

Error Codes and Handling Procedures for MDM Data Quality Issues

Reference Document | Model: BDWP-008

Table of Contents

- 1. Introduction
- 2. Scope and Purpose
- 3. Error Codes Overview
- 4. Common Error Codes
- 5. Error Code Details
 - 5.1 MDM-ID-3102: Identifier conflict
 - 5.2 MDM-LIN-2207: Lineage gap
 - 5.3 MDM-COM-1905: Commentary required
- 6. Diagnosis and Troubleshooting Procedures
- 7. Validation Checklists
- 8. Escalation Procedures
- 9. Resolution Checkpoints
- 10. Modules Affected & Examples
- 11. Appendices

1. Introduction

This document provides comprehensive guidance on error codes pertinent to Master Data Management (MDM) data quality issues within the enterprise data infrastructure. It aims to serve data stewards, MDM administrators, and system integrators by clearly outlining error symptoms, root causes,

troubleshooting steps, escalation channels, and resolution checkpoints. This resource facilitates consistent and effective resolution of data issues, thereby enhancing overall data integrity and operational efficiency.

2. Scope and Purpose

The scope includes key error codes encountered during MDM operations, specifically those affecting data integrity, lineage, and annotations. The document encompasses common error scenarios including identifier conflicts, lineage discrepancies, and commentary requirements across modules such as ADR mapping, ETF reconciliation, and vendor taxonomy updates. The purpose is to standardize troubleshooting workflows, define escalation paths, and provide validation checklists for efficient issue resolution aligned with enterprise policies.

3. Error Codes Overview

Within the MDM environment, error codes are generated when data issues violate validation rules, integrity constraints, or business policies. These codes are categorized based on their nature, such as identifier conflicts, lineage discrepancies, or data annotations requiring input. Proper interpretation of these error codes is essential to diagnose underlying issues rapidly, apply corrective actions, and prevent recurrence. The following sections detail specific error codes relevant to our collection.

4. Common Error Codes

Error Code	Description	Modules Affected	Typical Scenario
MDM-ID-3102	Identifier conflict detected when duplicate or inconsistent IDs are present	ADR mapping, Entity reconciliation	Two assets inadvertently assigned the same unique identifier during data loading

MDM-LIN-2207	Lineage gap indicating missing parent or lineage information for an entity	Lineage tracking, Data lineage modules
MDM-COM-1905	Commentary or annotation required for data approval or compliance reasons	Data stewardship modules, Metadata annotations

5. Error Code Details

5.1 MDM-ID-3102: Identifier conflict

Symptoms: When attempting data ingestion or synchronization, systems flag duplicate identifiers for distinct entities, preventing proper linkage.

Root Causes:

- Multiple data sources assigning the same ID to different assets
- Erroneous manual entries or outdated data loads
- Synchronization timing issues causing ID overlaps

Troubleshooting Procedure:

1. Verify the source of the conflicting IDs within the system logs or data load reports.
2. Check the master identifier records for duplicates using the data validation tools.
3. Query the data repository for all entities sharing the same ID:

```
SELECT * FROM assets WHERE identifier = 'conflicting_id';
```

- Identify the differences or discrepancies between the entities with the same ID.
- Cross-reference with source systems to determine which record is correct or needs updating.
- Apply deduplication or correction procedures, such as updating or merging records.
- Re-run the validation to confirm conflict resolution.

Prevention Tips: Implement automated duplicate detection rules on data entry forms and load processes. Regularly audit identifiers across sources.

5.2 MDM-LIN-2207: Lineage gap

Symptoms: Missing lineage links when tracing asset origination or dependencies, leading to incomplete data lineage diagrams.

Root Causes:

- Incomplete lineage data entry or omission during data updates
- Incorrect parent-child relationships in data loads
- Failures in lineage reconciliation processes

Troubleshooting Procedure:

1. Check the lineage metadata fields for the affected entity.
2. Compare the current lineage graph against source documentation or previous snapshots.
3. Identify missing parent or child references.
4. Verify data load logs for errors or skipped entries related to lineage.
5. If a lineage gap exists, reprocess or manually correct the lineage references:

```
UPDATE lineage SET parent_id = 'correct_parent_id' WHERE entity.
```

- Validate that lineage is now complete and correctly linked.

Prevention Tips: Enforce mandatory lineage fields in data entry validation rules and perform periodic lineage audits.

5.3 MDM-COM-1905: Commentary required

Symptoms: Data cannot be approved or published until mandatory comments or annotations are provided.

Root Causes:

- Pending review comments from data stewards
- Automated validation rules requiring annotations before proceeding
- Incomplete data submission with missing commentary

Troubleshooting Procedure:

1. Identify the data record flagged with the error code.
2. Review system prompts or validation logs to determine missing comments.

3. Communicate with the responsible steward or user to obtain or add required commentary.
4. Insert the comments or annotations into the appropriate metadata fields.
5. Re-submit for validation or approval.

Prevention Tips: Enforce mandatory comments in data entry workflows and provide user training on proper documentation requirements.

6. Diagnosis and Troubleshooting Procedures

For each error code, follow these structured procedures to diagnose and resolve issues efficiently. The steps include initial assessment, root cause analysis, corrective actions, and verification.

1. **Gather initial data:** Review system logs, validation reports, and data load histories associated with the error.
2. **Analyze symptoms:** Identify patterns, affected modules, and data anomalies.
3. **Identify root cause:** Use diagnostic queries, comparison reports, and data reconciliation tools.
4. **Implement corrective measures:** Correct data discrepancies, update lineage, or add missing comments as required.
5. **Verify resolution:** Re-run validation routines, check for recurrence, and document outcomes.

Maintain detailed logs at each step for audit trail and future reference.

7. Validation Checklists

7.1 Identifier Conflict Resolution Checklist

- Confirm duplicate identifiers across datasets or modules
- Identify source of duplication or mismatch
- Decide on record merge, update, or deactivation
- Perform data correction or deduplication
- Validate the correction through system checks

7.2 Lineage Gap Resolution Checklist

- Review lineage records for completeness
- Identify missing parent or child references
- Reprocess lineage data load if necessary
- Apply manual corrections where automation fails

- Confirm lineage integrity with visualization tools

7.3 Commentary Requirement Checklist

- Check for missing or incomplete comments
- Communicate with data stewards for missing information
- Insert or update comments in metadata fields
- Re-validate that commentary is now complete

8. Escalation Procedures

When troubleshooting steps do not resolve the issues, escalate to higher support tiers following the enterprise escalation matrix:

- **Level 1:** Data Steward / Data Analyst — For immediate data corrections and clarification.
- **Level 2:** MDM Operations Support — For systemic or recurring issues requiring technical intervention.
- **Level 3:** Data Governance or Data Quality Office — For policy violations, complex root causes, or authorization needs.

Proper incident documentation, including error codes, steps taken, and outcomes, should accompany escalation requests.

9. Resolution Checkpoints

Ensure the following points are verified before closing a troubleshooting case:

- The specific error code is no longer generated during data processing.
- Data integrity is restored with no residual conflicts or gaps.
- Lineage and annotations are complete and accurate.
- Validation reports reflect a clean status.
- Stakeholder verification or sign-off is obtained if necessary.

Maintain documented confirmation for audit purposes.

10. Modules Affected & Examples

The following modules are impacted by these error codes, with typical scenarios and corrections:

10.1 ADR Mapping Module

- Issue: Identifier conflicts during ADR data import.

- Example: Duplicate CUSIP codes assigned to different ADRs.
- Resolution: Deduplicate entries, update conflicting records, and re-import.

10.2 ETF Constituent Reconciliation

- Issue: Lineage gaps leading to missing ETF component data.
- Example: Missing parent ETF entity for a specific component asset.
- Resolution: Recreate lineage links, verify parent-child relationships, and re-validate.

10.3 Vendor Taxonomy Updates

- Issue: Missing commentary or approval flags in taxonomy updates.
- Example: Need for commentary before publishing Bloomberg taxonomy changes.
- Resolution: Obtain required comments, update records, and rerun validations.

11. Appendices

Appendix A: Relevant Data Validation Rules

Sample rules include:

- Unique entity identifiers must be non-null and exclusive across sources.
- Lineage references must exist for each child asset, referencing a valid parent.
- Annotations or comments are mandatory for all approved data changes.

Appendix B: Linkages to External Documentation

- [Vendor Taxonomy Update Procedures](#)
- [Moody's Sector Specification Guidelines](#)
- [FactSet Concordance Rules Documentation](#)
- [Steward Override Approval Matrices](#)
- [Lineage Documentation Templates](#)