

# Best Practices

**Integration of Salesforce and Microsoft Dynamics GP**

# Best Practices:

## Integration of Salesforce and Microsoft Dynamics GP

### Introduction

#### Introduction

**Customer Relationship Management (CRM)** is mainly used to help an organization to manage its interactions with customers. CRM uses technology to manage, computerize and synchronize various departments like sales, marketing, customer service, technical support etc. Salesforce.com and Microsoft CRM are the commonly used CRMs in many of the organizations.

Microsoft Dynamics GP is a business solution from Microsoft that delivers comprehensive, out-of-the-box business management functionality to power diverse small and midsize companies around the world.

In CRM/ERP world generally there is **Order to Cash process**. When the order is created, it must go through a cycle of invoicing, payment request and receiving payment. When Opportunities are closed, Estimates or Orders are created. Estimates usually represent an Opportunity.

Best practices in integration of CRM and Microsoft Dynamics GP is explained considering Salesforce as an example for CRM. Generally in Salesforce, process flow starts with the sales team's involvement in creation of **Accounts, Contacts** and then **Opportunities**. Flow in Salesforce completes when the product is sold and then an **invoice** needs to be sent. Customer makes the payment as per the received invoice. Accounts team receives the payment and updates the invoice as paid. Accounts Team which uses Microsoft Dynamics GP may have to update the invoice and other details manually in Salesforce & Microsoft Dynamics GP, if these two are not integrated.

## Purpose of Integration

Integration of CRM with Microsoft Dynamics GP must act as common platform to Salesforce (or any other CRM) and Microsoft Dynamics GP users. This integration is essential to achieve the following benefits.

- **Reduce Double Data Entry**

Re-entering the same data in Microsoft Dynamics GP which is already in Salesforce or vice versa may lead to lot of confusion if there is any difference in data. Integration reduces double data entry hence provides smooth process flow among sales and Accounts team.

- **Reduce Manual Entry**

Manual entry of data to Microsoft Dynamics GP from Salesforce or vice-versa would require an additional human resource to work on this. Integration reduces manual entry thereby reducing the human involvement.

- **Reduce Human Error**

It is observed that manual entry usually leads to human error. Human error is unavoidable in manual entry. Integration ensures that these human errors are avoided.

- **Save Time**

Double Data and Manual Entry is time consuming. As integration reduces Double Data and Manual Entry, time required for synchronization reduces significantly.

## Purpose of Integration:

## Best Practice

It is evident that integration leads to multiple benefits. It is also necessary that right practice is followed to ensure the integration process is successful and beneficial. The steps involved for the best practice for integration of Salesforce with Microsoft Dynamics GP is explained in detail with the following topics.

- Record the Requirement
- Define Master Data
- Data Flow Triggers
- Design and Data Mapping
- Configuring GP for Integration
- Testing and Validation
- Go-Live
- Data Migration into Salesforce

## Best Practice

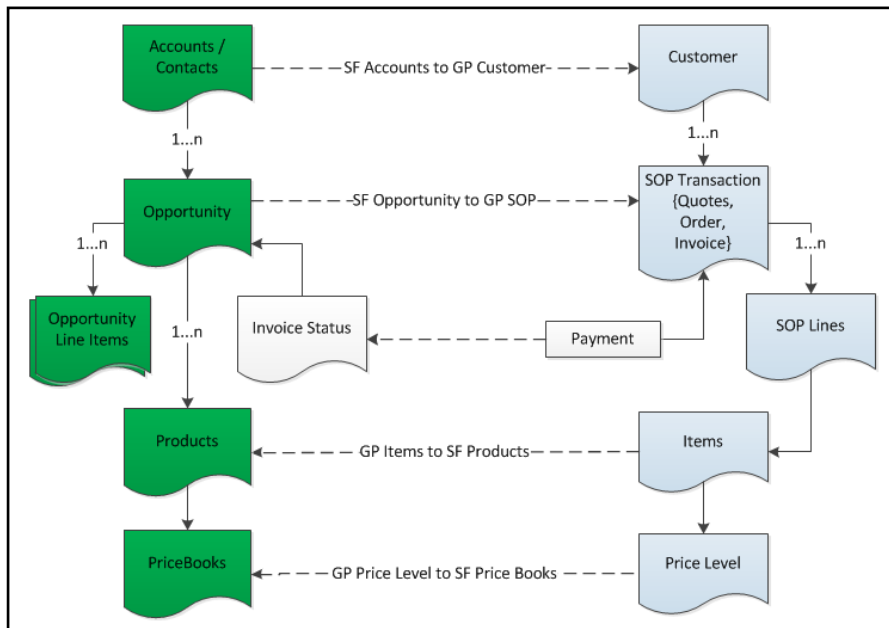
## Record the Requirement

- **Purpose and Goal Definition**

Purpose and Goal of the integration must be defined and the same must be written down to ensure that integration does not deviate from the original requirement. If there is a change in requirement, purpose and goal of the integration must be changed accordingly. The document versioning also plays a very important role for tracking changes and enhancements

- **Flow Chart for Integration**

A picture is worth a thousand words. Representation of data-flow or process-flow using diagrams like the one shown below listing out entity mapping. Flow diagrams must provide entity mappings listing out relationship between various entities concerned whether Unidirectional or Bidirectional flow etc. An example is illustrated below as shown.



- **Easy to Understand**

Documentation of requirements must be easily understood, so that a person not familiar with the business can also easily understand the need & process. The below table describes the symbols used in Flow Chart and makes this data flow self explanatory.

	Standard Salesforce Objects
	Custom Salesforce Objects
	Microsoft Dynamics GP Entities
1...n	one-to-many relationship
n...1	many-to-one relationship
1...1	One-to-One relationship
SF	Salesforce
GP	Microsoft Dynamics GP

## Record the Requirement

## Define Master Data

Master Data must be defined to decide the direction of data flow. Master Data can be either Salesforce or Microsoft Dynamics GP depending on a requirement. Defining Customer Master Data or a Product Master Data as per the nature of Business or a specific requirement can be explained as follows.

### Customer & Vendor Master

- Most companies prefer to have Salesforce as Master Data for Customers as all of Sales and Marketing is automated within Salesforce and hence Salesforce or CRM systems naturally takes precedence over Accounting (GP) in regard to Customer data.
- If GP is the source system of record for all Account Receivables and Account Payables, the recommended practice is to have the integration uni-directional with data flow from GP into target CRM. In this scenario GP will be the master data for Customer data.
- If 3rd party eCommerce apps are involved, then again it is preferred that Microsoft Dynamics GP is Master Data.

### Product Master Data

Service based firms prefer to have Salesforce as Master Data whereas Firms with inventory and Manufacturing Industry prefer to have Microsoft Dynamics GP as Master Data.

### Reporting

Salesforce always takes precedence over GP for all kind of report activities.

## Data Flow Triggers

Dumping all the customers into Microsoft Dynamics GP may inflate the GP system. Hence only the required customers must be integrated. It is a good practice or a recommended approach to move only those customers which have transactions. Triggers are very important in moving the required specific data from one system to the other.

- Records (Customers or Vendors or Items) should be pushed only if there is a valid “accounting transaction” like a Sales Order, Invoice or a Vendor PO associated. It is a usual practice to make use of timestamp changes as sync flags to trigger the integration data-flows as it will ensure faster data transfer between the two systems.
- Generally, for most of the companies using Salesforce as their CRM system, it is not recommended to integrate Estimate into GP as it might bloat your accounting system and hence Salesforce is a better place to manage Estimates and Opportunities.
- Decide “when” a transaction must flow into Microsoft Dynamics GP. Salesforce provides many ways to manage various stages of a given Opportunity and finally trigger the sync upon Closed-Won stage to create Sales Order in Microsoft Dynamics GP. This process holds good for manufacturing firms whereas for Service based business “Closed-Won” Opportunities can directly be translated into a Sales-Invoice.
- Identifying processes to know what triggers can be built leveraging Salesforce workflow or Apex (customizations in case of implementing a complex business workflow).
- If GP is the system of record for integration, then it is always recommended to use SQL Server database to retrieve data from GP using available database columns as sync flags or data-flow triggers. GP Customer, SOP Transaction and Item tables provide timestamp columns such as “DEX\_ROW\_TS” which is a datetime data-type in SQL Server. This column stores the last modified time for a given record in GP. DBSync ETL tool has an in-built functionality to store timestamp upon every successful sync. Using this unique feature of DBSync, one can compare difference in time stamp changes and trigger data-flows for only those records, which needs to be processed as opposed to pulling the entire table data. One such example is illustrated below.

**\$last\_success\_run:** DBSync variable which stores timestamp of last successful sync between GP and Salesforce.

**DEX\_ROW\_TS:** GP timestamp column to record last modified time for a given record in GP.

**LastModifiedDate:** It is a standard field in Salesforce for every object to record a timestamp change for a given record.

**RM00101:** Customer Master table in GP.

**Use-Case 1:** Query only those customers from GP which were edited / changed after the integration was completed.

```
SELECT CUSTNMBR, CUSTBLNC, LASTPYDT, LPYMTAMT, PYMTRMID, CONVERT(nvarchar, cast(CM.DEX_ROW_TS as datetime), 126) as 'DEX_ROW_TS' from RM00101 where DEX_ROW_TS > $last_success_run
```

**Use-Case 2:** Query only those Salesforce Accounts for which a change in the Last Modified Date has been triggered.

```
SELECT Id, Name, LastModifiedDate from Account where LastModifiedDate > $last_success_run
```



## Design and Data Mapping

The following guidelines can be referred to design and map the data flow for integration.

- Most of the CRMs are customizable and in order to have successful integration, CRM must be customized based on Microsoft Dynamics GP settings.
- The structure of “1 to many” relationships of Transactions (Sales Order or Invoice) to their respective line items should be maintained in Salesforce.
- Use a spreadsheet to document the data-mapping and get it reviewed by the accounting team as invariably the data insert is happening into the Accounting system.
- Create a field on each Salesforce object to indicate that the record was created or touched by the integration implementation of Salesforce and Microsoft Dynamics GP.
- Integration must also facilitate additional customization for future changes.
- Field format used in Salesforce and Microsoft Dynamics GP must be compatible for integration.
- Destination Limitations plays vital role while integrating: One of the examples for destination limitation is Text limitations. Salesforce field allows text of 255 characters whereas Microsoft Dynamics GP “Customer Name” allows only 64 characters. While integrating from Salesforce to Microsoft Dynamics GP, Microsoft Dynamics GP “Customer Name” can take only the first 64 characters of text of Salesforce field.

- **eConnect APIs or SQL Server - When to choose one over the other**

For any data-insert into GP, leveraging eConnect APIs is a recommended approach. Doing a data insert into GP using the SQL Server approach can be time consuming and complex. eConnect APIs can also be utilized for retrieving data from GP for mostly demographic data like Customers, Vendors and un-posted receivable transactions.

However, not all data can be retrieved from GP using eConnect APIs. The short-coming of eConnect APIs are seen while retrieving posted or historical transactions from GP as there are no given set-of API calls to retrieve historical data. In such a scenario, the data can still be retrieved using DBSync SQL Server connector by designing existing or custom views or queries from the corresponding GP tables.

Also, another use case when SQL Server approach takes precedence over eConnect APIs for integration with GP is when one needs to integrate GP out-of-the box reports such AR Aging Summary, Sales Comissions and so on. These reports are essentially a bunch of SQL SERVER views in the GP database and hence can be accessed using DBSync SQL Server connector to retrieve data directly from these SQL Server views. Some of the reports that can be exported directly form a GP database would be Current Receivables Aging Summary, GL Distribution, Commission Details, AR Apply Details and so on. These views can be utilized or a custom view can also be built to retrieve custom report based on a specific requirement or business process.

- **Multiple system integration** – Integration of multiple GP systems is a complex operation and hence care should be taken to ensure that Naming convention is unique across the two systems. It is always a recommended approach to make sure to use a single Salesforce instance with multiple GP Databases using prefixes on the external ID columns for the respective GP company database.

## Configuring GP for Integration

Before going live, get the latest version of Microsoft Dynamics GP from your production and test against the latest snapshot of your Salesforce instance.

It is also necessary that Microsoft Dynamics GP is configured for integration. Preferred GP configuration is explained below.

### Configuration Steps for Great Plains

- 1) Once the various Microsoft components are installed, the WSDL for GP can be accessed at.  
  
<http://<ipaddressofGPServer>/DynamicsGPWebServices/DynamicsGPService.asmx>.
- 2) The ISO currency codes have to be setup on GP.
- 3) The functional currency code has to be setup for the company.
- 4) eConnect runs as a COM+ component. I needed to set this up with “Administrator” access to get it to work. If you install eConnect with an Administrator user, it should already be setup with the right identity. The identity can be altered using the Component Services tool under Administrative tools on the Windows Server.
- 5) The Company Key for the company you are working with needs to be determined for the web service calls. The GP administrator should be able to provide this value. This value is used in most of the Web Service calls as the OrganizationKey in the Context object.
- 6) A Policy Key/Policy Object is needed most of the Web Service operations. This Policy Key is a unique GUID provided by Microsoft for each operation like (CreateCustomer, CreateGLTransaction etc.). You can get this GUID by running the GetPolicyByOperation Web Service. The Policy Name is needed as input for this Web Service call.
- 7) The Policy Key is populated in the Policy Object in most of the Web Service operations.

## Configuring GP for Integration

## Testing and validations

Testing and validation is essential to confirm that integration is working as per the requirement. The following guidelines must be referred to test and validate integration.

- Create User acceptance Test Cases which cover all the use case scenarios with your integration requirements (put link to the UAT document).
- Test and validate in Sandbox before moving into production. This is the Best practice for any integration.
- Test for volume and load to verify impact of any Apex developed (reaching governance limits) or Microsoft Dynamics GP performance limits.
- The GP hosted system or server should have enough resource allocation for a trouble and hassle free integration with GP. A minimum of 4 GB of RAM or higher and a dual core processor with a 64-bit Operating system is recommended
- In addition to the hardware configuration of RAM allocation is also essential to ensure better data transfer speeds with GP. It is also recommended to use an On-Premise edition of DBSync as opposed to an On-Demand version as higher transfer speeds can be achieved with the On-Premise version.
- eConnect Windows service needs to be configured preferably with an SA login or the user configured should have read / write permissions into the Database server where GP is hosted.

## Go Live

The below guidelines must be followed to Go-Live.

- Create a backup of your Microsoft Dynamics GP file.
- While performing Microsoft Dynamics GP to Salesforce, take a download of Objects touched in the integration using Data Loader.
- Implement the integration, and validate. Best time to do this is during downtime.
- Check and validate transaction and data flow.

## Testing and validations

## Go Live

## Data Migration into Salesforce

Pushing the old or existing data from Microsoft Dynamics GP to Salesforce is termed as Data Migration. During Migration the below guidelines must be followed.

- While performing Microsoft Dynamics GP to Salesforce, take a download of Objects touched in the integration using DBSync SQL Server Connector and identify the right table for data extraction for all entities in GP.
- Make sure each object in the migration has a field defined and mapped to indicate that the record has been pushed during migration. This will help in identifying issues, to delete and to reprocess the data import.
- Making sure Salesforce API limits are not exhausted while doing a data upload from the GP system as it can result in the incomplete migration.

## Data Migration into Salesforce

## Follow up Task

A regular follow up is required to ensure that all the best practices are incorporated during integration. In order to achieve this, the following steps can be referred as guidelines for integration.

- Define your Project and Goals.
- Identify your internal team from CRM and Accounting.
- Document your Requirements.
- Engage DBSync team to assist in reviewing your business process.
- Design your process, and create Maps using mapping spreadsheet.
- Identify and mitigate Risk.
- Register an account with DBSync.
- Setup the process in DBSync.
- Create a User Acceptance Test Document.
- Plan the dates for Test Completion and Go-Live.

Ideally Process of integration must involve the following mapping to simplify the process.

- **SF Account to GP Customer.**
- **SF Product to GP Item.**
- **SF Opportunity to GP Transaction.**
- **One-to-one relationship for SF Opportunity to GP Transaction integration.**
- **One-to-many relationships for SF Account to SF Opportunity.**