Certificate in Master Data Migration

Master Data Management Fundamentals

Attribute – A single piece of data that describes an entity, such as “Customer Name” or “Product Price”. Data Element, Field – Attributes are the building blocks of master records. Example: In a customer master, the attribute “Email Address” stores the contact email. Practical application: Defining attributes ensures consistent data capture across systems. Challenge: Over-defining attributes can lead to redundancy and increased maintenance effort.

Authority – The source or system that is designated as the definitive provider of a specific data element. Source of Truth, Golden Record – An authority is trusted to supply accurate and up-to-date information. Example: The ERP system may be the authority for product pricing. Practical application: Authority rules drive data reconciliation processes during migration. Challenge: Conflicts arise when multiple authorities claim ownership of the same attribute.

Business Rules – Logical conditions that govern how master data is created, updated, or validated. Data Governance, Validation Rules – They enforce consistency and compliance. Example: A rule that “Customer status cannot be ‘Inactive’ if there are open orders”. Practical application: Business rules are embedded in ETL scripts to prevent bad data from entering the target system. Challenge: Keeping rules synchronized with evolving business policies.

Canonical Model – A unified, abstract representation of data that enables interoperability between disparate systems. Enterprise Data Model, Integration Layer – It acts as a common language for data exchange. Example: A canonical model for product data includes standardized categories, units of measure, and identifiers. Practical application: Simplifies data mapping during migration projects. Challenge: Designing a model that accommodates all legacy variations without becoming overly complex.

Change Data Capture (CDC) – A technique that tracks and records data modifications in source systems in near real-time. Incremental Load, Data Replication – CDC supports ongoing synchronization after the initial migration. Example: Capturing every insert, update, and delete on the Customer table to keep the target master up-to-date. Practical application: Reduces migration window by allowing parallel processing of changes. Challenge: Managing CDC latency and ensuring no data loss during high-volume periods.

Cleanse – The process of detecting and correcting inaccurate, incomplete, or inconsistent data. Data Quality, Standardization – Cleansing improves reliability of master data. Example: Normalizing address formats to a standard postal code schema. Practical application: Cleanse scripts are run before loading data into the master repository. Challenge: Determining the appropriate level of cleansing without discarding valuable historical nuances.

Data Governance – The set of policies, procedures, and responsibilities that ensure data is managed as a strategic asset. Stewardship, Ownership – Governance defines who can create, modify, or delete master data. Example: A data governance council approves new product attributes. Practical application:

Governance frameworks guide migration scope and approval workflows. Challenge: Achieving cross-departmental buy-in and maintaining enforcement over time.

Data Integration – The process of combining data from multiple sources to provide a unified view. ETL, Middleware – Integration is essential for consolidating master data. Example: Merging supplier information from procurement, finance, and CRM systems. Practical application: Integration platforms transform and load data into the master repository. Challenge: Handling schema mismatches and differing data quality levels across sources.

Data Lineage – The traceability of data from its origin through all transformations to its final destination. Provenance, Audit Trail – Lineage provides transparency for compliance and debugging. Example: Documenting that a product's "Launch Date" originated from the marketing system, was adjusted by the sales team, and finally stored in the master. Practical application: Lineage diagrams assist auditors in verifying migration integrity. Challenge: Capturing lineage for legacy systems with limited metadata.

Data Migration – The systematic transfer of data from legacy environments to a new target system. ETL, Cutover – Migration includes extraction, transformation, loading, and validation. Example: Moving customer records from a mainframe to a cloud-based CRM. Practical application: A phased migration approach minimizes business disruption. Challenge: Balancing speed with thorough data validation to avoid post-go-live issues.

Data Model – A logical representation of data structures, relationships, and constraints. Entity-Relationship Diagram, Schema – The model guides database design and integration. Example: A star schema for product master data featuring a central fact table and dimension tables. Practical application: Aligning the data model with business processes ensures relevance. Challenge: Updating the model to reflect new business requirements without breaking existing integrations.

Data Quality – The degree to which data is fit for its intended purpose, measured by accuracy, completeness, consistency, and timeliness. Profiling, Cleansing – High quality is crucial for reliable master data. Example: A completeness score of 95 % for mandatory customer fields. Practical application: Data quality dashboards monitor migration progress. Challenge: Establishing realistic quality thresholds and remediation plans.

Data Steward – An individual responsible for the day-to-day management of master data, including validation, issue resolution, and policy enforcement. Ownership, Governance – Stewards act as custodians of data integrity. Example: A product data steward reviews new items for correct categorization. Practical application: Stewards approve changes during migration cutover. Challenge: Allocating sufficient time and authority to stewards across multiple domains.

Data Warehouse – A centralized repository that stores integrated, historical data for reporting and analysis. OLAP, Dimensional Modeling – While not a master data store, it often consumes master data for reference. Example: A sales analytics warehouse uses the product master to enrich transaction data. Practical application: Synchronizing master updates to the warehouse ensures consistent reporting. Challenge: Managing latency between master updates and warehouse refresh cycles.

Data Warehouse – (Duplicate entry removed; see above).

Data Warehouse – (Ensuring no duplication; continue).

Entity – A distinct object or concept about which data is stored, such as “Customer”, “Product”, or “Supplier”. Master Record, Business Object – Entities define the scope of master data. Example: The “Customer” entity contains attributes like name, address, and credit limit. Practical application: Entity definitions drive the design of migration mapping tables. Challenge: Aligning entity definitions across business units that use different terminology.

Entity Relationship Diagram (ERD) – A visual representation of entities, their attributes, and relationships. Data Model, Schema – ERDs help stakeholders understand data structures. Example: An ERD showing a one-to-many relationship between “Customer” and “Order”. Practical application: ERDs are used to validate migration mapping logic. Challenge: Keeping ERDs up-to-date as systems evolve during a multi-year migration.

ETL (Extract, Transform, Load) – The three-step process used to move data from source to target. Data Migration, Integration – ETL tools handle large-scale data processing. Example: Extracting product data from a CSV file, transforming units of measure, and loading into the master repository. Practical application: Scheduling ETL jobs to run during low-usage windows reduces impact on production systems. Challenge: Designing transformations that preserve data lineage and auditability.

Golden Record – The single, authoritative version of a master entity after consolidating duplicates and reconciling conflicts. Master Record, Single Source of Truth – It represents the highest quality data. Example: A unified customer record that merges information from CRM, billing, and support systems. Practical application: Golden records are the target of deduplication routines during migration. Challenge: Defining merge rules that satisfy all stakeholder expectations.

Hierarchy – A structured arrangement of entities that reflects parent-child relationships, such as product categories or organizational units. Tree Structure, Drill-Down – Hierarchies support roll-up reporting. Example: A “Product Category” hierarchy with “Electronics” → “Computers” → “Laptops”. Practical application: Maintaining hierarchies ensures accurate aggregation in downstream analytics. Challenge: Reconciling divergent hierarchies from legacy systems without losing granularity.

Identifier – A unique key that distinguishes each master record, often a numeric or alphanumeric code. Primary Key, Business Key – Identifiers enable reliable linking across systems. Example: A SKU (Stock Keeping Unit) uniquely identifies a product. Practical application: Mapping source identifiers to target identifiers is a core step in migration. Challenge: Handling legacy identifiers that are non-unique or have changed format.

Integration Hub – A centralized platform that facilitates data exchange between multiple applications using a common runtime. Middleware, Service Bus – It streamlines master data distribution. Example: An integration hub routes customer updates from the CRM to the ERP and analytics platforms. Practical application: The hub reduces point-to-point connections, simplifying migration architecture. Challenge:

Ensuring the hub scales to high transaction volumes during cutover.

Informatica PowerCenter – A widely used ETL tool that provides data integration, transformation, and quality functionalities. ETL, Data Migration – It offers built-in profiling and cleansing modules. Example: Using PowerCenter to extract supplier data, apply standardization, and load into the master repository. Practical application: Leveraging its metadata repository to document lineage for compliance. Challenge: Licensing costs and the learning curve for complex mappings.

Job Scheduling – The process of automating the execution of ETL, cleansing, and validation tasks at predefined times. Batch Processing, Workflow – Scheduling ensures orderly migration phases. Example: Scheduling nightly loads of incremental customer changes. Practical application: Coordinating job schedules with business windows minimizes disruption. Challenge: Handling job failures and cascading dependencies in a tightly timed cutover.

Latency – The delay between a data change in the source system and its reflection in the target master. CDC, Real-Time Sync – Low latency is critical for operational master data. Example: A 5-minute latency for price updates ensures sales teams see current values. Practical application: Monitoring latency metrics during migration validates performance targets. Challenge: Network constraints and batch processing can increase latency beyond acceptable limits.

Logical Data Model (LDM) – An abstract representation of data entities, attributes, and relationships without physical storage details. Conceptual Model, Data Model – LDM bridges business requirements and technical implementation. Example: An LDM for the “Supplier” entity includes attributes like “Tax ID” and “Bank Account”. Practical application: LDMs guide the creation of database tables and integration mappings. Challenge: Keeping the LDM synchronized with evolving business rules throughout the migration lifecycle.

Master Data – Core, non-transactional data that is shared across multiple business processes, such as customers, products, and locations. Reference Data, MDM – Master data is the focus of governance and migration. Example: The product master includes SKU, description, and weight. Practical application: Consolidating master data reduces redundancy and improves data quality. Challenge: Identifying all sources of master data in a fragmented landscape.

Master Data Management (MDM) – A discipline and set of technologies that create and maintain a single, consistent view of master data across the enterprise. Data Governance, Golden Record – MDM enforces standards, stewardship, and synchronization. Example: An MDM hub reconciles customer records from CRM, e-commerce, and billing. Practical application: MDM serves as the authoritative source during and after migration. Challenge: Integrating MDM with legacy systems that lack modern APIs.

Metadata – Data that describes other data, including definitions, lineage, data types, and usage contexts. Data Catalog, Documentation – Metadata supports understanding and governance. Example: Metadata indicating that “Customer Birthdate” is stored in YYYY-MM-DD format. Practical application: Metadata repositories are consulted when mapping source fields to target attributes. Challenge: Incomplete or outdated metadata hampers accurate migration planning.

Normalization – The process of organizing data to reduce redundancy and improve integrity, often by separating repeating groups into related tables. Data Modeling, De-Duplication – Normalization is essential for relational databases. Example: Moving address information into a separate "Address" table linked by a foreign key. Practical application: Normalized structures simplify updates to shared attributes during migration. Challenge: Over-normalization can degrade performance for reporting workloads.

Object-Relational Mapping (ORM) – A technique that maps objects in application code to relational database tables. Data Access Layer, Integration – ORMs can affect migration by abstracting data structures. Example: An ORM layer translates "Product" objects to rows in the "PRODUCTS" table. Practical application: Understanding ORM mappings helps identify hidden data transformations during migration. Challenge: ORM caches may retain stale data, leading to inconsistencies post-migration.

Operational Data Store (ODS) – A database designed to hold current, integrated data for operational reporting and short-term analysis. Staging Area, Real-Time Integration – ODS often receives near-real-time feeds from source systems. Example: An ODS consolidates daily customer updates before they are loaded into the master repository. Practical application: Using an ODS as a staging layer smooths data flow during migration cutover. Challenge: Managing data freshness and ensuring ODS does not become a bottleneck.

Outlier Detection – Techniques used to identify data points that deviate significantly from expected patterns. Data Quality, Profiling – Outliers may indicate errors or exceptional cases. Example: Detecting a product weight of "10 000 kg" when typical weights range from 0.1 To 5 kg. Practical application: Flagging outliers for review before loading into the master. Challenge: Distinguishing true anomalies from legitimate extreme values.

Parallel Load – Loading data into the target system using multiple concurrent processes to accelerate migration throughput. Bulk Load, Performance Tuning – Parallelism reduces overall migration time. Example: Splitting a 10 million-record customer file into four streams processed simultaneously. Practical application: Configuring parallel threads in the ETL tool to match target system capacity. Challenge: Managing transaction conflicts and ensuring order-independent data integrity.

Pivot Table – A data summarization tool that aggregates and reorganizes data for analysis, often used to validate migration results. Reporting, Data Profiling – Pivot tables quickly reveal discrepancies. Example: Creating a pivot to compare record counts by region before and after migration. Practical application: Stakeholders use pivots to verify data completeness. Challenge: Large datasets may cause performance issues in spreadsheet tools.

Primary Key – A field or combination of fields that uniquely identifies a record within a table. Identifier, Constraint – Primary keys enforce entity integrity. Example: "Customer_ID" as the primary key in the Customer table. Practical application: Mapping primary keys from source to target ensures referential integrity during migration. Challenge: Legacy systems may lack explicit primary keys, requiring surrogate key creation.

Profiling – The systematic analysis of data to assess its quality, structure, and content patterns. Data Quality, Assessment – Profiling informs cleansing and transformation decisions. Example: Running a profile that

shows 12 % of product records have missing “Release Date”. Practical application: Profiling reports guide prioritization of data remediation tasks. Challenge: Profiling large volumes can be resource-intensive and may miss subtle inconsistencies.

Reference Data – Static or slowly changing data that classifies or categorizes master data, such as country codes, currency lists, or industry classifications. Lookup Tables, Master Data – Reference data is essential for validation. Example: ISO 3166 country codes used in customer addresses. Practical application: Maintaining synchronized reference tables across systems prevents mismatches. Challenge: Aligning differing reference standards from multiple legacy applications.

Replication – The process of copying data from one system to another to ensure consistency and availability. CDC, Synchronization – Replication can be uni- or bi-directional. Example: Replicating product master updates from the MDM hub to the e-commerce platform. Practical application: Replication keeps downstream systems current during migration. Challenge: Conflict resolution when concurrent updates occur in both source and target.

Rollback – A contingency operation that restores the system to its pre-migration state if critical failures occur. Recovery, Cutover Plan – Rollback plans are essential for risk mitigation. Example: Restoring the previous customer database snapshot after a corrupted load. Practical application: Maintaining backup copies and transaction logs enables swift rollback. Challenge: Ensuring rollback procedures are tested and that data consistency is preserved across all integrated systems.

Schema – The structural definition of a database, including tables, columns, data types, and constraints. Data Model, DDL – Schemas dictate how data is stored and accessed. Example: The “Product” schema defines fields such as SKU, Name, and Price. Practical application: Schema comparison tools identify differences between source and target structures before migration. Challenge: Reconciling schema mismatches without extensive re-engineering.

Security – Controls and policies that protect data from unauthorized access, alteration, or loss. Encryption, Access Control – Security measures must be maintained throughout migration. Example: Encrypting customer PII during transit between source and target. Practical application: Role-based access ensures only authorized personnel can approve master data changes. Challenge: Balancing stringent security with the need for rapid data movement during cutover.

Service-Oriented Architecture (SOA) – An architectural style that uses loosely coupled services to enable interoperability. Integration Hub, APIs – SOA facilitates modular data exchange. Example: Exposing a “GetCustomer” service that returns master data in XML. Practical application: Leveraging SOA services to pull master data on demand during migration validation. Challenge: Legacy systems may lack service endpoints, requiring wrapper development.

Source System – The original application or database that holds data to be migrated. Legacy System, Extraction – Understanding source structures is the first step in migration planning. Example: An on-premise mainframe holding vendor master records. Practical application: Conducting source system audits to catalogue tables, fields, and data volumes. Challenge: Dealing with undocumented customizations and lack

of technical support.

Staging Area – A temporary storage location used to hold extracted data before transformation and loading. ODS, Temporary Tables – Staging isolates raw data from production environments. Example: Loading CSV extracts into a SQL staging table for cleansing. Practical application: Staging enables bulk validation and error handling without impacting source systems. Challenge: Ensuring sufficient capacity and security for sensitive data in the staging environment.

Surrogate Key – An artificially generated identifier, often numeric, used as a primary key when natural keys are unsuitable. Identifier, Mapping – Surrogate keys simplify joins across heterogeneous sources. Example: Assigning a sequential "Customer_Surrogate_ID" during migration. Practical application: Mapping source keys to surrogate keys preserves relationships while standardizing identifiers. Challenge: Maintaining traceability back to original business keys for audit purposes.

Synchronization – Ongoing alignment of master data between multiple systems to keep them consistent. Replication, CDC – Synchronization can be real-time or batch-driven. Example: Syncing product pricing updates from the ERP to the sales portal nightly. Practical application: Post-migration, synchronization ensures that downstream systems receive the latest master changes. Challenge: Handling synchronization conflicts when two systems modify the same record simultaneously.

Target System – The destination platform where master data will reside after migration. MDM Hub, Data Warehouse – Target design influences mapping and transformation logic. Example: A cloud-based MDM solution that will serve as the new product master. Practical application: Configuring target data structures to accept incoming records without loss. Challenge: Aligning target capabilities with legacy data complexities, especially for custom fields.

Test-Data-Set – A representative subset of data used to validate migration processes before full production runs. Pilot, Validation – Testing reduces risk by uncovering issues early. Example: Using 5 % of customer records to verify transformation rules. Practical application: Running end-to-end test migrations to confirm data integrity and performance. Challenge: Ensuring the test set captures edge cases and data diversity.

Transformation – The set of operations applied to source data to conform to target standards, including cleansing, mapping, and enrichment. ETL, Business Rules – Transformations are central to migration success. Example: Converting dates from "DD/MM/YYYY" to ISO "YYYY-MM-DD". Practical application: Defining transformation scripts in the ETL tool and documenting each rule. Challenge: Managing complex transformations that involve multiple dependent attributes.

Trusted Data Source – A system or dataset recognized as reliable for a specific attribute or entity. Authority, Golden Record – Trust is established through governance and historical performance. Example: The finance system is the trusted source for "Credit Limit". Practical application: Authority hierarchies dictate which source overrides others during conflict resolution. Challenge: Maintaining trust when source systems undergo upgrades or data model changes.

Unified Data Model (UDM) – A comprehensive model that integrates multiple domain models into a single,

coherent structure. Canonical Model, Enterprise Data Model – UDM supports cross-domain analytics. Example: Combining customer, product, and supplier models into one unified view. Practical application: UDM serves as the blueprint for migration mapping across domains. Challenge: Balancing the need for a common model with domain-specific nuances.

Validation – The process of checking that data meets predefined rules and constraints before it is accepted into the target system. Testing, Data Quality – Validation ensures data integrity. Example: Verifying that every product has a non-null SKU before loading. Practical application: Automated validation scripts generate error reports for remediation. Challenge: Designing comprehensive validation without causing excessive load on the system.

Versioning – Maintaining multiple iterations of master data to track changes over time. Audit Trail, Change Management – Versioning supports rollback and historical analysis. Example: Keeping a version history of product specifications for regulatory compliance. Practical application: MDM platforms often provide built-in version control for master records. Challenge: Managing storage growth and ensuring users access the correct version.

Workflow – A defined sequence of tasks, approvals, and notifications that govern how master data is created or updated. Process Automation, Governance – Workflows enforce business rules. Example: A new supplier onboarding workflow that requires finance approval before the record becomes active. Practical application: Configuring workflow engines to route change requests during migration. Challenge: Over-engineering workflows can slow down data entry and increase user resistance.

XML (eXtensible Markup Language) – A flexible text format for representing structured data, often used for data exchange between systems. Data Integration, APIs – XML facilitates interoperability. Example: Exporting customer records as XML for import into the MDM hub. Practical application: Defining XSD schemas ensures that exchanged data conforms to expected structures. Challenge: Large XML files can be memory-intensive; parsing performance must be monitored.

YAML (YAML Ain’t Markup Language) – A human-readable data serialization format, frequently used for configuration files and lightweight data exchange. Configuration, Metadata – YAML is concise and easy to edit. Example: Storing ETL job parameters in a YAML file for version control. Practical application: Using YAML to define mapping rules that can be reviewed by business analysts. Challenge: Strict indentation rules can cause parsing errors if not carefully managed.

Zero-Downtime Migration – A strategy that aims to move data without interrupting business operations, often by running parallel systems. Blue-Green Deployment, Cutover – Zero-downtime minimizes revenue impact. Example: Deploying a new MDM hub while the legacy system continues to serve requests, then gradually shifting traffic. Practical application: Incremental data syncs keep both systems aligned until the final cutover. Challenge: Complex coordination and increased infrastructure costs to support dual environments.