Master Data Quality Improvement and Recovery Timeline after Vendor Schema Changes

Category: Guides | **Model:** BDWP-010

Description: This guide provides a timeline and step-by-step approach for data quality improvement and recovery following significant vendor schema changes, such as Bloomberg's 2025-08-19 update. It covers initial impact assessment, resolution of conflicts, validation procedures, and normalizing metrics, to help teams systematically restore data integrity and document the process.

Table of Contents

1. Introduction

This document serves as a comprehensive guide for Data Management teams tasked with addressing data quality issues that arise due to vendor schema updates, particularly focusing on Bloomberg's update on August 19, 2025. The purpose is to establish a clear, systematic timeline and procedural framework to restore data integrity, reconcile conflicts, and validate normal operations following schema modifications.

This guide also aligns with policy adherence, quality assurance standards, and best practices for data lineage, steward approvals, and metadata management, ensuring sustainable data governance.

2. Vendor Schema Change Overview

2.1. Nature of the Changes

The Bloomberg schema update of August 19, 2025, introduced several structural changes, including:

- Suffix pattern modifications for identifiers, e.g., `.LN`, `.GR`, `.HK`.
- New data fields added to vendor data feeds and existing fields renamed or deprecated.
- Updates to taxonomy and classification codes, such as ISIN, SEDOL, and CUSIP recalibrations.

2.2. Affected Modules

Module	Impact Description
ADR Mapping	Changes to ADR identification suffixes and embedded attributes.
ETF Constituent Reconciliation	New classification codes affecting reconciliation processes.
FactSet Concordance Rules	Updated rules for CUSIP, ISIN, SEDOL matching; refer to section 7 for rules details.
Steward Override Procedures	Adjustments required for approval matrices to accommodate new schema patterns.)

2.3. Data Lineage Considerations

All affected modules require updated lineage documentation to track the origin and transformation of new fields and modified identifiers, ensuring auditability and compliance with governance policies.

3. Impact Assessment and Initial Analysis

3.1. Prerequisites

- Access to raw vendor data feeds pre- and post-update.
- Existing data lineage maps and metadata repositories.
- Notification from vendor regarding schema change release notes.

3.2. Step-by-Step Impact Assessment

1. **Identify affected data fields:** Use schema diff tools or version comparison scripts to detect changes in structure and data types.
2. **Map affected modules:** Cross-reference impacted processes and systems, such as ADR, ETFs, and reference data stores.
3. **Generate initial impact report:** Document affected data flows, potential conflicts, and criticality levels.
4. **Engage stakeholders:** Notify Data Stewards, Data Owners, and Vendor Relations teams for collaborative assessment.

3.3. Example Impact Assessment Report Outline

- Impact Summary Table
- Affected Data Fields and Modules
- Potential Data Conflicts Identified
- Recommended Immediate Actions

Sample snippet of impact report:

```
Impact Summary:
- Affected Modules: ETF Reconciliation, ADR Mapping, FactSet I
- Key Changes Detected:
  - CUSIP suffixes now include '.HK', '.LN', '.GR'
  - SEDOL new fields for market codes
- Estimated Data Conflict Risk: High
- Required Actions: Schema updates, reconciliation rule revisi
```

4. Resolution of Data Conflicts and

Conflict Management Procedures

4.1. Conflict Types

Conflict Type	Description
MDM-ID-3102	Identifier conflict arising from duplicate or inconsistent Vendor IDs after schema change.
MDM-LIN-2207	Lineage gap where previous lineage records are incomplete or inconsistent with current data.
MDM-COM-1905	Commentary required pending resolution or clarification from vendor or data steward.

4.2. Step-by-Step Conflict Resolution Workflow

1. **Identify conflict:** Use Data Quality dashboards and exception reports, focusing on error codes.
2. **Root Cause Analysis:** Trace back affected data points to source files, feed configurations, or transformation logic.
3. **Engage Stewards and Vendor Contacts:** Escalate unresolved conflicts for decision approval, referencing the approval matrix.
4. **Apply Resolution:** Use steward override procedures or vendor corrections to rectify conflicts.
5. **Document Resolution:** Record the conflict, resolution steps, date, and responsible personnel in the lineage tracking system.

4.3. Conflict Resolution Examples

- **Example 1:** Duplicate CUSIP with different suffix—resolve by standardizing suffixes via reconciliation rules.
- **Example 2:** Missing lineage data for a security—trace lineage back to raw vendor source and update lineage records accordingly.

5. Validation and Verification Procedures

5.1. Validation Checklist Preparation

Use comprehensive checklists that include:

- Field-level validation for new vs. old data formats.
- Consistency checks across modules (e.g., ETF shares vs. constituent data).
- Reference data verification (e.g., SEDOL to Bloomberg classification).

5.2. Step-by-Step Validation Process

1. **Data Load Validation:** Execute automated scripts to verify key fields match expected formats and values.
2. **Reconciliation Validation:** Compare pre- and post-change reconciliation reports to identify discrepancies.
3. **Lineage Validation:** Confirm data lineage links are intact and accurate.
4. **Vendor Metadata Validation:** Cross-verify vendor metadata against the latest release notes.

5.3. Verification Output Examples

Sample validation report section:

```
Validation Results:
- Total records checked: 1,200,000
- Records with identifier suffix conflicts: 3,245
- Discrepancies in ETF constituent counts: 8,920
- Lineage completeness: 99.5%
- Resolution status: Pending final reconciliation
```

6. Normalization of Data Metrics After Schema Changes

6.1. Objectives

- Standardize data reporting metrics across affected modules.
- Ensure comparability of data pre- and post-vendor update.
- Establish normalized metrics for ongoing monitoring.

6.2. Methodology

1. Identify key metrics impacted, e.g., number of Suffix conflicts, lineage gaps, validation errors.
2. Calculate baseline metrics before change for reference.

3. Apply normalization formulas considering the expected schema alterations, for example:

```
Normalized Conflict Rate = (Post-change conflicts / Total
```

4. Document normalized metrics and update dashboards accordingly.

6.3. Example Metrics

Metric	Pre-Change	Post-Change	Normalized Value
CUSIP suffix conflicts	150	5,350	Adjusted considering new suffix patterns
Lineage gaps	1,200	250	Standardized to current data volume
Validation errors	2,500	11,200	Corrected through reconciliation process

7. Standard Operating Procedures (SOPs)

7.1. Schema Update Handling SOP

1. Receive vendor notification and release notes.
2. Broadcast internal alerts to impacted data teams.
3. Implement schema change in staging environment.
4. Run impact analysis and conflict detection scripts.
5. Update reconciliation and mapping rules.
6. Perform validation and verification steps.
7. Deploy updates to production after approvals.

7.2. Data Reconciliation SOP

1. Extract raw data before and after changes.
2. Compare key fields and identifiers.

3. Flag discrepancies exceeding thresholds.
4. Engage data stewards for resolution.
5. Document reconciliation outcomes.

7.3. Data Steward Override Procedure

1. Identify conflict and notify steward.
2. Review conflict details with supporting lineage and metadata.
3. Use override matrix to determine approval level.
4. Apply steward override with documented comments.
5. Audit override decisions periodically.

8. Error Codes and Troubleshooting

8.1. Common Error Codes

Error Code	Symptoms	Root Cause	Resolution
MDM-ID-3102	Duplicate identifiers detected during reconciliation.		
MDM-LIN-2207	Lineage inconsistency alerts.		
MDM-COM-1905	Comments pending or missing for conflicting data.		

8.2. Troubleshooting Workflow

1. Identify error code from logs or exception reports.
2. Consult the error code documentation for details.
3. Check recent schema change logs and impact reports.
4. Perform targeted data validation or reconciliation scripts.
5. Engage relevant data stewards or vendor contacts if unresolved.
6. Record the troubleshooting outcome and update procedures if needed.

8.3. Error Prevention Tips

- Regularly update validation scripts to accommodate schema changes.
- Implement automated alerts for key error codes.
- Maintain up-to-date lineage and metadata documentation.
- Conduct periodic reconciliation and audit checks.

9. Validation Checklists and Examples

9.1. Data Load Validation Checklist

- Verify new identifier suffixes are correctly assigned.
- Ensure no duplicate identifiers post-load.
- Check for missing or null critical fields.
- Confirm compliance with updated taxonomy standards.
- Validate counts and totals against baseline.

9.2. Reconciliation Validation Checks

- Compare pre- and post-change aggregate totals.
- Verify correct mapping of new identifier suffixes.
- Confirm lineage links are complete and accurate.
- Review exceptions flagged during reconciliation.

9.3. Sample Reconciliation Checklist Table

Check Item	Status	Notes
Identifier variant consistency	[ ]	
Lineage links present	[ ]	
Data counts match expectations	[ ]	
Validation errors resolved	[ ]	

10. Lineage Documentation and Steward Procedures

10.1. Updating Lineage Records

1. Trace affected data elements back to source feeds.
2. Record transformation logic, including new suffix patterns and classification codes.
3. Update lineage diagrams and metadata repositories with timestamps and responsible stewards.

   </