management, lineage, and structured storage.
Phase 3: Advanced analytics, BI integration, and event-driven processing.
Phase 4: Full regulatory automation and continuous quality monitoring.

7.2 Key Milestones

- Release of core data architecture documentation and models.
- Deployment of metadata registry and enterprise data dictionary.
- Integration of automated quality and profiling tools.
- Regulatory compliance certification and audit readiness.

7.3 Resource Requirements

- Data Architect/Modeler
- Data Stewards and Governance Lead

- Backend Engineers (API & Data)
- QA/Test Automation
- Compliance and Security Officer

7.4 Success Metrics

- 100% coverage of documented data models for all document types.
- <1% error rate on data validation at ingestion and processing.
- Achieve and maintain compliance certifications (GDPR, SOX, etc.).
- Positive user feedback and business adoption.

---

8. Appendices

8.1 Glossary

- **Data Architecture**: The structure of data assets, flows, and management.
- **Data Steward**: Individual responsible for data quality and governance.
- **API-First**: Designing systems such that all functionality is accessible via APIs.

8.2 Reference Models

- DMBOK 2.0 Data Architecture and Quality frameworks.
- ADPA Logical Data Model diagrams.

8.3 Tools and Technologies

- **Validation**: Joi, Zod, AJV (JSON validation)
- **Modeling**: ERD tools, JSON Schema, UML
- **Integration**: Swagger/OpenAPI, TypeSpec, OAuth2 libraries

8.4 Case Studies

- [Reserved for future real-world implementations]

8.5 References

- DAMA-DMBOK 2.0
- PMBOK 7th Edition
- BABOK v3
- ADPA Project Documentation

Prepared by:
ADPA Data Architecture Team
Version: 1.0
Date: [2024-06-XX]

For further details, see ADPA documentation and DMBOK-aligned policies.