The Science of Data Migration: Bridging Theory and Practice in Real-World Scenarios

Laxman Vattam

March 20, 2025

International Journal of Leading Research Publication (IJLRP)

E-ISSN: 2582-8010 ● Website: www.ijlrp.com ● Email: editor@ijlrp.com

The Science of Data Migration: Bridging Theory and Practice in Real-World Scenarios

Laxman Vattam

LaxmanVattam@gmail.com
Independent Researcher, Texas, USA

Abstract

Salesforce data migration is the process of moving data from external sources or legacy systems, typically stored in databases, into the Salesforce platform—a cloud-based customer relationship management (CRM) system. Organizations typically perform this migration when implementing Salesforce to handle customer data, sales, marketing, and various business operations. Successful data migration is crucial for the overall success of the project and user adoption. However, the difficulties and intricacies involved in data migration are often overlooked. This document serves as a collection of best practices, tips, and resources, drawn from extensive experience in managing intricate data migration projects for numerous customers.

Keywords: Salesforce, Data Migration, Map Legacy Data, Data Cleansing, Data Analysis, Dispatcher, Large Data Volumes

Introduction

Almost every Salesforce customer aims to populate their new organization with data from the legacy system(s) they are replacing. Technical Architects, Data Architects, Solution Architects and Quality Assurance team all have a role to play in successful data migration. Everyone involved in developing and testing the data migration process might need to undergo data security training depending on the type of industry and type of data involved. Use a RACI matric to clearly define the responsibilities of all the team members involved.

Data Migration Project Planning:

1. Align with customer to makes sure customers provides the source data, staging environment and counts of records. Customer is responsible for deprecating the existing system and user training on the new system.
2. Clearly define the cut over plan, go-live dates and the communication that needs to be sent to their internal users.
3. Effective field service management brings multiple advantages to external stakeholders.
4. Determine data security requirements based on the industry customer is operating and the level of sensitivity of the data.
5. Establish procedures to avoid the data getting manipulated in unintended ways to keep the integrity of the data.

6. Discuss on a contingency plan in case the data migration does not get completed as planned.

Salesforce Data Model:
Designing your data model is a fundamental step in any implementation. Even when following an agile approach, it is important to establish an initial data model early in the process and implement a change control mechanism. This is particularly crucial when dealing with complex integrations, a data migration team developing ETL processes, or multiple teams working concurrently.

Best Practices for Salesforce Data Model Design

- Interconnection of Data Modeling and Sharing Design: Relationships between objects, such as lookup or master-detail (M-D), have major implications for data sharing. Address these considerations early, as changing them later in the implementation process can have widespread effects.
- Identify Essential Salesforce Objects: Determine the key Salesforce objects required to support the intended functionality.
- Future-Oriented Data Structure: Design the data model based on how the customer envisions using the data rather than simply replicating the legacy system. Directly copying an old data model can result in inefficient solutions.
- User-Centric Approach: Focus on the data users need to interact with to be successful. Evaluate current data usage—if a data element exists in the legacy system but is rarely used or left empty, avoid migrating it.
- External Data Sources: For data that will remain in external systems, consider utilizing external objects instead of migrating all data into Salesforce.
- Data Volume Estimation: For each Salesforce object, assess the amount of data to be migrated and project its growth over time.
- Storage Planning: Calculate the necessary data storage and verify that the customer's licenses provide adequate capacity. Utilize the Data and File Storage Calculator for accurate estimations.
- When handling large data volumes during migration, skinny tables can enhance the efficiency of reports, dashboards, and queries.

Object Relationships and Key Considerations:

- Understanding Relationships: Analyze the best relationship structures for your objects, including entity-relationship diagrams (ERDs) for Salesforce standard objects.
- Use Standard Objects When Possible: Avoid creating custom objects to replicate standard Salesforce functionality (e.g., Leads, Contacts, Opportunities).
- Denormalization for Usability: Unlike traditional databases, excessive normalization in Salesforce can make reporting, querying, and navigation more difficult due to complex joins. Common areas where normalization may seem intuitive but can create challenges include addresses, phone numbers, and user preferences. Keep data structures optimized for Salesforce's environment.

Types of data:
Configuration Data:

Configuration data is typically static data that needs to be loaded into Salesforce, often into Custom Metadata Types or Custom Settings. Custom Metadata Types are recommended because they allow for easier migration using packages or Metadata API tools.

Unstructured Data:

Attachments are being phased out in favor of Files. To migrate attachments, you can first move them to the target Salesforce org using Data Loader and then convert them to Files. Several tools, such as Magic Mover from Salesforce Labs, can help with this process.

Master Data:

Master data is information managed outside Salesforce, such as customer data maintained in an MDM system. Periodic synchronization is usually required for this data. Proper external keys must be established to ensure smooth integration with the system of record.

Historical Data:

Historical data may be needed for audits or to provide a complete customer view. Salesforce does not support importing history data into field history tables; instead, custom objects must be used to store that information. For audit field migration (e.g., Created By, Created Date), we need special permission to be enabled, but note these fields can only be populated during record creation, not updates.

Transactional Data:

Transactional data is operational data generated and maintained in the source system, such as work order data, with the legacy system being the authoritative source. It’s important to decide what transactional data should be migrated to Salesforce, as un-migrated data will either need to be archived or lost once the legacy system is retired. Alternatively, transactional data can be connected as an external source in Salesforce.

Knowledge Articles:

Migrating knowledge articles to Salesforce Knowledge requires attention to nuances such as Lightning Knowledge's use of record types instead of article types. If using Knowledge Classic, articles must first be consolidated into a single type before migrating. Data Loader can update the record types afterward.

User Data:

Migrating user data from a legacy system involves several considerations, such as mapping legacy users to Salesforce users, handling inactive or expired users, and addressing licensing, profile, and role hierarchy. It’s recommended to load user records into Salesforce for audit purposes but it adds complexity. Different strategies can be applied based on legacy data relevance, such as mapping legacy users to generic Salesforce users for older records.

Approaches for User Data Migration

- Map legacy users to new Salesforce users for lookup fields.
- Request extra licenses temporarily to load user data.
- Aggregate legacy users into fewer Salesforce users based on business needs.

Be cautious with email data, as loading email addresses may trigger unintended emails to customers and users.

Considerations for PII and Encryption:

If personally identifiable information (PII) is encrypted in the legacy system, it must be decrypted before being re-encrypted in Salesforce. Verify security requirements to ensure PII data can be stored in Salesforce. Additionally, if test environments are used for migration, ensure tools or procedures exist to mask any PII data.

Large Data Volume (LDV):

For large-scale user data migrations, turning off the “available license” check during the load (through Support) can significantly improve load times. Always follow LDV best practices to avoid data skew.

Data Cleanup:

Data Quality Assessment

- Estimate Data Volume: Determine the expected volume for each object to be loaded into Salesforce.
- Compare Current Data to Desired State: Assess how the existing data in the legacy system compares to the ideal state in Salesforce. For instance, evaluate how many duplicate contacts exist in the current system versus what should be in Salesforce. The difference will help gauge the level of effort needed for data cleansing and preparation.
- Understand Data History: Investigate any historical data issues that could indicate the need for new governance processes, or adjustments in the new system. Ask about past challenges that affected data quality.

Source Data:

- Intentional Duplicates: Sometimes, duplicate accounts are necessary, such as when customer and partner accounts represent the same business entity.
- Data Cleanup: Based on data profiling results, enrich the data, remove duplicates, and restructure child records as needed.
- Duplicate Detection Tools: Use Salesforce duplicate jobs (available in Performance and Unlimited editions) for post-migration or sandbox environments. Ensure matching and duplicate rules are set up and activated.

- Third-Party Tools: Consider using tools like CloudDingo, Trillium, and Informatica for deduplication, or leverage data on the Lightning Data Exchange.
- B2B and B2C Data: For B2B data, use Data Exchange packages such as Dun & Bradstreet and customized assessment reports. For B2C data, use custom packages with Axciom APIs and services.
- Account Hierarchies: Build customer account hierarchies with third-party sources like Dun & Bradstreet.

Picklists:

- Value Overlap: Ensure picklist values are mutually exclusive and eliminate overlapping values.
- Data Standardization: Establish a data standard for each domain and map legacy values to these new standards, addressing granularity and missing values.
- Reference Codes: Identify legacy reference codes that suggest poor data modeling and correct them before migrating to Salesforce.

Transaction Data:

- Profile and Review: Analyze transaction data to identify duplicates that need to be removed during migration.

History Data:

- Evaluate History Data Needs: Determine if migrating historical data is necessary. In many cases, it may be more efficient to archive old data or move it to a data warehouse instead of migrating it to Salesforce.
- External Objects: Consider providing access to historical data through external objects or other integration methods.

Data Analysis and Profiling:

- Real Data Analysis: Data profiling must be done on production-like data, not dummy or obfuscated data, as this has security and environment management implications.
- Data Accessibility: Ensure that only authorized individuals can access sensitive data, and define where the data can be stored securely.

Data Cleansing:

- Fix Issues Before Migration: After profiling and analysis, address any data quality issues in the source system before extraction.
- User Involvement: Provide reports or tools to help users cleanse the data, and manage this as a project task assigned to the appropriate personnel or teams.

Data Mapping and Transformation rules:

- Create a document that maps from source data onto target Salesforce org. Make sure the data is compatible with target field.
- Design and apply transformation rules as needed by business

ETL Tools:

In most of the cases, many clients already have a ETL tool selected for companywide IT activities. They can leverage the same tool for the data migration if it supports huge data volumes as needed.One of the popular ETL tool is Informatica.

If the data volumes are low, we can use Data Loader for the data migration activities.

Large Data Volume Considerations:

Depending on the volume of data to be migrated, choose a tool that can handle the volume without performance issues.

Consider below options to speed up the data loads:

Defer Sharing Rules:

When making many configuration changes, it can result in lengthy sharing rule evaluations or timeouts. To prevent such issues, an administrator can suspend the sharing calculations—specifically for sharing rules and group membership—and then resume them after the required changes are in place. This helps ensure that the changes are processed without causing performance issues. This is also applicable during large data volume data loads. We can defer sharing while the loads are in progress and resume them after the data loads are complete. Please make sure to run the sharing rules for all objects to recalculate after all the data loads are complete.

Master Detail:

When loading child records that are in master detail relationship, parent records get locked. To avoid data load failures, sort the data on the master data field. Make sure parent record doesn’t get referred in multiple threads to avoid locking failures.

Data Skew:

Data Skew can happen is large number of child records (greater than 10000) have a lookup onto one parent record. One of the common situation I Account Data Skew. When a large number of Contacts or Opportunities are created under the same parent Account record, Account data Skew can occur.

If assigning a lot of records under a parent is unavoidable, distribute the records under different parent records to avoid lookup skew.

Ownership Skew:

When loading child records that are in master detail relationship, parent records get locked. To avoid data load failures, sort the data on the master data field. Make sure parent record doesn't get referred in multiple threads to avoid locking failures.

If Sharing Model for Contacts is Private and a lot of sharing rules are created for Contacts, whenever a change is made to sharing rules, it will force a recalculation. If the number of contacts are is high under the same Account, it can cause performance issues for the Salesforce users.

To avoid Ownership Skew, avoid records being owned by a dummy system user. If this is unavoidable, assign these records to a owner who is at the top of role hierarchy or has full access to all records.

Bulk API with possible parallel loading is one the optimized solution to load huge volumes of data.

Knowledge Migration:

Lightning Knowledge Migration Tool can be used to migrate Classic Knowledge into Lightning Knowledge.
There are few things to consider about Lightning Knowledge Data Structure post data migration:

- User Permissions: New author permissions are needed to continue with Lightning Knowledge
- RecordTypes: We need to map article Types to RecordTypes

After successful migration, validation of RecordTypes ,fields, label and Files are migrated as expected.

Quality Assurance Testing:

Unit Testing:

- First step of testing is to perform unit testing to uncover any potential issues with data migration.
- A good set of test cases need to be created to test various aspects of data quality and usability.
- Make sure to include the test cases specifically to verify data transformation logic.
- Test various business processes using the data migrated to make sure the data is compatible and none of the automations are broken.
- If data has downstream processes, testing needs to be done to make sure the volume of data loads is being handled without any failures. Make use of queuing where possible to avoid performance impacts on downstream systems.

UAT/User Acceptance Testing:

- UAT users need to be setup with the real profile and permissions in the UAT sandbox.
- UAT users need to test the end-to-end process and make sure the results are aligning with their expected output.
- UAT users need to validate the various field data on UI/Page Layouts. They need to make sure they have necessary permissions to view and edit various fields.
- UAT users need to test the processes that trigger actions on external systems and make sure the data flows as expected and it does not break external systems.
- UAT users can run reports to filter and validate the data as needed.

Performance Testing:

We need to perform end to end data migration in a full copy sandbox and make sure the to have users work on the system while data loads are going on. This is to test whether there are any records locks or performance issues caused by data loads that affects the user

Data Migration Planning:

- A decision needs to be made on whether business needs to operate while Data Migration activities are in progress and how it could impact them.
- A plan must be made on whether business needs to complete certain actions in their legacy systema before the data is migrated to Salesforce.
- If there are any impacts to integrations, make sure all the migrated data has external ids that are aligned with external systems.
- Ensure no automated emails are triggered for the data created by data migration process

Speed up the load performance:

To improve the data load performance:

1. Turn off the triggers, validation rules and any other logic that gets triggered on data insert and update.
   If business users need to operate in parallel to data migration, include logic in triggers and validation rules to stop them for firing for that user that is being used for Data Migration.
2. If the data volumes are high, consider turning off sharing calculations while the data loads are in progress. While this approach can improve data load performance, it will need additional steps to be performed to calculate sharing after all the data has been loaded.
3. For the data has lot of lookups, if the lookup values are transformed to include the Salesforce Ids, it will improve the performance drastically.

Smoke Testing and Final verifications:

- Reports can be created to filter and verify various types of data.
- Verify the final record counts and review the failed records.

Contingency plan:

- Make sure there is more than enough storage to be able to load all the data. There is additional storage needed as soon as we go live and the users start creating new data.
- Make sure to have support for various tools used in the data migration process. If any of the tools stop working, we need to engage the support immediately to avoid the data migration process from being halted.

Roll back plan:

If this is a brand-new Salesforce org, we can skip this step. Otherwise, we need to make sure there is a data back up of all the existing data and incase, data migration is not completed as planned, we need a strategy to load/update the existing data to its original state.

Conclusion

The process of data migration in Salesforce demands thorough preparation and significant resource allocation for development, testing, and execution. The initial phase involves analyzing business data and determining how to effectively map the source data model onto the target model while considering the relationships between various objects. Establishing a data governance policy is crucial, with clearly defined roles and responsibilities. It is essential to keep key stakeholders informed about the migration plan, including any potential downtime for users. Additionally, creating a backup of existing Salesforce data and testing a rollback strategy are critical steps. Preparing the target organization by eliminating duplicate and poor-quality data ensures a smoother transition. Selecting an appropriate data migration tool that can efficiently manage the required data volume is also vital. With the right tools in place and a well-structured action plan, the migration process can be executed successfully.

References

[1] Salesforce, “Data and File Storage Allocations,” *Salesforce Help*, Available:https://help.salesforce.com/s/articleView?id=xcloud.overview_storage.htm&type=5[accessed Oct19, 2024].

[2] Salesforce,“Salesforce Anti-Patterns: A Cautionary Tale”,*Salesforce Blogs*. Available:https://developer.salesforce.com/blogs/engineering/2014/11/salesforce-anti-patterns-a-cautionary-tale[accessed Oct25, 2024].

[3] Salesforce, “Avoid Account Data Skew for Peak Performance,” *Salesforce Blogs*. Available:https://developer.salesforce.com/blogs/engineering/2012/04/avoid-account-data-skew-for-peak-performance [accessed Nov 15, 2024].

[4] Salesforce, “The Salesforce Bulk API – Maximizing Parallelism and Throughput Performance,” *Salesforce Blogs*.

Available:https://developer.salesforce.com/blogs/2014/02/the-salesforce-bulk-api-maximizing-parallelism-and-throughput-performance-when-integrating-or-loading-large-data-volumes [accessed Dec 6, 2024].

[5] Salesforce, “Lightning Knowledge Migration Tool,” *Service Cloud*. Available: https://help.salesforce.com/s/articleView?id=service.knowledge_migration_tool.htm&type=5 [accessed Jan 15, 2025].

[6] Salesforce, “Data Loader,” Salesforce Help. Available: https://help.salesforce.com/s/articleView?id=xcloud.data_loader_about.htm&type=5 [accessed Jan17, 2025]