This document aims to provide detailed information regarding implementation of a synchronization architecture between Marketo and an external CRM system. Entities are described as well as the specifics of performing initial synchronization, or “bootstrapping,” and maintaining synchronization of new and updated records.
# Table of Contents

Marketo-CRM Synchronization Architecture ............................................................................. 1

Introduction ................................................................................................................................. 1

Overview ..................................................................................................................................... 4

Typical Synchronized Entities ..................................................................................................... 4

Leads ........................................................................................................................................... 4

Companies .................................................................................................................................. 6

Opportunities ............................................................................................................................... 7

Opportunity Roles ....................................................................................................................... 8

Sales Persons ............................................................................................................................... 9

Extended Synchronized Entities ................................................................................................. 10

Custom Objects ............................................................................................................................ 10

Activities ..................................................................................................................................... 11

Programs ..................................................................................................................................... 13

Program Membership .................................................................................................................. 14

Static Lists ................................................................................................................................... 15

Synchronization ............................................................................................................................ 16

Configuration ............................................................................................................................... 16

Synchronized Fields ..................................................................................................................... 16

Synchronized Custom Objects ...................................................................................................... 16

Synchronized Activity Types ......................................................................................................... 16

Bootstrapping Configuration ........................................................................................................ 16

Synchronization Interval ............................................................................................................. 16

Bootstrapping from Marketo .......................................................................................................... 17

Leads ........................................................................................................................................... 17

Companies .................................................................................................................................. 17

Opportunities and Roles ............................................................................................................. 17

Sales Persons ............................................................................................................................... 18

Custom Objects ............................................................................................................................ 18

Programs and Membership .......................................................................................................... 18

Activities ..................................................................................................................................... 18

Lists and Membership .................................................................................................................. 19

Recommended Bootstrapping Order ............................................................................................ 19
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maintaining Synchronization</td>
<td>19</td>
</tr>
<tr>
<td>Recommended Synchronization Order</td>
<td>19</td>
</tr>
<tr>
<td>Leads and Companies</td>
<td>20</td>
</tr>
<tr>
<td>Sales Persons</td>
<td>21</td>
</tr>
<tr>
<td>Opportunities and Roles</td>
<td>21</td>
</tr>
<tr>
<td>Activities</td>
<td>22</td>
</tr>
<tr>
<td>Custom Objects</td>
<td>22</td>
</tr>
<tr>
<td>Program-Opportunity Attribution</td>
<td>22</td>
</tr>
<tr>
<td>First-Touch Attribution(FT)</td>
<td>22</td>
</tr>
<tr>
<td>Multi-Touch(MT)</td>
<td>23</td>
</tr>
</tbody>
</table>
Overview

Marketo recommends implementation of CRM synchronization applications using a two-way polling model, retrieving changes and new records from either system, and pushing those changes as mapped in the application configuration to the other system. The middleware system should query for changes of selected record types periodically and then perform any necessary translation and filtration before pushing the record updates to CRM or Marketo respectively.

Typical Synchronized Entities

A minimally functional CRM integration with Marketo will provide two-way synchronization of Leads, and Companies, and one-way CRM-to-Marketo integration of the following entities:

- Opportunities
- Opportunity Roles
- Sales Persons

Global Notes for Object Models

Some object schemas in Marketo are dynamic and user-editable, so it is extremely desirable to allow dynamic updates to the schema, and this should be done as a routine part of a synchronization cycle. For these types of objects, a Describe (Object) endpoint is made available, which will return all of the fields available to the Marketo APIs, along with datatype and length information where applicable. Where available, the Describe endpoint should be used as the exclusive source of truth for the object schema. Administrative users should be permitted to add, remove and alter mapping for non-ID fields at any time.

Leads

Primary Key: id

The integer id of a Marketo lead record and the primary key are system managed by Marketo, and may only be assigned by Marketo. Any insert operations attempted by a foreign system which include id will be rejected.
In Marketo, leads represent any person-record which is a sales or marketing target. They are distinct from sales person records, which represent a sales person who has responsibility over lead records. All Smart Campaigns, commonly referred to as a “workflow” in non-Marketo systems, filter, trigger, and operate on lead records, based on their characteristics and actions. Unlike most typical CRM systems, Marketo does not have two separate record types for Lead and Contact records, rather, each person-record in Marketo has additional fields which match those available for Companies. The set of company type fields on a given record is no longer updateable when the externalCompanyId field is set for that record, and is deferred to the linked company record.

**Model**

Leads are highly extensible in Marketo and may include an unlimited number of custom fields. When synchronizing any particular subscription, a set of standard fields should not be relied upon, and the Describe Lead function of the REST API should be used as the exclusive source of truth to determine field availability and updateability in a particular subscription.

The model for a lead is also potentially dynamic, as fields may be added or hidden by end-users at any time, so the schema of lead records should be maintained whenever a synchronization cycle is initiated.

A custom lead field should be created in Marketo to hold the primary key of the CRM, in order to permit unambiguous upsert operations for lead records.

**Mapping**

Lead field mapping should be determined by the available person types in the CRM system. A typical CRM implementation would only include Leads and Contacts and would be mapped to the logical corresponding field for each record type. When possible a suggested list of mappings should be given for fields which are known in both Marketo and the CRM, but users should be allowed to change these field mappings freely to account for different usages. If leads are mapped to multiple object types, then mappings for each type should be allowed to be mapped independently. It is possible and sometimes desired to have leads mapped to multiple types of records (e.g. Leads and Contacts in the CRM), differentiating the type on the basis of a custom CRM type string field, referring either to the ID of the particular record type, or a plaintext string representing the type.

**Relationships**

Leads are related to numerous accessible object types in Marketo:

<table>
<thead>
<tr>
<th>Object</th>
<th>Relationship Type</th>
<th>Link Type</th>
<th>Link Field</th>
</tr>
</thead>
<tbody>
<tr>
<td>Company</td>
<td>N:1</td>
<td>Lead Field</td>
<td>externalCompanyId</td>
</tr>
<tr>
<td>Sales Person</td>
<td>N:1</td>
<td>Lead Field</td>
<td>externalSalesPersonId</td>
</tr>
<tr>
<td>Opportunity</td>
<td>N:N</td>
<td>Opportunity Role Object</td>
<td>leadId</td>
</tr>
<tr>
<td>Custom Objects</td>
<td>1:N, N:1, or N:N</td>
<td>See Custom Object Section</td>
<td>See Custom Object Section</td>
</tr>
<tr>
<td>Activities</td>
<td>1:N</td>
<td>Activity Field</td>
<td>leadId</td>
</tr>
<tr>
<td>Program</td>
<td>N:N</td>
<td>Program Membership Object</td>
<td>See Program Membership</td>
</tr>
</tbody>
</table>
Accessibility

Leads may be read and written to in Marketo freely provided that the Read-Write Lead Permission has been awarded to the API user being used. Leads can be read through the following endpoints:

- Get Lead by ID
- Get Multiple Leads by Filter Type
- Get Multiple Leads by List Id
- Get Multiple Leads by Program Id

Leads can be written to through the following endpoints:

- Create/Update Leads
- Import Lead

Lead Records may be merged through the Merge Leads endpoint. A discussion of this endpoint’s behavior may be found here.

Leads may be deleted through the Delete Leads endpoint.

Lead Metadata is read through the Describe Lead endpoint.

Companies

Primary Key: externalCompanyId, id

externalCompanyId is an arbitrary string field set upon creation by the external system. It is guaranteed to be unique. It is not updateable. Id is a unique system-generated integer id.

Company objects represent the organization to which lead records belong. Leads are added to a Company by populating their corresponding externalCompanyId. Leads linked to a company record will directly inherit the values from a company record as though the values existed on the lead’s own record. Attributes available on the company record are available for triggering and filtering on lead records from within the application.

Company records may only be created by external systems, and the CRM should be treated as the source of truth, except when changes to company records are detected in Marketo.

Model

- Companies are fully extensible and may have any number and type of custom fields
- The schema should be dynamic and capable of ingesting or evicting fields from the model at any time. It should be refreshed with Describe Company as a routine part of synchronization.
- Describe Company should be used as the exclusive source of truth for the schema of company

Mapping

Company records in Marketo should be mapped to an equivalent object type in CRM. These are typically referred to as Companies, Accounts, or Organizations, but terminology may vary for different
CRMs. Users should be allowed to edit the mappings of Marketo Company fields to CRM Company fields.

The primary key of company records in CRM should be mapped to the externalCompanyId field whenever possible.

**Relationships**

<table>
<thead>
<tr>
<th>Object</th>
<th>Relationship Type</th>
<th>Link Type</th>
<th>Link Field</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lead</td>
<td>1:N</td>
<td>Lead Field</td>
<td>externalCompanyId</td>
</tr>
<tr>
<td>Opportunity</td>
<td>1:N</td>
<td>Opportunity Field</td>
<td>externalCompanyId</td>
</tr>
</tbody>
</table>

**Accessibility**

In order to read and write to company records, an API user must have the Read-Write Company permission.

Companies may be read through a single endpoint, Get Companies By Filter Type. Companies may only be filtered on a limited number of fields, which are provided in the searchableFields attribute of the Describe Company result.

Companies can be written to using the Create/Update Companies endpoint.

Companies may be deleted through the Delete Companies endpoint.

**Opportunities**

Primary Key: externalOpportunityId, marketoGUID

externalOpportunityId is an arbitrary string field which is set upon creation by the external system. It is guaranteed to be unique and is not updateable. marketoGUID is a system-generated GUID string which is guaranteed to be unique.

Opportunity objects loosely represent sales deals in Marketo, and are responsible for attributing revenue to leads, companies and programs. Opportunity usage varies between both CRM implementations and organizational implementations, so they may encompass anywhere from a sales deal complete with individual line items to broader records of sales efforts. It may not be possible to accommodate an organization’s use cases for tracking sales deals through the implementation opportunities alone, so additional extension through Custom Objects may be required.

To perform proper Revenue Attribution, Opportunities must be linked to leads via roles, and linked to companies via the externalCompanyId field. For information on Revenue Attribution, please see the <Program-Opportunity Attribution> section.

**Model**

Like leads and companies, opportunities have a flexible and user-extensible schema, which should be routinely updated through the Describe Opportunity endpoint.

**Mapping**
In general, the same mapping strategy should be employed for Opportunities as Leads and Companies, but Opportunities have some special-case fields which should be mapped with care:

- The “Amount” field in Marketo is the field from which all revenue calculations are made and should always be mapped to the Opportunity field which represents the estimated amount of the proposed deal before close, and the realized amount of the deal after close.
- The “Revenue Expected” metric uses the Marketo “Close Probability” field to derive the metric from amount. The Close Probability percentage is multiplied against Amount to return Revenue Expected.
- Stage has a set of calculated values based on the “closed” and “won” fields on opportunity records. Closed should be set to true if the sales deal is no longer being pursued, either because the effort failed, or because the deal was won. Won should be set to true only if closed is also true and the deal has been won. Different CRMs may treat these characteristic differently, and so they may require translation from the system.

externalOpportunityld should be mapped to the primary key in the CRM whenever possible.

externalCreatedDate should be mapped to the CRM system’s canonical creation date for opportunities.

**Relationships**

<table>
<thead>
<tr>
<th>Object</th>
<th>Relationship Type</th>
<th>Link Type</th>
<th>Link Field</th>
</tr>
</thead>
<tbody>
<tr>
<td>Company</td>
<td>N:1</td>
<td>Opportunity Field</td>
<td>externalCompanyId</td>
</tr>
<tr>
<td>Lead</td>
<td>N:N</td>
<td>Opportunity Role</td>
<td>leadId</td>
</tr>
<tr>
<td>Program</td>
<td>N:N</td>
<td>See Program-Opportunity</td>
<td>See Program</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Attribution section</td>
<td>Opportunity</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Attribution section</td>
</tr>
</tbody>
</table>

**Accessibility**

In order to read and write to opportunities, an API user must have the Read-Write Opportunities permission.

Opportunities can be read via the Get Opportunities by Filter Type endpoint. Like companies, they have a limited set of searchable fields which can be retrieved via the Describe Opportunity endpoint.

Opportunities may be created or updated via the Create/Update Opportunities endpoint.

Opportunities may be deleted through the Delete Opportunities endpoint.

**Opportunity Roles**

Primary Key: Compound Key [leadId, role, externalOpportunityId], marketoGUID.

Opportunity roles have a compound key of leadId, role and externalOpportunityId. leadId and externalOpportunityId are relationship fields to their respective lead and opportunities. To insert a record, or update it via its dedupeFields key, all three fields must be passed in the record. marketoGUID is a unique GUID string, which is system-managed and is generated on creation of the record.
Opportunity roles are link objects from lead records to opportunity records. Without this relationship, leads will not be associated to Opportunities in any way. They are also required relationships to perform end-to-end Program Opportunity Attribution.

Model

The schema for Opportunity Roles is non-extensible and should be derived from the Describe Opportunity Role endpoint once. Primary contacts on an opportunity are designated in Marketo by the isPrimary flag on the opportunity role. If the primary contact concept exists in the CRM being integrated, this should be implemented as part of the model and mapping.

Mapping

isPrimary should be matched if there is a concept of primary contacts in the CRM being integrated.

externalCreatedDate should be mapped to the canonical date in the CRM which corresponds to the date that a person became a member of the opportunity.

Relationships

<table>
<thead>
<tr>
<th>Object</th>
<th>Relationship Type</th>
<th>Link Type</th>
<th>Link Field</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lead</td>
<td>N:1</td>
<td>Opportunity Role Field</td>
<td>leadId</td>
</tr>
<tr>
<td>Opportunity</td>
<td>N:1</td>
<td>Opportunity Role Field</td>
<td>externalOpportunityId</td>
</tr>
</tbody>
</table>

Accessibility

To read and write opportunity roles, an API user must have the Read-Write Opportunities permission.

Opportunity Roles can be read via the Get Opportunity Roles by Filter Type endpoint. Like companies, they have a limited set of searchable fields which can be retrieved via the Describe Opportunity endpoint.

Opportunity Roles may be created or updated via the Create/Update Opportunity Roles endpoint.

Opportunity Roles may be deleted through the Delete Opportunity Roles endpoint.

Sales Persons

Primary Key: externalSalesPersonId, marketoGUID

externalSalesPersonId is an arbitrary string field set upon creation by an external system. marketoGUID is a unique system-managed GUID string which is set upon creation by Marketo.

Sales Persons represent an internal entity which has ownership of a set of person records, typically for sales purposes. The information present on these records is used to derive certain formula fields and tokens for lead records in Marketo.

Model

The schema for Sales Persons is non-extensible and should be read once from the Describe Sales Person endpoint.
Mapping

externalSalesPersonId should be mapped to the primary key for the corresponding user or owner object type in the integrated CRM.

Relationships

<table>
<thead>
<tr>
<th>Object</th>
<th>Relationship Type</th>
<th>Link Type</th>
<th>Link Field</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lead</td>
<td>1:N</td>
<td>Lead Field</td>
<td>externalSalesPersonId</td>
</tr>
</tbody>
</table>

Accessibility

In order to read and write to Sales Persons, an API user must have the Read-Write Sales Person permission.

Sales Persons can be read via the Get Sales Persons by Filter Type endpoint. Like companies, they have a limited set of searchable fields which can be retrieved via the Describe Sales Persons endpoint.

Sales Persons may be created or updated via the Create/Update Sales Persons endpoint.

Sales Persons may be deleted through the Delete Sales Persons endpoint.

Extended Synchronized Entities

Custom Objects

Primary Key: marketoGUID, Additional User Defined keys

Custom objects always have a unique system-generated marketoGUID which is set upon creation. There will be at least one additional key, and possibly more, which are user-defined in the custom object definition. Keys may be single fields for types which are linked directly to leads or companies, and may have compound keys for types which are linked to leads or accounts and another custom object type. Keys can be determined by using Describe Custom Object to retrieve the list of dedupeFields.

Marketo allows the definition of Custom Object types by users to extend the Marketo schema. Marketo custom objects may be related to leads or companies, in either a one record to many custom object configuration, or a many-to-many configuration through the usage of intermediate custom objects.
Model

The schema for Custom Objects is dynamic and variable across all subscriptions. Each Custom Object may have any number of fields which may be updated.

Mapping

Custom Object mapping should be driven by customer requirements and can’t be rigidly defined in general.

Relationships

Marketo Custom Objects may only be related to either leads or companies, in either a 1:N or N:N configuration through the usage of intermediate Custom Objects. Relationships can be derived from the relationships parameter of the result of Describe Custom Object.

Accessibility

To read and write to custom objects, an API user must have the Read-Write Custom Object permission. Custom Objects have the standard set of Get, Create/Update, and Delete endpoints. A List Custom Objects endpoint is also provided to give a means of determining what Custom Object types are available in a given subscription.

Activities

Primary Key: id

Activities have a unique integer id that is set upon creation by Marketo.

Activities are records of activities associated to lead records in Marketo. They may record activity of many different types, as indicated by their activityTypeId. Activities are read-only in Marketo. Certain activity types are pruned after 90 days in the Marketo system. Pruned Activities:
• Data Value Change
• Add to List
• Remove from List
• Visit Web Page
• Click Link

The types of activities available in a given subscription are variable depending on many factors, including type of subscription. The available types and their metadata should always be determined by calling Get Activity Types from the target subscription.

Activities are read-only

Model

Activities have a semi-strict schema. The following fields are defined, but not necessarily used for all activity types:

<table>
<thead>
<tr>
<th>Name</th>
<th>Datatype</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>Integer</td>
<td>Unique id</td>
</tr>
<tr>
<td>leadId</td>
<td>Integer</td>
<td>Id of the linked lead. Maps to id on lead records</td>
</tr>
<tr>
<td>activityTypeId</td>
<td>Integer</td>
<td>Id of the type of activity corresponding to a result of Get Activity Types</td>
</tr>
<tr>
<td>activityDate</td>
<td>Datetime</td>
<td>Date that the activity occurred</td>
</tr>
<tr>
<td>primaryAttributeValue</td>
<td>String</td>
<td>Value of the primary attribute</td>
</tr>
<tr>
<td>primaryAttributeValueId</td>
<td>Integer</td>
<td>Id of the primary attribute</td>
</tr>
<tr>
<td>attributes</td>
<td>Array</td>
<td>Array of name/value pairs representing the attributes of the activity</td>
</tr>
</tbody>
</table>

Each activity type has a primary attribute which corresponds to a value of some kind. The attribute may be related to any type of asset or object type in Marketo. For example, the primaryAttributeValue of the Visits Web Page type corresponds to the name or URL of the web page which was visited. The Marketo name will be presented if the page was a landing page, and the URL if it is not a Marketo page. Attributes consists of an array of name/value pairs, naming each of the fields for an activity type and the corresponding value.

Activities may be modeled as either a single object type with an array of attributes, or as individual object types, depending on the requirements or best practices for the external CRM system.

Relationships

Activities in Marketo are always related to lead records through the leadId field. Some activity types may have a relationship to other Marketo assets through their primaryAttributeValue.

Accessibility

In order to read activities, an API user must have the Read-Only Activity permission. To write custom activities the Read-Write Custom Activities permission is required.
Activities can be read through the Get Lead Activities, Get Lead Changes, and Get Deleted Leads endpoints. Custom Activities may be written through the Write Custom Activities endpoint if a custom object type has been defined.

Programs
Programs in Marketo are both a primary organizational asset and a key attribution component. Programs represent marketing campaigns, individual marketing tactics and assets, and operational workflows. Each program may encapsulate marketing assets, Smart Lists (i.e. dynamic segmentation lists) and Smart Campaigns (i.e. workflows). Opportunity-related revenue is attributed to programs based on a lead’s relationship to a program, represented either by the Acquisition Program lead field, or by a lead’s program membership in that program.

It is a common practice in CRM integration for a program not to be directly represented in CRM but to be translated to another type of object. For example, in Marketo’s native SFDC sync, the Program is mapped to the Campaign type, and the Program’s membership is mapped to the Campaign Membership type.

Model
The schema for programs is mostly static. The number of fields available for program records will not change, but the Channels and Tags have user customizable values, which should be modeled if choosing to allow creation of new programs from CRM to Marketo. A program will have the following fields:

<table>
<thead>
<tr>
<th>Name</th>
<th>Datatype</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>Integer</td>
<td>Unique id of the program</td>
</tr>
<tr>
<td>name</td>
<td>String</td>
<td>User-defined name</td>
</tr>
<tr>
<td>description</td>
<td>String</td>
<td>User-defined description</td>
</tr>
<tr>
<td>createdAt</td>
<td>Datetime</td>
<td>Date of creation</td>
</tr>
<tr>
<td>updatedAt</td>
<td>Datetime</td>
<td>Date of last update</td>
</tr>
<tr>
<td>url</td>
<td>url</td>
<td>Url of the program in the subscription</td>
</tr>
<tr>
<td>type</td>
<td>String</td>
<td>Type of the program, one of: Default, Event, Event with Webinar, Nurture, or Email</td>
</tr>
<tr>
<td>channel</td>
<td>String</td>
<td>Channel of the program. Defines acceptable statuses for program membership</td>
</tr>
<tr>
<td>folder</td>
<td>Object</td>
<td>Parent folder of the program</td>
</tr>
<tr>
<td>status</td>
<td>String</td>
<td>Status of the program</td>
</tr>
<tr>
<td>workspace</td>
<td>String</td>
<td>Parent workspace of the program</td>
</tr>
<tr>
<td>tags</td>
<td>Array</td>
<td>Array of user-defined tags. Tags provide customizable information for programs.</td>
</tr>
<tr>
<td>costs</td>
<td>Array</td>
<td>Array of costs. Used to define costs related to running a program.</td>
</tr>
</tbody>
</table>

Mapping
The CRM to be integrated may have an object type which maps to the Marketo program naturally. Programs broadly represent marketing campaigns, and are linked to leads through the Program Membership object which defines the lead’s status, and success in the marketing effort.
Relationships
Programs are related to leads through the Program Membership object. Programs also have other relationships to assets which are not pertinent to the discussion of CRM integration.

Accessibility
To read programs, the Read-Only Asset permission is required. Programs can be read through Browse Programs, Get Program by Id, Get Program by Name, and Get Program by Tag Type.

To write programs, the Read-Write Asset permissions is required. Programs can be created through Create Program, can be updated through Update Program, cloned through Clone Program, and Deleted through Delete Program.

Program Membership
Primary Key: Compound [Id of Program, Id of Lead]

The primary key for program membership is a unique compound key of the id of the linked program and the id of the linked lead. There may only be one program membership record for each unique pair.

Program Membership represents a lead’s relationship to a program and their status in that program.

Model
Program Membership has a strict schema. In relation to CRM some fields may be considered optional depending on customer requirements. Stream, isExhausted and nurtureCadence will only be populated when the membership record is related to a Nurture Program, and may not have external relevance.

<table>
<thead>
<tr>
<th>Name</th>
<th>Datatype</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>progressionStatus</td>
<td>String</td>
<td>Status of the lead in the parent program</td>
</tr>
<tr>
<td>Stream</td>
<td>String</td>
<td>Name of the stream that the lead is a member of</td>
</tr>
<tr>
<td>nurtureCadence</td>
<td>String</td>
<td>Cadence of the lead in a nurture program. Either Normal or Paused</td>
</tr>
<tr>
<td>isExhausted</td>
<td>Boolean</td>
<td>Whether the lead is exhausted in its current Nurture Stream</td>
</tr>
<tr>
<td>acquiredBy</td>
<td>Boolean</td>
<td>Whether the lead was acquired by the parent program</td>
</tr>
<tr>
<td>reachedSuccess</td>
<td>Boolean</td>
<td>Whether the lead is in a success status in the parent program</td>
</tr>
<tr>
<td>reachedSuccessDate</td>
<td>Datetime</td>
<td>The date that the lead became successful in the program</td>
</tr>
<tr>
<td>membershipDate</td>
<td>Datetime</td>
<td>The date the lead became a member of the program</td>
</tr>
</tbody>
</table>

The id of the parent program and is of the parent lead are not explicitly returned by Get Leads by Program Id, but are inferred by the given program id, and the returned parent lead.

Mapping
Like programs, program membership may have a natural mapping to an existing object type, but may also need to be represented by a custom object type in the integrated CRM.

Relationships
Program membership is related to programs and leads by their respective ids.
**Accessibility**

Reading and writing program membership requires the Read-Write Lead permission. Program membership may be read through Get Leads by Program Id, and written through Change Lead Program Status.

**Static Lists**

Primary Key: id

Static Lists have a unique system-managed integer id which is set at creation.

Static Lists represent non-dynamic groupings of lead records in Marketo (opposed to Smart Lists, which are dynamic and update automatically as new leads meet the criteria or existing leads cease to meet the criteria). Static Lists are only created by users from the UI for marketing purposes. Static Lists may or may not be important in external systems depending on customer requirements, and have an extremely broad set of use cases. Static Lists are used to flexibly represent groups of people.

**Model**

Static Lists have a strict schema defined below.

<table>
<thead>
<tr>
<th>Name</th>
<th>Datatype</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>Integer</td>
<td>Unique Id</td>
</tr>
<tr>
<td>Name</td>
<td>String</td>
<td>User-defined name</td>
</tr>
<tr>
<td>Description</td>
<td>String</td>
<td>User-defined description</td>
</tr>
<tr>
<td>programName</td>
<td>String</td>
<td>Name of the parent program if applicable</td>
</tr>
<tr>
<td>createdAt</td>
<td>Datetime</td>
<td>Date the list was created</td>
</tr>
<tr>
<td>updatedAt</td>
<td>Datetime</td>
<td>Date the list was last updated</td>
</tr>
</tbody>
</table>

List Membership is defined as a simple record containing the id of the parent list and the id of the parent lead.

**Mapping**

Static Lists and their membership may have a natural corollary in the targeted CRM.

**Relationships**

Static Lists are related to leads through the list membership object containing the id of both the lead and the list.

**Accessibility**

The Read-Only Leads permission is required to read Static Lists. The Read-Only Lead permission is required to read membership, and the Read-Write Lead permission is required to add or remove list membership.

Lists may be read through Get List By Id and Get Multiple Lists.

Membership may be read through Get Leads by List Id and written through Add Leads to List, Import Lead and Remove Leads from List.
Synchronization

Configuration
To maintain the best possible performance, a set of configuration options should be presented to administrative users.

Synchronized Fields
Not all fields are necessary to be mapped and synchronized between Marketo and an associated CRM system. Admin users should be presented with the option to enable or disable synchronization of specific fields for each entity type where this is appropriate (Leads, Companies, Opportunities, Custom Objects). Only specialty fields as discussed in the entities section should be mandatory for synchronization, while all others should be optional. Reducing the number of synchronized fields will improve synchronization performance in all cases.

Synchronized Custom Objects
Not all Custom Objects are necessary to include in an associated CRM system. Admin users should be presented with the option to enable or disable synchronization of Custom Objects and be made aware of any manual data extension which needs to occur in the CRM prior to enablement of synchronization. Where possible, your server should perform automatic provisioning of associated Custom Objects in the CRM for objects where bootstrapping from Marketo to CRM is being performed. In the case that this is not available, manual and flexible mapping should be permitted to available types and fields. For CRM-to-Marketo synchronization, the definition should be derived from the type in CRM.

Synchronized Activity Types
Many activity types do not have any relevance in a CRM system and should not be synchronized ever, while others may or may not be needed on a client-by-client basis. Admin users should be able to select which types are and are not synchronized to limit API call usage and improve synchronization performance.

Bootstrapping Configuration
An admin user or implementer upon initial configuration should be able to configure which set of fields and record types need to be retrieved from Marketo. An option should also be given to these users allowing them to configure the earliest date from which to synchronize for each particular record type. Initial synchronization time can be reduced dramatically when the earliest time is more recent.

Synchronization Interval
Marketo’s native synchronization connectors queue a new batch of pushes and pulls five minutes after the completion of the previous batch. This covers a great deal of cases and strikes an acceptable compromise between relatively low synchronization latency, and utilization of API calls. For your solution, you should base the synchronization interval based on how many API calls a typical synchronization cycle will take. For accounts provisioned or renewed after March 2016, the default number of API calls per day is 50,000 with optional additional API calls purchased in groups of 10,000/day.
The number of changes that a client expects to occur for their lead records and their synchronized activities will be the greatest influence on the cumulative number of API calls which will be used in a given day, and this should influence your design. Your baseline for continuous synchronization where no changes are retrieved in any given cycle should not exceed 7,500 calls per day.

Bootstrapping from Marketo

If there is existing data in a Marketo subscription which will be required for future use in the corresponding CRM system, then this data should be retrieved from Marketo for the primary objects Leads, Companies, Opportunities, Roles, and Sales Persons.

The bootstrapper code should be separate from the continuous synchronization code, as sometimes bootstrapping will not be necessary at all. Bootstrapping from Marketo should only be performed if the client indicates that there is existing data for a given type in Marketo. Wherever possible, the retrieval should also be focused to the smallest possible set of data required.

Leads

The most effective way to bootstrap all lead records is by retrieving all New Lead activities from the target instance, and then retrieving all of the corresponding leads by Id. The typical process is as follows:

- Call Get Paging Token with a sufficiently early timestamp. This information should be provided by the client.
- Call Get Lead Activities for the New Lead activity, and page through the set of results, storing the corresponding values of leadId in the activity.
- Call Get Leads by Filter Type using id as the filterType, with 300 of the stored Ids for each call. This may be done in parallel with the retrieval of the New Lead activities.
- The resulting lead records may then be stored as mapped.

Companies

The necessity to bootstrap company records from Marketo will be rare as of this writing in April 2016.

Companies should be bootstrapped on the basis of the presence of externalCompanyId on bootstrapped leads. The typical process is as follows: Call Get Companies with filterType of idField. filterValues should be sets of up to 300 values taken from the externalCompanyId fields of the Leads already obtained.

Opportunities and Roles

It should be noted that it is not at all ideal to bootstrap opportunities from Marketo. If there is another available data source to retrieve this data, it is likely the source that should be used, as Marketo does not originate Opportunity data from within the application.

Bootstrapping Opportunities and roles should be done based on the data present in the Add to Opportunity activity type. The typical process to bootstrap opportunities and roles is as follows:

- Call Get Paging Token with a sufficiently early timestamp
- Call Get Lead Activities for the Add to Opportunity Activity and page through the set of results
- If the externalCreatedDate and updatedAt fields are not required, the Role records may be inferred.
• The list of opportunities should be retrieved with Get Opportunities based on the available Oppty ID in the activity results, using idField as the filterType.
• If required the role records should be retrieved with Get Opportunity Roles, based on the leadId, role, and Oppty Id fields, using dedupeFields as the filterType.

Sales Persons
The necessity to bootstrap salesperson records from Marketo will be rare as of this writing in April 2016. Salespersons should be bootstrapped on the basis of the presence of externalSalesPersonId on bootstrapped leads. The typical process is as follows: Call Get Salespersons with filterType of idField. filterValues should be sets of up to 300 values taken from the externalSalesPersonId fields of the Leads already obtained.

Custom Objects
Bootstrapping custom objects from Marketo should be carefully considered. Marketo currently does not expose the ability to pull incremental updates from Marketo custom object records, so the option may present limited usefulness.

The method of bootstrapping from Marketo will depend on the definition of the object. For 1:N relationships with leads or accounts, there will only be one object type to retrieve. For N:N relationships there will be two types. Refer to the Entities section for more specific information on Custom Object definitions.

The typical process for retrieving custom objects is as follows:

• Retrieve the activity type for the custom object. It will follow the naming pattern “Add to {Custom Object Name}”
• Call Get Paging Token with a sufficiently early timestamp
• Call Get Lead Activities for the Add to Custom Object activity type retrieved earlier
• Call Get Custom Objects based on the keys retrieved from the activities. This will need to be cross-referenced to the corresponding Describe Custom Object result.

Programs and Membership
Note: If implemented, it is recommended that membership for programs only be synchronized on an on-demand basis.

To retrieve all programs from Marketo, the client simply needs to call Browse Programs and page through the results by incrementing the offset parameter.

To retrieve all membership from Marketo, the client needs to call Get Leads by Program Id for each synchronized program and page through each result set, retrieving the information in the child membership object.

Activities
Activities may only be retrieved for up to ten activity types at a time, so it is best to maintain separate synchronization queues for each set of up to ten types which have been indicated for synchronization. The typical process is this:

• Call Get Paging Token with the earliest desired time
• Call Get Lead Activities with a set of up to ten activityTypeIds and page through the result set until moreResult is false
• Initiate the previous step for the next set of ten activity types if necessary

Lists and Membership
To retrieve all Lists from Marketo, the client needs to call Get Multiple Lists and page through the result set until the end.

To retrieve all list membership, the client needs to call Get Multiple Lists by List Id for each synchronized list, and page through the result set.

Recommended Bootstrapping Order
1. New Lead Activities since instance creation
2. Lead Records by leadId for New Lead Activities
3. Company records by externalCompanyId from Leads
4. Add to Opportunity activities
5. Opportunities by Opportunity ID from activities
6. Opportunity Roles by leadId, externalOpportunityId, and role from activities
7. SalesPersons by externalSalesPersonId from Leads
8. Other record types as required

Maintaining Synchronization
The simplest and most consistent way to maintain continuous synchronization is to implement a poller which retrieves changes to records in Marketo and pushes them to CRM and then from CRM and pushes them to Marketo, and then repeats the cycle after a predetermined period each time.

To retrieve changes from Marketo, two high watermarks must be maintained, the first for changes to lead/company fields which have occurred since the most recently retrieved change, and for the synchronization case, and the second for the activities which have occurred since the most recently retrieved lead activity. These are datetime values. When retrieving changes or activities in a subsequent synchronization cycle, the exact datetime of the most recently created record should be used. Since Marketo maintains only per-second and not per-millisecond resolution for datetimes, it is possible that activities may occur with the same datetime as the high-watermark. This will result in retrieval of duplicate activities which may be safely ignored based on the id of the activity.

Recommended Synchronization Order
1. Create and Update Companies
2. Get Companies
3. Create and Update Salespersons
4. Create and Update Leads
5. Get Leads
6. Get Lead Merges and Deletions
7. Get Lead Activities
8. Delete Leads
9. Create and Update Opportunities
10. Create and Update Roles
11. Delete Roles
12. Delete Opportunities
13. Delete Companies
14. Delete Salespersons
15. Create and Update Custom Objects

Leads and Companies
Synchronization for Leads and Companies is maintained primarily using the Get Lead Changes endpoint which retrieves data value change records which occur after a timestamp given by a paging token which is retrieved via the Get Paging token endpoint. This endpoint will return both New Lead activities, which indicates the creation of a new known lead in Marketo, and data value change activities for a set of fields given in the parameters of the call.

The change activities should be applied in the order of the createdDate, from earliest to latest, given in the activity to the records in CRM which correspond to the record given by leadld in the activity. New Lead activities should be added to a queue of new lead records which need to be retrieved by id, “leadld” in the activity, using the Get Leads by Filter Type endpoint, with id as the filterType. These may be retrieved up to 300 at a time. It is recommended to wait until there are 300 records to retrieve, and then to make the call to retrieve these records, instead of calling whenever these become available. If, upon reaching the end of the set of changes, there are less than 300 records, then the set should be retrieved.

In addition to changes and new leads, in order to maintain synchronization the Merge Lead and Delete Lead activity must be retrieved in order account for lead records which are merged together. The merge activity indicates that two records have been merged into a single record. The CRM may or may not choose to honor the merge/delete, delete the losing record and retrieve the changes from the winning lead, or it may be ignored and have a “Deleted in Marketo” flag set to indicate that the Marketo Lead ID for that record is no longer valid.

For implementation details on retrieving activities, see the activities section.

As part of a standard synchronization cycle, changes from the CRM should also be retrieved. Ideally only fields which have been updated since the most recent sync cycle should be retrieved, but this may not be possible given the constraints of the system. If this is available, all the changes for a given record should be aggregated into a lead record to be submitted to Marketo. If a changes-only option is not available, then it is viable to retrieve the whole record with all of the Marketo-mapped fields for submission to Marketo.

To push changes for lead records, there are two options, the bulk Import Lead API and the Create/Update Leads endpoint. Import Lead allows a spreadsheet of leads as rows to be submitted into Marketo for creation or update, which may be up to 10MB in size. Create/Update lead allows for the input of up to 300 lead records as JSON. There are advantages to either one. Import Lead is asynchronous and has a higher total throughput, but requires an individual polling implementation to check on the status of the import, and it shares a queue with Marketo list imports, so it is possible for the call to remain enqueued for a significant period of time, resulting in a great deal of latency.
Create/Update Lead offers much faster feedback, but will consume significantly greater numbers of API calls to import large numbers of leads per batch as it may only submit up to 300 records per call.

In either case, for incremental syncing of updates from CRM, the lookupField should be specified as the primary key selected from the CRM system, and the createOrUpdate mode should be used when using Create/Update lead. This allows sharing of the same queue by net new leads and lead updates which need to be pushed into Marketo.

In the case that a Marketo Lead is linked to a company record via externalCompanyId, commonly conceptualized as a contact, the company type fields that were part of the lead record are no longer writeable through the lead record and are deferred to the linked company record.

**Companies**

It is important to determine which fields are Company-type fields, and which fields are Lead-type fields. This can be done with the Describe Company endpoint. All fields listed there are Company-type fields, of which most are mirrored as lead fields for unlinked leads. If a Company-type field is reflected in a Change Data Value operation, then the change should be reflected against the company record in CRM if the change was made against a lead which is linked to a company record via externalCompanyId. If not, the change should just be reflected against the lead record.

**Sales Persons**

*Note:* In a single-CRM case Sales Person records should be managed by the CRM itself, so these records should not be pulled from Marketo after the bootstrapping process.

Sales Person records should be created or deleted in Marketo whenever a corresponding event occurs in the CRM system. This check should be performed by the integration software upon every sync cycle to see if changes are required. In the event that a salesperson record requires deletion, all leads which have that salesperson’s externalSalesPersonId must have that value changed to a new value, or have the value set to null before the record can be deleted.

Synchronization of lead ownership should be performed as part of lead synchronization via the externalSalesPersonId field.

**Opportunities and Roles**

*Note:* In a single-CRM case it is exceedingly rare for Opportunities and Roles to be created by non-CRM sources. It is unlikely to be a requirement to pull these types of records from Marketo.

Opportunity and roles records should be created or deleted whenever a corresponding event occurs in the CRM system. This should be performed as part of a routine sync cycle. If an opportunity requires deletion, all of its child role records must be deleted prior to its deletion. The order of operations for opportunities should be as such:

1. Create and Update Opportunities
2. Create and Update Opportunity roles
3. Delete Opportunity Roles
4. Delete Opportunities
Activities

Lead activities are read-only after creation, and so do not require a two-way implementation. They are read from Marketo via the Get Lead Activities endpoint, in groups of up to 10 activity types. The endpoint accepts an earliest creation date via a paging token which is retrieved via the Get Paging Token endpoint. The set of results should be paged through until the moreResult parameter in the response is returned as false, and the corresponding activities written to CRM. The most recent datetime from the activities retrieved should be stored as the high watermark to begin the next sync cycle with.

Activities are used to retrieve incremental data regarding custom objects.

Custom Objects

To perform Marketo to CRM synchronization of custom objects, the “Add to {Custom Object Name}” activity must be retrieved, and the corresponding records retrieved with the Get Custom Objects endpoint. Custom Objects may have different relationship types which need to be modeled. For more information, please see the Custom Object section in Entities.

Creation, updates, and deletion of Custom Objects should be handled exactly as opportunities or salespersons are.

Program-Opportunity Attribution

Marketo’s most important attribution model is based on the relationship between Leads, Programs and Opportunities, where programs represent some level of unified marketing effort, and opportunities represent sales deals. These relationships are inferred, and the amount of money earned in the closure of an opportunity is attributed to the associated marketing effort, one or more programs, through the associated leads’ program membership record. Marketo has two forms of attribution, First-Touch (FT), and Multi-Touch (MT), which have related but distinct methods of calculation. Care should be taken to ensure that the relevant fields for opportunities and leads are mapped correctly to provide for proper revenue attribution of opportunities.

First-Touch Attribution (FT)

First Touch attribution identifies the programs that are most effective at acquiring the right leads (i.e. new leads that eventually convert to sales opportunities, pipeline, won opportunities or revenue). It does by assigning full marketing credit for the deal/revenue to the program that first brought the lead into the database. This is done by designating Acquisition Program and acquisition date for a lead record and a few other requirements:

- The lead must have an Acquisition Program
- The lead’s acquisition date must occur before the CreatedDate of the Opportunity
- The lead must reach Program Success in the Program

Meeting these prerequisites will give first-touch attribution to the program for Opportunities associated to the lead. These requirements alone should populate the Cost of Opportunity (based on Program Cost), Pipeline Created, Pipeline Created (Still Open), and Revenue Expected measure in a Program Opportunity Analysis report.

To populate the Revenue Won and Revenue to Investment metrics, the Opportunity must be in Stage “Closed Won.”
Multi-Touch (MT)

Multi-Touch Attribution is used to indicate the usefulness of a program in moving known leads toward a purchase (i.e. closed-won opportunities) by spreading credit for the revenue across all the programs that successfully touched the leads on closed-won opportunities. Whereas FT attribution is used to optimize the leads you’re bringing into your database, MT attribution is used to optimize the movement of those leads through the marketing funnel.

An opportunity will have its Pipeline and Opportunities Created measures attributed via MT to a program when a lead reaches success in that program prior to the Created date of the opportunity. This is used to determine the efficacy of a program for generating potential deals from existing leads.

An opportunity will have its Revenue Won, Opportunities Won, and Revenue to Investment attributed to a program via MT when a lead reaches success in a program between the Created Date and the Close Date of an Opportunity. This is used to determine the efficacy of a program in closing deals.