



Path to Success for Party Domain

Version 1.4

CONFIDENTIALITY LEVEL:

Confidential

Contents

1	Objective.....	3
2	General Considerations.....	3
3	Understanding Business Challenges	3
4	Early Access to Data	4
5	Matching Considerations	5
6	Match Tuning	7
7	Data Model Best Practices.....	7
8	JavaScript for Matching	8
9	Web UI Considerations.....	9
10	Integration Considerations.....	9
11	User Privileges	10
12	Third-Party Services.....	10
13	Troubleshooting Tips	12
14	General Performance Considerations	13
15	CMDM Enablement Team	13

1 Objective

This document is intended to provide high-level best practices and recommendations for practitioners who are embarking on a party domain MDM implementation with Stibo Systems.

2 General Considerations

As Party-domain implementations begin, several considerations will ensure a smooth start to the project, including:

- Understanding the customer's current business challenges.
- Articulating and assisting to formulate their short-term and long-term MDM vision.

The recommendations in this document are focused on driving an efficient Stibo Systems party domain implementation as well as promoting a foundational understanding and alignment with the customer.

3 Understanding Business Challenges

Understanding the existing business challenges at a fundamental level enables identifying new use cases and positively influences a customer's MDM journey. The following are discussion points to consider:

- Understand the customer's primary business and industry to help pinpoint common industry-specific challenges and identify recommendations.
- Familiarize yourself with the customer's primary stakeholders and the internal organization(s) that are driving the MDM initiative.
 - With new initiatives, it is paramount to know the expectations of the stakeholder(s) as well as evangelize participation from key business units within the organization.
 - As with any enterprise initiative, the IT and Business teams must align on expectations and execution.
- Articulate the customer's short-term and long-term MDM vision.
 - The MDM vision contributes to identifying the implementation strategy (Consolidation, Coexistence, Centralized, or Registry)
- Understand the current challenges facing the business and business users.
 - A foundational understanding of the challenges and allows addressing of the problems at the root rather than focusing on complex solutions that may not directly address the challenges.
- Document clearly and agree on the business goals and functional requirements.
 - Translating business challenges into functional requirements allows for zeroing in on specific use cases and understanding how to convey best practices to enhance user experience.
 - This activity promotes alignment in scope across all parties.

- Align with the customer and/or partner on delivery scope.
 - This includes alignment on the customer's implementation strategy.
- Identify the primary users of MDM and the activities they are expected to carry out.
 - This assists in defining the relevant user journeys within MDM. Business processes are often complex and involve many internal stakeholders working collaboratively.
- Familiarize yourself with Stibo Systems' Implementation Methodology Approach (SIMA).
 - SIMA is aimed at executing projects and delivering solutions.
 - SIMA incorporates our expert knowledge and best practices to drive a common approach for Stibo Systems practitioners globally.
 - SIMA addresses both technical and organizational aspects of MDM initiatives.

4 Early Access to Data

Understanding data volumes and gaining access to real data is crucial in the ability to assess and understand the customer's data. Such activities should be included in the initial project plan with proper allocation of time and resource. Furthermore, having access to data is extremely relevant in identifying and fine-tuning your match strategy.

Gaining early access to data allows for:

- Profiling and analyzing to understand the quality of the data and other metrics, such as completeness, uniqueness, etc.
- Gauging the uniqueness of identifiers in source systems and harmonizing values within and across systems, i.e., aligning LOV values.
- Understanding address quality and cardinalities of relationships, etc.

Early access to data impacts areas of solution design, such as data model, matching strategy, information flows, and business processes; all of which are to be accounted for by the project plan.

- For match tuning, it is recommended that:
 - The project plan includes the task and estimated effort to obtain real data as soon as possible.
 - There is a clear understanding of the state and quality of the data.
 - Determine if the customer has a thorough understanding of their current-state data.
 - In addition to pre-existing processes and tools, use the STEP data profiling capabilities to help gain insight.
- For initial data load and subsequent data migration activities:
 - Clearly articulate the volume of data both for initial data migration and year-over-year growth.

- To facilitate initial data migration activities, it is recommended to plan for data load rehearsals and exercises. It is likely that such exercises will reveal complexities that must be accounted for in the solution. This also allows for analysis of application performance and resolves any bottlenecks.
- Data migration exercises should not be carried out in sandbox environments since the infrastructure is not sized to match production environments.
- It is important to understand where data is coming from and what the intent of the data is within MDM.
 - What source system(s) does the data come from?
 - What system(s) are the owner of the data?
 - How will the data be manipulated within STEP?
 - What is the use or role of the data within STEP?
 - Where will the data be delivered?
- For data migration activities, data cleansing exercises often require planning and careful consideration to cleanse data before migration, as part of migration, or post go-live.
 - If using tools such as D&B or Loqate to assist in data cleansing, ensure the respective license allows for multiple iterations / volumes to be accounted for.

5 Matching Considerations

With a proper Matching strategy, organizations can consolidate their data from a variety of source systems and generate singular, definitive, golden records.

For more information on matching configurations, see the **Configuring Matching Algorithms and Match Codes** topic of the **Customer & Supplier MDM Implementation Guidelines** in the **Solution Enablement** section of online help documentation.

- Prior to considering a matching configuration approach, a data cleansing exercise is recommended. Having cleansed and normalized data lends to a better-performing matching solution.
 - Before creating a Matching configuration, it is important to understand the customer organization's data and identify challenges for which the algorithm must account. With this knowledge, a clear match strategy can be defined.
- Once a strategy is defined, the Matching Algorithm is typically configured first to identify what data points specific to the customer organization will provide reliable matching. Following this, define the attribute combinations that will contribute to the match codes.
 - Match codes should be defined so the size of match code groups is minimized (recommend < 100). Large match code groups likely mean more comparisons are being made than necessary, thus adding to processing time.

- For optimal performance, consolidate the matching logic into as few algorithms as possible. It is not recommended to have multiple matching algorithms for the same entity running at once.
- For B2C Householding solutions, leverage the Match & Link strategy. In this approach, the member records are individual customer golden records which are linked to the household entity.
- For Unmerge, prior to data load, the component model for 'Matching – Merge Golden Record' must be properly configured to leverage traceability.
- Carefully consider how upstream or downstream systems are to process the merge and/or unmerge actions from MDM.
- Deduplicate Contact Persons within the organization / supplier they are referenced to since contact information (email, phone, mailing address) often changes as individuals move from one place of employment to another.
- For optimal event processor performance, all triggering attributes should be internally maintained. Externally maintained attributes contribute to longer processing times.
- Ensure the relevant matching component model(s) have been properly configured.
 - Matching Component Model
 - Matching - Link Golden Record Component Model
 - Matching - Merge Golden Record Component Model
- For issues with records not being matched appropriately:
 - Determine if match codes are being generated / refreshed.
 - Verify each algorithm has been properly configured, using the system-provided visual indicator of errors in the configuration.
 - Confirm that match rules accommodate specific data points and/or match scenarios, i.e., unique identifiers automatically result in a high match score.
 - Check that the match event processor is enabled and running.
- If data is not present on the resulting golden record:
 - Determine if the missing data is properly mapped in the import configuration.
 - Determine if the missing data is accounted for in survivorship rules.
 - Check that match codes are being generated.
- Consider updating Transformation Lookup Tables to account for data samples specific to the customer.

6 Match Tuning

Match tuning is an iterative process and a necessary activity to ensure the matching algorithm is optimized for the customer's data set. The goal of this optimization is to produce the optimal matches within each of the auto-merge, clerical review, and reject thresholds that is acceptable to the business.

Failure to adequately match tune may result in a poor user experience and negatively impact the business and analytics with false positives and/or an unnecessary volume of manual clerical review tasks.

For more information, see the **Match Tuning** topic in the **Matching, Linking, and Merging** section of online help documentation.

The following are further considerations:

- Verify that match tuning been accounted for in the implementation project plan.
- Solicit customer stakeholders to actively participate in reviewing the match tuning results.
 - It is important to align with stakeholders on the match strategy, including how to handle false-positive and false-negative results.
- Carry out match tuning using real customer data.
 - The general recommendation is tuning against ~20% of total data volume.
 - For optimal performance, it is recommended to match tune against data sets of no more than 1 million records at a time.
 - Since obtaining such a data extract often requires some effort, it is recommended to request this early in the project timeline.
 - It is recommended to start with a 'wide net' and narrow the Match Criteria during tuning.
- Algorithms that are not properly tuned can lead to an abundance of false positives which will require reloading data.
- The Matching Tuning configuration does not allow a pre-processor for standardizing addresses. Instead, it is recommended to use the batch processor tool (only for Loqate Local/On-Prem) to standardize addresses in the data set prior to match tuning.
 - For details on the Loqate Command Line tool, see the **Loqate Local Command Line Tool** topic in the **Data Integration** section of online help documentation.

7 Data Model Best Practices

Identifying a robust data model is crucial to ensuring the customer requirements are fulfilled and influences the user-experience. The initial configurations provided by the CMDM enablement team contain recommended data models for various use cases, however, minor changes are commonly introduced since each implementation is unique.

It is important to distinguish between transactional data and master data. It is not recommended for transactional data to be modeled and stored in STEP as it is not the primary purpose of an MDM solution.

- Data Containers
 - It is recommended to use data containers when modeling attributes or groups of attributes that will be repeated for an entity. This reduces complexity and the need for redundant attributes. For example, Address, Email, and Phone information often occurs more than once for a given record.
 - It is recommended to use data containers when modeling Addresses vs modeling them as separate objects. Since ownership and maintenance of addresses are often specific to each organization, modeling addresses as data containers reduces the complexity of maintenance by an organization vs address objects that are shared across multiple organizations.
 - For performance considerations, it is not recommended for a single record to contain more than 1000 data containers. If the solution dictates this volume of data containers, reconsider the nature of the data being represented and use an alternative data model.
- Organizational Hierarchies
 - When constructing a hierarchy, build the relationships to be displayed starting from the bottom of the hierarchy.
- Data Profiling
 - For data profiling, it is recommended to create a collections root to contain collections for various profile configurations.
 - Profile configurations can be shared across collections to reduce redundancy in set up.
- Supplier Self-Service
 - For supplier self-service solutions, associate all entities to a supplier classification, otherwise it can be viewed by all suppliers.
 - Entities with no supplier association should be referenced to a 'No Self-Service' supplier classification.
 - Since external supplier users access STEP to manage their supplier entities, user privileges must be carefully considered and executed to ensure governance.

8 JavaScript for Matching

Writing and executing JavaScript may be required to handle complex requirements for survivorship. Below are general guidelines when considering scripting for survivorship rules:

- Scripted survivorship rules should not write to objects other than the target golden record as it hinders the matching and survivorship process. This is safe guarded with STEP 10.2 and later versions.

- Source record data should not be written to data containers on the target golden record object as it is subject to unnecessarily high volume of writes over time. Rather, carefully consider what the business requirement is for retaining source record information and to what extent.
- For optimal performance, survivorship rules should not write to externally maintained attributes.
- Survivorship rules should not republish events as it is likely to lead to unnecessary events. Rather, consider triggering events by a change in workflow state or approval status.
- An accumulative survivorship library for data containers created by the CMDM enablement team is available in the **Initial Configurations** which are available by request.

9 Web UI Considerations

The following are general considerations for working with the Web UI. Some Web UI components are specific to party domain and may require applying dedicated components and/or licenses. For example, the Data Policies functionality requires that the data monitoring license is enabled.

- For data containers, it is recommended to use the 'Global Data Container Representations' configuration.
- High Priority Status Flags can be configured to indicate the workflow tasks that have high priority.
- For attributes displayed in the Web UI, it is more efficient to create and display dedicated attribute display groups.
- When multiple hierarchies are required, separate Web UI screens should be configured for each hierarchy.
- To simplify navigation in the Web UI, the search bar can be configured to search on source record ID.

10 Integration Considerations

Consider the integration requirements and options that STEP provides and analyze the following requirements:

- Synchronous vs asynchronous integrations
- Full-feed vs event-based processing
 - For typical integration patterns, delta feeds are common.
 - If periodic full loads are required, account for this in solution design, operational planning, and system maintenance activities.
- Message transformations
- Value harmonization across systems
- Protocol conversions
- ID mappings in master data and reference data

- Error-handling (transport, transformation, communication, data integrity constraints, etc.)

As the variety of downstream systems and processes that consume master data are numerous, it is recommended that master data is published to a data warehouse for external systems and processes to extract. This is common when there are downstream reporting and analytics platforms which require master data.

- Determine the external systems that need to integrate with STEP.
 - Considering the necessary upstream and downstream systems enables identifying the integration pattern that is most appropriate for the customer.
- Decide how each system will obtain data from STEP (web service endpoint, integration endpoint, etc.)
- Consider these party domain-specific web services:
 - For details on Find Similar web service, see the **Web Service Endpoint - Find Similar** topic in the **Data Exchange** section of online help documentation.
 - For details on Match & Merge web service, see the **Web Service Endpoint - Match and Merge** topic in the **Data Exchange** section of online help documentation.

11 User Privileges

When considering user permissions, it is important to evaluate the business context of each user group's role within the system. Carefully consider external users accessing MDM via the Web UI, as well as internal steward users.

- The most common user privilege assignments determine if the user has read and write privileges for certain objects and when.
- It is generally recommended that permissions are governed using STEP privilege configuration (assigning 'read-only' privileges to users and groups) as opposed to Web UI configuration (checking the 'read-only' box in every Web UI screen configuration). This reduces repetition across multiple screen configurations which can lead to oversight and user error.
- For detailed information on user permission best practices, see the **Privilege Rules** topic in the **System Setup** section of online help documentation as well as SIMA.

12 Third-Party Services

STEP supports general integration capabilities with third-party services. The third-party CMDM services that are natively supported by STEP include:

- Loqate
 - Address verification comes with multiple licensing and deployment options.
 - Local Cloud – Address verification via a cloud API

- Local On-Premise/Local – Local deployment of address data packs for large volume address verification
- CASS certification for Loqate Local – CASS certified address verification for US-based addresses
- Experian
 - Email Validation is an asynchronous batch solution.
 - Expected process times for email validation are the following:
 - < 10k, 12 hours
 - 10k to 100k, 20 hours
 - 100k to 250k, 40 hours
 - 250k to 500k, 60 hours
 - 500k to 1m, 72 hours
 - 1m to 3m, 10 days
 - 3m to 10m, 15 days
- Dun & Bradstreet
 - STEP natively supports integration with D&B’s Identity Resolution and Company Profile with Executives, Linkage, and Financials.
 - Other D&B Direct+ offerings are also accessible through additional configuration.

To ensure network accessibility to third-party services, allow access as follows:

- For Loqate:
 - <https://download.loqate.com>
 - <http://saas.loqate.com>
 - <https://licensing.loqate.com>
- For Experian:
 - <http://www.qas.com>
 - <https://api.experianmarketingservices.com>
- For Dun & Bradstreet:
 - https://toolkit.dnb.com/ws/DNB_WebServices.Providers.LookUp:wsp_LookUp

The workflow for deploying any third-party service includes:

- Inform the Stibo Systems Account Manager or Alliances Director for Technical Partners to ensure proper Stibo Systems and third-party subscription licenses keys are obtained and shared with the implementation team.
- Validate that the Stibo Systems' Technical Services team ensures the proper Stibo Systems (Prism) licenses have been enabled and STEP components applied to all relevant environments.

The considerations for deploying any third-party service include:

- As third-party services incur transaction-based costs, be clear on the number of transactions purchased when conducting large-scale testing, data migration activities, and the number of environments that require these services.
- Third-party services generally rely on specific Component Models to be deployed in the application. Component Models provide configuration instructions for functionalities that require multiple objects, references, or other elements to perform a specified operation.

13 Troubleshooting Tips

It is recommended that when troubleshooting issues, practitioners use an evidence-based investigation approach. This ensures that issues are logically identified, resolved, and that time is not wasted.

For all issues that require Stibo Systems Support assistance, use the established support process by first creating a JIRA ticket.

- In JIRA, to reduce turn-around time, provide steps to reproduce the issue, include thorough explanations of the expected behavior vs actual behavior, and attach images and videos to illustrate the problem.
- Convey to the Stibo Systems Support team the urgency of the issue and provide a Business Impact Assessment when relevant, i.e., production is down, issue is an immediate blocker to customer go-live, etc.
- Identify any deprecated components that are being used.
- Specify if the proper Stibo Systems license(s) have been applied for the specific STEP functionality

When troubleshooting issues related to third-party services, it is helpful to differentiate between Stibo Systems integration and/or solution issues vs third-party data issues. Since Stibo Systems does not own third-party data, specific inquiries or enhancements are routed to the support process managed by each respective third-party service. Some recommended troubleshooting guidelines include:

- Ensure third-party license keys have been received, deployed, and/or updated.
- For Loqate Local (On-Premise) deployments, check that there is adequate disk space available to deploy the necessary data packs and ensure the system properties are properly configured.
- For Loqate Local, the address data packs require periodic refreshes to leverage most recent information. The recommended interval is once a year.
 - CASS data packs are required to be refreshed every 105 days

14 General Performance Considerations

For questions and issues related to general performance and other issues:

- Identify if In-Memory has been enabled for the proper environment(s).
- See the **Matching Considerations** section above for information regarding match codes group sizes.
- For initial data loads or other data migration activities, it is recommended that these exercises are rehearsed to streamline the effort.
- Since sandbox environments are typically sized differently than production environments, it is not recommended to rehearse migration activities in sandbox environments.
- STEP log information is important in facilitating investigation of system issues. Avoid setting logging for specific STEP components to 'FINE' or 'FINEST' as this may render the logs ineffective, i.e., logs are overwritten too quickly to be of any value.
- It is recommended to avoid using externally maintained attributes to ensure matching event processor efficiency. Externally maintained attributes do not work with survivorship rules.
- Survivorship rules should only write data to the surviving golden record.
- When configuring data profiles, it is more efficient to profile against a collection of objects as opposed to profiling on individual objects.
- Avoid including calculated attributes in data profile configurations.
- Avoid using the Source Record ID attribute for any other purpose aside from modeling the customer/supplier record ID as provided by their source system(s). This is a look-up key and can lead to performance challenges if used in parts of the solution other than as metadata on the source system reference.
- For STEP 10.3 and newer, it is recommended to leverage the new background process execution management framework which streamlines the parallelization of background processes for improved performance. For more information, refer to the **BG Processes Execution Management** topic in the **System Setup** section of online help documentation.

15 CMDM Enablement Team

The CMDM product enablement team regularly supports ongoing Party Domain MDM implementations by providing best practices and solution guidance, while also gathering insight to business use cases to expand our understanding of the domain. The CMDM product team schedules regular touchpoints with the implementation team to gain direct insight into the progress of projects and to assist in solution design challenges.

Notify the CMDM Product team as new party-domain engagements begin with new / existing customers.

- Enablement Guidelines

- Ensure the practitioners are familiar with the **Customer & Supplier MDM Solution Enablement** in the **Solution Enablement** section that complements the online help documentation
- Initial Configurations
 - The CMDM Initial Configurations provide base system configuration including data model (object types, attributes, and references), workflows, user privileges, Web UIs, profile configurations, etc.
 - Ensure the practitioners have been made aware of / requested the **Initial Configurations** that is available from the CMDM Product Enablement Team.
 - The CMDM **Initial Configurations** accommodate the following use cases, and each includes a base data model, matching algorithms, sample data, workflows, and Web UI configurations.
 - B2C for Individuals in Coexistence/Consolidation-style implementation
 - B2B for Organizations and Suppliers in Coexistence/Consolidation-style implementation
 - B2B for Suppliers in Centralized-style implementation and with Self-Service
 - Base data model for SAP R/3 Customer and Vendor with Debmas and Cremas iDoc XSLT
 - Base data model for SAP S/4 Business Partner
 - Dun & Bradstreet Web UI configuration



BETTER DATA.
BETTER BUSINESS.
BETTER WORLD.

Stibo Systems, the master data management company, is the trusted source of MDM. Our solutions are the driving force behind forward-thinking companies around the world that have unlocked the strategic value of their master data, empowering them to improve the customer experience, drive innovation and growth and create an essential foundation for digital transformation. We give companies the transparency they require and desire – a single, accurate view of their master data – so they can make informed decisions and achieve goals of scale, scope and ambition. Stibo Systems is a privately held subsidiary of the Stibo A/S group, founded in 1794, and is headquartered in Aarhus, Denmark. For more information, visit stibosystems.com