OPENPRISE

White paper

The complete deduplication survival guide

for marketing, sales and support

Table of contents

Introduction 4

Why the need to dedupe 5

Preventing multiple reps from calling the same leads 5

Linking trial users to other program leads 5

Automating sales, marketing, and fulfillment processes 5

Saving money and improving performance of Marketing Automation Platforms 5

People and process considerations 6

Data ownership 6

One-time or continuous dedupe? 6

Dedupe new records, some records, or all of them? 7

Here are some ways you should consider to reduce the problem 7

Which system to dedupe against 7

Pre-deduplication checklist: Salesforce.com and marketing automation data 8

Document your deduplication logic 8

Verify your data sync status between Salesforce.com and your marketing automation solution 8

Check for data verification rules and other automation that may interfere 10

Check for bad data and business processes that may interfere . . . 10

Identifying duplicates in Salesforce.com and Marketo 11

Which data fields to use? 11

How many dedupe fields to use? 12

To fuzzy or not to fuzzy? 13

Determining the surviving records 14

Peel-the-onion logic 14

It's all about your logic 14

Test and iterate in a safe environment 15

Clean and normalize before deduping 15

Table of contents, cont'd.

Merging duplicates **16**

First establish a default logic, then add exceptions 16

Types of merge logic 16

Manual review and overwriting results **19**

When manual dedupe review is not required 19

Legitimate dupes **21**

Brokers and channel partners 21

Preserving historical context 21

Business units and divisions 22

Merge blockers **23**

Merging a lead with a contact 23

Merging contacts with other contacts 23

Invalid record owners 23

Contacts without accounts 23

Invalid or missing field values 24

Violation of validation rules 24

Five key considerations for effective deduplication **25**

1. First, stop the bleeding 25

2. The hard parts are the surviving logic and merge 25

3. One-time project vs. continuous process 25

4. People > process > data > technology—in that order 25

5. It involves more than one system 25

Introduction

Deduplication is one of the most critical data quality improvement processes that every marketing, sales, and revenue operations professional has experienced firsthand. Data deduplication is simple in concept but can be quite complex in execution, especially when dealing with records distributed across multiple systems.

Keep in mind, this isn't a one-time exercise. To be effective, you need to implement deduplication as a continuously running program—because duplicate records can trickle in at any time and from multiple sources, like list imports, broken syncs between systems, and manual record creation.

In this guide, we'll:

- Share detailed data deduplication how-to's and best practices.
- Cover what you need to consider before, during, and after a deduplication project.
- Address the people, process, and system issues inherent in every deduplication project.

Why the need to dedupe

Data geeks like us do it for fun, but most people spend money and effort on deduping because of the huge ROI they get from the following activities.

Preventing multiple reps from calling the same leads

This is probably the number one driver our customers cite in favor of deduping data. When you have duplicate accounts, contacts, and leads, you can easily end up in a situation with multiple account reps and sales development reps calling on the same lead or account. This problem is more acute with a large sales team and a round-robin system for distributing new leads. You can end up with multiple reps working on the same account for an extended time, resulting in sub-optimal account engagement and commission disputes. Worse yet, having an SDR calling an existing customer can also make your company appear clueless and sloppy.

Linking trial users to other program leads

Many software and consumer service products offer a free trial, so anybody can sign up via self-service and kick the tires. This is a proven lead generation tactic that can be extremely productive for the right product and buyer persona. Whether your marketing automation platform (MAP) handles the trial sign-up process or your product handles it, you can quickly generate a massive number of duplicate leads from the trial program. If your trial has a time constraint, you'll likely have leads that have signed up multiple times using different email addresses. Trial users are valuable "mid-funnel" leads where you need to maximize your conversion rate. To accomplish that, you need to correlate trial users' activities across all records and programs, which means you must dedupe.

Automating sales, marketing, and fulfillment processes

There are many good reasons to automate your sales, marketing, and fulfillment processes, and plenty of great software solutions that can help you do so. Deduping helps to streamline your workflow and transactions across different systems and departments. However, automating business processes when your database has a large number of duplicates can cause more trouble than it's worth. Duplicate records can cause redundant transactions and processes, creating confusing and repetitive touchpoints with the customer, and propagating the duplicate data into your finance, order management, and help desk systems.

Saving money and improving the performance of MAPs

Most MAPs are priced according to the size of the database that they use. A large number of duplicates directly costs you money in terms of the license fees you pay. If your duplicate count is especially large, say over 20% of your database, it can negatively affect your MAP's performance. Processes that used to be real time may lag significantly, creating issues with the Service Level Agreement (SLA) you have between your marketing and sales organizations.

People and process considerations

As with most any marketing topic, you usually start with people and process before moving on to data and technology. Dedupe is no different. Here are some of the key people and process questions you should answer before embarking on your dedupe journey.

Data ownership

Deduplication can include lead, contact, and account data. This data often has different systems of record and different owners. Marketing often owns lead data, and the system of record is usually the MAP. Sales generally owns contacts and accounts, and the system of record is generally the Salesforce automation platform. Suppose user data is part of the project scope. In that case, we add product and customer success teams as potential owners and your application and help desk platforms as additional systems of record. This data can be completely separate, partially synchronized, or fully synchronized.

The data owners must agree on the deduplication scheme and process. If the process requires significant time, effort, and budget commitment, the data owners need to be fully committed to the project's success. For example, if sales insists on manually reviewing every account record merge, then the dedupe process must consider how to involve every account executive in the process efficiently.

Data ownership can be a multifaceted issue that includes both departments and data hierarchies. For example, did you know it's possible to have contacts without an account affiliation in Salesforce? These "private contacts" are considered private data to most account reps. If your CRM enables private contacts, should they be included in the dedupe effort?

One-time or continuous dedupe?

Is the deduplication process a one-time (periodic) batch process, a continuous process, or a combination of both? One-time processes involve a massive cleanup that happens periodically, from once a quarter to once every few years. For one-time processes, a manual or semi-automated solution is perfectly acceptable, as long as the solution proposed can accommodate the time and budget requirements. If you plan to continue running dedupe as an ongoing process after the initial cleanup effort, then you'll need to consider automation solutions from the start. A manual or even semi-manual process simply won't be scalable or manageable.

Given your people and process constraints, decide whether a continuous process is realistic. If not, determine how close to an ideal state is acceptable, and consider whether you can supplement it using smaller scale periodic batch cleanups. For example, if sales insists on manually reviewing dedupe results and merging account records, then to have a continuous dedupe process, you must get a Service Level Agreement from the sales team on how quickly they can review the dupes.

People and process considerations

Dedupe new records, some records, or all of them?

It often makes the most sense to separate the two objectives into different processes that work together to create a comprehensive dedupe solution. The new dupe prevention process can run continuously, supplemented by a full dedupe process that runs less frequently, say once a quarter. For example, if list loading is a significant source of lead input for you, then simply preventing dupes from being created while loading a list can dramatically slow the growth of dupes in your database. While fully deduping your database may involve more stakeholders and take longer to figure out, you may be able to implement a dupe prevention process for quick list loading, as marketing has full control over both the data and the process of list loading.

Here are some ways you should consider to reduce the problem:

- If you understand the primary sources of your dupes, try to manage those sources first to "stop the bleeding."
- If multiple databases are involved that are separate or partially synchronized, consider first deduping each database, or parts of the database separately, before combining them one at a time and deduping in phases.
- If a specific subset of the data is more complicated to dedupe due to data quality, people, or process issues, consider leaving them out in the first phase of the deduping effort. Deduping success doesn't require achieving perfection. Having a 95% clean database this month is better than doing nothing and never getting started.

Which system to dedupe against

If the data you're looking to dedupe exists in multiple systems, you need to decide where the dedupe should happen. In general, when data syncs between different systems, one of the systems is the master. This is the system we recommend you dedupe against in most cases.

Note that the master system may not be where the data originated. One typical example is Salesforce contacts vs. Marketo leads. The lead record may have originated in Marketo and then synced with Salesforce. As long as the person remains a lead, Marketo is considered the system of record, and you should dedupe against Marketo. Once the lead converts to a contact, Salesforce now becomes the system of record. In this case, it's best to dedupe a contact directly against Salesforce.

There may be constraints that will limit your choices. For example, you may not have purchased API access (available only in the Enterprise subscription of Salesforce) for the system you wish to dedupe. You may not own or have authorization to change the data in certain systems. In these situations, you may have to dedupe against the secondary system and let the data synchronization propagate the changes to the system of record.

It's a lot easier to prevent new duplicates from being created than to remove existing dupes.

Pre-deduplication checklist: Salesforce.com and marketing automation data

You're excited about getting rid of those pesky duplicate records in your Salesforce and marketing automation solutions (well, we get excited about this kind of stuff). You want to jump right in because those duplicate records have annoyed you for too long, and you want them GONE! GONE! GONE!!!

Before you get started, a little bit of planning can save you a lot of frustration as you progress. Here's a handy dedupe checklist to help with your project planning:

- Document your deduplication logic.
- Verify your data sync status and arrangement.
- Check for data verification rules and other automation that may interfere.
- Check for bad data and business processes that may interfere.

Document your deduplication logic

Deduping is one of those seemingly simple tasks that's actually fairly complex to execute. There are many nuances you're probably unaware of unless you've done it many times before.

The first thing you need to do is think through and document your deduplication logic in both Salesforce and your marketing automation solution. There's no such thing as "generic dedupe logic." If a vendor tells you it has a proprietary algorithm that can magically dedupe your database, you should run fast because your database will probably end up being ruined.

Your dedupe logic must accommodate the following:

- The current state of your data
- The sources of dupes
- The systems and syncing technologies involved
- The controls and automations that are in place
- The people and the processes that your dupes touch (see our blog on this topic: Dedupe Project Considerations: People & Process)

It's worthwhile to audit your various databases to verify if they're actually in sync.

There are three key parts to the deduplication logic you must figure out:

- How to identify the duplicates
- How to select the surviving/winning record
- How to merge the non-surviving/losing records, accounting for system restrictions

Verify your data sync status between Salesforce.com and your marketing automation solution

Whether you're deduping leads, contacts, or accounts, chances are the data set you're trying to dedupe exists in multiple systems. For B2B marketers looking to dedupe leads and contacts, the data usually lives in both your CRM and MAP and possibly other systems like help desk, finance, and customer success platforms. Chances are, they're synchronized to some degree. We often see these scenarios:

1. Salesforce and marketing automation systems are fully synced, so every record exists in both systems.
2. Salesforce solution has more leads than the marketing automation platform because only marketable leads that are CAN-SPAM compliant are in the marketing automation platform.

Pre-deduplication checklist: Salesforce.com and marketing automation data

3. The MAP has more leads than the Salesforce system because only marketing qualified leads (MQLs) are pushed into Salesforce to focus the sales team on hot leads.

4. A combination of scenarios 2 and 3.

Depending on how many systems are involved and how you've implemented syncing, you may be able to get away with just deduping one database and letting the changes ripple through via sync. Or you may need to dedupe each database independently.

If you can dedupe only one database, which one is better? In general, you should dedupe on the system of record for each dataset. For example, you should dedupe leads in your MAP and dedupe accounts in the sales system.

Just because your databases are supposed to be fully synced doesn't mean they are. Many things can cause syncing algorithms to miss records, and they can accumulate to substantial amounts quickly. Here are typical reasons why your data syncs may not be perfect:

- Many syncing and data automation technologies only fire one time when the record is created. Subsequent changes to that record may not get synced.
- Many syncing technologies don't handle deleted and merged records properly, so when one record is deleted in one system, the other systems are unaware that the record no longer exists.
- When syncing is interrupted, backed up, or runs into errors for whatever reason, not all syncing technologies can gracefully recover and resume, resulting in records being out of sync.
- If you have bidirectional syncing, the syncing logic may not be built or configured to be the same in both directions. Sometimes syncing works better or is supposed to work only in one direction.

If you're doing a one-time deduplication exercise, you can work around these conflicts by suspending part of the process for the time being. However, if you're setting up a continuous deduping process, then you'll need to rationalize which technology does what so they don't step on each other.

Pre-deduplication checklist: Salesforce.com and marketing automation data

Check for data verification rules and other automation that may interfere

Your systems may have data verification rules, and other types of features turned on that can interfere with duplicate removal. If so, you'll see these symptoms when you try to merge records:

- A record you deleted in System A is not deleted in System B and becomes orphaned in System B.
- A record you deleted in System A is not deleted in System B and is later recreated in System A by System B.
- A record you updated in System A is partially updated in System B, because some data field updates were blocked by System B. The records remain out of sync forever, or get back in sync the next time System B initiates the sync.

Here are some examples of interfering automations:

Your Salesforce contact record may have required fields that the lead record doesn't have. To merge a lead with a contact record, the lead record must first be converted to a contact. The conversion will fail if the required data is missing.

- If you have duplicate blocking turned on in SFDC, the above scenario will also fail while attempting to convert a lead to a contact.
- Salesforce is the system of record for account data. That means any change to the company information for a lead in Marketo that is a contact in SFDC won't propagate to SFDC. Also, the Marketo company data will revert the next time SFDC initiates a sync.

Check for bad data and business processes that may interfere

Your system may have bad data that can prevent successful deduping. You may never figure out how some of these bad data situations came to be. In some cases, they've been explicitly allowed by your business processes.

Any skilled craftsperson will tell you, "measure twice and cut once." Deduplication is no different. Proper planning upfront can save you a lot of pain, frustration, and damage control later on.

For example:

- A record owner may no longer be a valid user in the target system. Subsequently, the system may reject any attempt to update or merge such a record.
- If you allow a contact record in Salesforce to have no account affiliation (private contacts), any attempt to merge a contact without an account will require additional logic on contact-to-account matching.
- A record may contain an old data value that has since been changed to a picklist. Any attempt to update such a record will require additional logic to reset the outdated record.

Identifying duplicates in Salesforce.com and Marketo

Once you've considered people and process and have your checklist ready, the next order of business is to write down your dedupe logic. The first part of the dedupe logic is how you identify duplicate records in your RevTech database, whether that's Salesforce, Marketo, Pardot, Eloqua, or something else.

Which data fields to use?

The most common data field used by B2B marketers to identify duplicate records is email address, which makes a lot of sense. However, that's just the starting point. Here are a few more options to consider, which can improve your ability to catch those more elusive duplicate records.

Mobile phone number

Mobile phone numbers have evolved into a unique identifier, primarily for these four reasons:

- We're now able to keep our mobile phone numbers when switching phone companies.
- Interstate long distance charges have pretty much disappeared.
- Large metropolitan areas now have overlay area codes, so we have to dial the full 1 + (area code) + phone number even if we're calling within our own area code.
- Company-issued phones are now rare since most people don't want to carry multiple phones.

So now, when people move to a new job or even move across the country, our mobile phone number stays with us, and it's becoming part of our identity. Deduping based on mobile phone number can help you identify contacts across different company affiliations and contacts with different email addresses.

Before you can dedupe records based on the phone number, however, you must first normalize the phone number format. We recommend normalizing all your phone numbers to the international format.

Domain

If company is one of the data fields to dedupe on—whether for account or contact record—consider using domain matching first because it can be tricky to match on company names (we'll cover this next). "Domain" is a more exact way to match companies. For readers not familiar with what a "domain" is, here's an example:

Website: www.usa.acme.com
Email: jdoe@usa.acme.com

Some additional considerations when using domain as a dedupe field:

- In most cases, root domain is the best matching option. Use the full domain if you want to keep divisions of large corporations separate as different accounts.
- You can extract domains from both emails and websites using a RevOps data automation tool.
- Before you extract a domain, it's best to clean up the email and website data, so you don't end up extracting the domain "acme.con" from a bad email address "jdoe@usa.acme.con" with an invalid suffix.

The full domain for the website and email address is "usa.acme.com." The root domain for the website and email address is "acme.com."

Identifying duplicates in Salesforce.com and Marketo

- Filter out email addresses from ISPs (Internet Service Providers), free email providers, and email anonymizer services. Openprise provides a list of these email domains in our Open Data Library.
- A company can own multiple domains, and the domains used for websites and emails may be different. Data providers like Dun & Bradstreet can append your company master record with additional domains.

Company name

For account record dedupe, company name is usually the secondary match field after domain. Company name is often involved in deduping contact records as well, because you can affiliate contacts with multiple companies.

Before you use company name as a dedupe field, we recommend you do the following first:

- Clean up the company name. Instead of trying to match on "Toyota Motors USA Corporation," "Toyota Motors (USA)," "Toyota Motor USA," "Toyota Motors USA Corp.," you can dramatically improve your match rate if you clean up all these names and standardize to "Toyota Motors USA."
- Normalize the company name across its aliases. For example, "Toyota Motors USA" may also appear in your database as "Toyota Motors Sales," "Toyota Motor Sales USA," and "Toyota Cars." Normalizing them all to "Toyota Motors USA" is best. For companies with multiple divisions like "Toyota Forklifts" and "Toyota Financial Services," decide whether to treat these business units as an alias of the parent account or as separate accounts.
- Consider using a data service to normalize the name or use a unique identification code like a DUNS number as the gold standard.

Address

To dedupe on address, you must first clean up and normalize the address. There are just too many different variations of address formats in your database. Pick one of the mapping services as your gold standard and be consistent, whether it's Google, Bing, Here, Tomtom, or USPS, etc. Run your address data through these services to fill in gaps, correct mistakes, and standardize on format.

How many dedupe fields to use?

There's no universally correct answer to this. It depends on your business and your database. That said, here are a few things to keep in mind:

- Use as many matching criteria as you can, even if they yield only incremental results. The only incremental costs are processing resources and time. But with the right automation technology, these incremental costs are trivial. For example, you may try all of these matching criteria on a contact record:
    - Email
    - Mobile phone
    - First name + last name + company name
    - User ID

Identifying duplicates in Salesforce.com and Marketo

- Matching on a combination of data fields may be required if matching on a single field doesn't provide sufficient uniqueness. A couple of examples:
    - A contact may be affiliated with different companies in different roles, like a broker or a service partner.
    - After a contact has moved to another company, you may want to preserve the old contact record to properly archive the historical data associated with the opportunity, enabling a more accurate analysis of win/loss and ideal customer profile.

To fuzzy or not to fuzzy?

Fuzzy matching can be a potent tool for identifying duplicate records. However, no machine algorithm is perfect, so anytime you use fuzzy matching, you're going to be trading off between false positives and false negatives. False positives are incorrect matches found. False negatives are matches that were missed. The general rules of thumb are:

- The more you can clean up and normalize your data, the less you'll require fuzzy matching. So clean up your data as much as you can before deduplication.
- Experiment with the fuzzy factor to decide which configuration yields the best tradeoff between false positives vs. false negatives for your business needs.
- If you're short on time, start with exact matching, then introduce fuzzy matching, and gradually increase the fuzzy factor. This is the most conservative approach.
- There are different fuzzy matching algorithms. Some algorithms are better suited for certain types of data. If your RevOps data automation solution provides different algorithm options, experiment with them.
- You may apply different fuzzy algorithms and factors on different data fields within the same matching criteria, for example:
    - Firstname with fuzzy = 0.6
    - AND lastname with fuzzy = 0.8
    - AND domain with exact match

Determining the surviving records

Once you've identified the duplicate records in your RevTech databases, the next step in the dedupe logic is identifying the surviving/winning records. These are the records you'll keep. The other records are the non-surviving/losing records. You can either merge the non-surviving records into the surviving records or simply discard them.

Determining the surviving records is the most complex part of your dedupe logic.

Peel-the-onion logic

When coming up with your surviving logic, you'll feel as if you're peeling an onion (and we don't mean the "tearing up" part, or maybe we do). Every resolution logic step you write down leads to another question. Here's a very typical example of figuring out the surviving logic in a Marketo lead and Salesforce contact and lead dedupe project.

- If you have both leads and contacts within a group of dupes, then the contacts should survive.
- If there's more than one contact within the duplicate group, then the contact associated with an opportunity should survive.
- If there's more than one contact dupe associated with opportunities within the duplicate group, then the contact record associated with the opportunity at the most advanced stage should survive.
- If there's no contact within a group of dupes, then the leads that have signed up for a free trial should survive.
- If there's more than one lead within a group who've signed up for a free trial, then the lead that has completed specific tasks within the free trial should survive.
- If no leads have signed up for a free trial, then the lead from the most trusted lead source, based on a ranked list of lead sources, should survive.

As you can see, this can get pretty involved, pretty quickly.

It's all about your logic

Now you can see that there's no such thing as a proprietary or secret sauce algorithm that any technology vendor can provide that will magically figure out which one of your records should survive. Every company's logic for this is different, depending on how it conducts its business and what its data sources are. There's no way around this. You must document your surviving record logic; then, you need a flexible technology to execute your logic. Ultimately, this is an exercise in prioritization, which is something that a black box algorithm cannot figure out for you.

We often hear people say there's no consistent logic in some cases because it involves human judgment. We always challenge that claim. Humans don't make arbitrary decisions. When a human makes a one-off dedupe decision between a set of records, they're applying some set of consistent logic in their head, whether they realize it or not—document that logic if your goal is to automate deduplication.

Determining the surviving records

Test and iterate in a safe environment

Your initial logic is likely to have gaps because you haven't finished peeling the onion yet— which is OK. Come up with the most comprehensive logic you can think of, then test it. Make sure the deduplication technology you use can support testing outside of your system of record, whether that's Marketo or Salesforce. To come up with the complete deduplication logic, you'll need to go through at least a few iterations of:

- Running your algorithm.
- Reviewing the dedupe results.
- Making adjustments to your dedupe logic.
- Rinsing and repeating.

It's best to do this type of iterative development and testing within your data management tool or a sandbox. Update your system of record only after you've thoroughly tested your dedupe algorithm.

Clean and normalize before deduping

You can't just jump into a deduplication project with a dirty database. A dirty database can greatly hinder how well your dedupe logic performs. We highly recommend cleaning and normalizing the data fields involved in your dedupe logic first.

For example:

- Clean up bad email addresses like "jdoe@acme.con," so it matches with "jdoe@acme.com."
- Clean up company names like "Acme Corp." And "Acme Corporation" so they match.
- Extract domains from URLs and emails like "acme.com" to use as matching criteria.
- Normalize phone numbers so that "415.555.1212" will match with "+1 (415) 555-1212."
- Normalize lead source names so "Dreamforce 2020" and "DF20" match.
- Clean up and remap old status values like lead and opportunity status.
- Consider integrating more data sources.

In the example above, executing that logic sequence requires more than just your lead and contact data. Specifically, you'll also need:

- Opportunity data from Salesforce
- Opportunity contact role data from Salesforce
- A ranked list of opportunity stages from Salesforce
- A ranked list of lead sources from Marketo or Salesforce

In addition to pulling data sets from your Salesforce and marketing automation platforms, you may even need to leverage data from other systems like your help desk, product database, and finance systems.

If your deduplication logic requires data from other data sets, you'll need a data integration tool to pull the data together. A RevOps data automation application like Openprise combines integration, cleansing, normalization, and deduplication capabilities all in one, greatly simplifying your dedupe project and helping you save money and time using multiple tools.

Merging duplicates

Once you've identified the duplicate records and figured out which surviving records to keep, the last part of your deduplication logic is merging the non-surviving records into the surviving records. In some cases, you may want to simply discard the non-surviving records. That simple scenario requires no further discussion.

First establish a default logic, then add exceptions

Chances are you have more than a handful of fields in the data you're looking to merge, perhaps hundreds. We've seen thousands. To scale, you should first establish a default merge logic to apply to all data fields. Once you have a default logic, you can then define exceptions for specific data fields. The most common default logic is "fill if empty." We'll discuss the various merge logics next.

Types of merge logic

Fill if empty

This logic says that if any data field in the surviving record is empty, then attempt to fill it with a non-empty value from one of the non-fill surviving records. You also need to provide additional logic on what sequence to sort through the non-surviving records. Here's an example of three records in a duplicate set, with the non-surviving records sorted with the more recently updated records on top. The merge logic is "fill if empty using latest modified record."

This is the most common and (not surprisingly) most popular default logic.

Non-surviving 1	J. Doe	Jdoe@acme.com	VP marketing	
Non-surviving 2	John M. Doe	Jdoe@acme.com	Acme Inc.	CMO
Surviving merged	John Doe	Jdoe@acme.com	Acme Inc.	VP marketing

Merging duplicates

Always replace

It applies the merge logic to all the records in the duplicate group, including the surviving record, picks the value that meets the requirement, then replaces the value in the surviving record, empty or not. Common examples include:

- Always take contact information from the last modified record.
- Always take the lead source from the oldest record.

Here's an example of three records in a duplicate set sorted by latest modified date on top. The merge logic for email is to use the latest modified date. The merge logic for lead source is to use the earliest modified date. The default merge logic is "fill if empty."

This is the same logic as the one above, except it doesn't require the surviving record data field to be empty.

Non-surviving 1	J. Doe	Jdoe@acme.com	Acme Inc.	Webinar
Surviving original	John Doe	Jdoe@looney.com		Dreamforce 20
Non-surviving 2	John M. Doe	Jdoe@tunes.com	Tunes Corp.	Free trial
Surviving merged	John Doe	Jdoe@acme.com	Acme Inc.	Free trial

Append

With most merge logic, you throw away some data you believe isn't as good as the data you're keeping. In some cases, you want to keep it all. This is common with unstructured data, like notes or multi-value categories and segmentation data. For these data fields, use append logic. Here's the same example as above, but instead of keeping only the earliest modified lead source, we want to append lead source.

Non-surviving 1	J. Doe	Jdoe@acme.com	Webinar
Surviving original	John Doe	Jdoe@looney.com	Dreamforce 20
Non-surviving 2	John M. Doe	Jdoe@tunes.com	Free trial
Surviving merged	John Doe	Jdoe@acme.com	Webinar, Dreamforce 16, free trial

Merging duplicates

Based on a formula

For numerical or binary data fields, it often makes sense to apply a mathematical formula, like:

- Pick the maximum or minimum value.
- Calculate a sum or an average value.
- Set to "true" only if all records are true.

Here's the same example as above with two numerical fields: behavioral score and demographic score. The merge logic is to pick the highest demographic score, but sum the behavioral score.

			Behavior	Demographic
Non-surviving 1	J. Doe	Jdoe@acme.com	15	100
Surviving original	John Doe	Jdoe@looney.com	10	10
Non-surviving 2	John M. Doe	Jdoe@tunes.com	50	50
Surviving merged	John Doe	Jdoe@acme.com	75	100

Do not merge

This one's simple. For some data fields, you just don't want to merge.

Manual review and overwriting results

So far, we've covered all the necessary planning, setup, and automated dedupe steps, including the first three:

1. Identifying the duplicates
2. Picking the surviving records
3. Merging the non-surviving records into the surviving records

Frequently, the next step is to review and overwrite the automated dedupe results manually. We have a clear and firm position on this. Beyond the initial setup validation and some exceptions discussed below, we believe manually reviewing dedupe results is unnecessary and a waste of human capital. Read the section below to find out why this process can and should be automated.

When manual dedupe review is not required

If there's logic to it, you can automate it

The most frequently-given justification for why people insist on manually reviewing dedupe results is that they want to introduce a human decision maker to the process. Our question to that response is always, "What consistent logic is that human using to make those decisions? Or is it just random judgment?" Almost always, when a human reviews and overwrites the automated dedupe results, they're applying a consistent set of logic. If the automated dedupe results require a large amount of manual correction, then this is due to one of two causes:

- The automated dedupe logic is incomplete.
- The data is so poor that a human has to do additional research and append the data to make a decision.

Suppose the data isn't good enough to support the necessary logic. In that case, we recommend you append the missing data first by either using third-party data providers, or data from your other systems of record. It might sound counterintuitive to spend money to append data that you may throw away after deduping it, but if your data quality is poor and unable to support your dedupe logic, then you may end up keeping the incorrect data and throwing away the good data. That can be more costly than the money and effort spent proactively appending your database before deduping.

If a human reviewer is applying additional logic that the automated dedupe algorithm isn't using, then the simple answer Is to document and incorporate that logic into the algorithm.

Manual review and overwriting results

If there's no logic to it, it won't make any difference

If a human reviewer is indeed making ad-hoc, intuitive, random, eye-test decisions, then you're better off not doing it because the net result will be worse. Plus, you will have wasted precious human resources while gaining nothing in return—if not causing additional damage to your data.

While it's true that with any one specific record, a human action could potentially make it better, there's also an equal chance that the human action could make it worse. If the human decision is indeed "random" because the claim is that there's no consistent logic, then statistics say that over a large enough data set (which most marketing databases constitute), the positive and the negative impacts of the human action are a wash. With 10,000 records or more, the net effect of human action is most likely ZERO. It's just like if you were to flip a coin enough times, you get 50/50 heads/tails.

A much better approach is to focus your valuable human resources on more strategic projects rather than on painful tasks that yields no results and that they're guaranteed to hate doing.

It won't scale, even if it gets done at all

Manual dedupe review is such a painful and slow process that most people tasked with it will procrastinate, procrastinate some more, and procrastinate again after they've finished procrastinating the second time—that's bad for business! They'll do anything else first if they have a choice. Any dedupe project that requires manual review, especially the ones that involve a large number of people—like the sales team—to participate in, will simply never get done. We don't like it, but it's simply human nature.

You have two options:

Option 1: spend the time and effort to "push on the rope" for a long time and never get the project done.

Option 2: move the project forward without the manual review and spend the effort dealing with any complaints and remediations after the fact.

From our experience, Option 2 is much better. It gets results with much lower effort.

When manual dedupe review is required

When you set up the dedupe algorithm, you absolutely should review the results to ensure the algorithm is correct, complete, and working as expected. It's an iterative process, and, as we've shown you so far, a robust deduping algorithm isn't a trivial development and is rarely as simple as you think it will be going into it. But the purpose of the review is not to manually overwrite the results produced by the algorithm, but to provide feedback to improve the algorithm.

When the data set is small and remediation costly

The classic example here is CRM account records, especially strategic/named accounts. This is a small data set, typically no more than a thousand records, with ownership distributed to a rather large set of account reps. So each rep owns about 20 or so strategic account records. Any type of automated deduplication here can create costly (typically not financial, but political) consequences. Thus, the better alternative is to just identify the dupe for the sales team and let the account reps take the manual merge actions for their own small data sets.

Legitimate dupes

Not all duplicate records are created equal, and not all duplicate records should be removed. There are many business situations where it makes sense to keep duplicate records. We like to call these "legitimate dupes." Here are some of the more common types of legitimate dupes.

Brokers and channel partners

Many businesses sell through partners. The Consumer Packaged Goods business uses brokers and distributors. The semiconductor business uses design partners. These channel partners have multiple relationships with retailers.

In the Salesforce world, these partners are contact records that must be associated with multiple accounts. Until recently, SFDC only allowed each contact to associate with a single account. One common way businesses used to sidestep this limitation was to create multiple contact records for the same partner. In this situation, where you want to preserve the partner-to-customer relationships within the contact-to-account construct, make sure you only dedupe contact records within each account and not across accounts.

Accounts. This is an extensive category covering many types of relationships across different industries.

If you decide to keep these partners as legitimate dupes, you should consider syncing (vs. merging) these legitimate contact dupes, so all the duplicate records for the same partner have the same data all the time.

Preserving historical context

When a contact leaves account A for account B, you'll have to decide whether to:

- Create a new contact for account B.
- Change the account assignment for contact from account A to account B.

Marketing often prefers to reassign the contact from account A to account B. This preserves the engagement history with the contact for marketing's records so they can properly attribute their campaigns using the full log of historical data. Also, this reduces the number of dupes, and marketers hate dupes!

Sales, in general, prefers to create a new contact for account B. This is because sales sees each opportunity and account in its own separate context. Any previous conversation with a contact at account A doesn't have much relevance to a new conversation with a contact at account B. Keeping the old contact record with account A also preserves the proper context for account and opportunity history.

Legitimate dupes

As more technologies become available to help the marketing and sales organizations analyze ideal customer profiles, preserving the historical context about how a deal went down and who was involved at the time can be a powerful justification for keeping historical data that represents dupes of the same person.

If you take this need for historical preservation a step further, you could argue that every time a contact changes job, gets promoted, or relocates, you should create a new duplicate contact. That way, you can see that at the time an opportunity was closed years ago, the contact was a manager in marketing, although they're now an executive in business development.

There's no universal right answer to this, but we see a clear split in marketing and sales teams' preferences.

In Salesforce, things get even more complicated if the contact that left account A is now just a lead at a company that isn't even an account yet. Of course, you can't un-convert a contact in Salesforce. In this case, you'll have a dupe across contacts and leads.

Business units and divisions

For account records, every sales organization has a different take on what a duplicate account is. There's absolutely no universal right answer here. It all depends on the structure of your sales organization. Here are some common examples of tricky situations:

- Business units by location: Toyota Motors USA, Toyota Motors Canada, Toyota Motors Mexico
- Conglomerate business units sharing the same name: GE Healthcare, GE Transportation, GE Appliances
- Conglomerate business units not sharing the same name:
    - Alphabet, Google, Waymo, Verily
    - Keiretsu: Mitsubishi Steel, Bank of Tokyo, Meiji Mutual Life, NYK Line

One typical example related to this is ship-to vs. sold-to addresses being modeled as different accounts.

Merge blockers

Depending on the system you're working with, you may encounter situations where your system simply won't allow merging. Here are the most common scenarios we see in Salesforce and how to remediate them.

Merging a lead with a contact

You can't merge records of different object types. Although leads and contacts are both fields about people, they're separate objects in Salesforce.

The remediation is first to convert all the leads to be merged into contacts. During the conversion, make sure to assign the leads to the right accounts, and confirm that all the required fields for the contact objects are filled in. Otherwise, Salesforce will block the lead-to-contact conversion request as well.

Merging contacts with other contacts

Often, duplicate contacts may exist in different accounts, especially if different owners create them. However, you can't merge contacts that exist in different accounts.

If the accounts are duplicates, then the answer is to merge the accounts first, which will then merge the activities, opportunities, and contacts as part of the account merge operation. But if the accounts aren't duplicates, then the only way to merge the contacts is to first move the contacts to the same account before merging.

It's essential to validate the duplicate identification logic with contacts. Using only email is insufficient. At the very least, you'll need to use both email and account.

We always recommend that contact owners review these duplicates as well, as contacts are highly sensitive records for the sales team.

Invalid record owners

Very often, a user who's no longer in the system can still own a Salesforce record. If you try to merge a record assigned to an invalid user, it won't be allowed.

The remediation is to ensure you assign all records to valid users. We recommend handling this as a separate data quality maintenance process, independent of the deduplication process. However, if necessary, search for all invalid users and update the ownership to valid users.

Contacts without accounts

In Salesforce, It's possible to allow private contact records. These are contacts not affiliated with an account. Salesforce will reject any attempt to merge a contact without an account to one with an account.

The solution is to ensure that all contacts to be merged have an account affiliation. This will require an additional step, to assign all contacts without accounts to the same account for the surviving record.

This is essentially a generic problem of invalid dependency that can occur when deduping other objects. For example, an opportunity record must be associated with an account.

Merge blockers

Invalid or missing field values

If the record's field value is outdated, containing a value that's no longer part of the current picklist, any attempt to merge this record will also be blocked.

The solution is to ensure all field values are valid with respect to the current allowable values. This is best handled as a separate data quality maintenance process, independent of the deduplication process. However, if necessary, simply update the field with the valid value.

A related case is missing required field values. This usually happens when a required field is added after record creation, and a default value isn't retroactively assigned.

The answer is similar to the above: either continue maintaining data quality as a separate process or just fill it in before merging.

Violation of validation rules

It's important to understand the validation rules that are active within Salesforce. Merge operations can fail due to validation rules that aren't related to required fields. Since validation rules are unique custom rules in each environment, it's essential to review the active rules within your CRM and determine if merging the various objects would violate any of them.

Five key considerations for effective deduplication

If you've managed to read this far, you're now a dedupe expert! To bring it all together, let's highlight five key takeaways.

1. First, stop the bleeding

Properly resolving duplicates may involve several teams and changing or re-defining processes, which can take a long time to execute. Before you try to fix the existing bad data, look at what it will take to prevent the problem from getting worse. In other words, stop the bleeding. It can often be much easier and quicker to implement without involving as many stakeholders. This will enable you to deliver fast and valuable results and help rally the troops to support the bigger project of cleaning up existing data.

Procrastination and analysis paralysis only makes it easier for the duplicate problem to build up and become increasingly more expensive and time-consuming to fix.

2. The hard parts are the surviving logic and merge

Deduplication isn't a trivial and straightforward task, despite what your solution vendors may say. Vendors like to talk about how comprehensive their duplicate identification algorithm is, but that's the easiest of the three parts. Defining your surviving logic and merge processes are way more difficult steps in comparison. Too many deduplication projects start with a bang, but end with a whimper because of system and process issues when executing these two later steps.

Make sure you select a vendor and a technology that can handle the entire end-to-end process, and not just the identification part. You'll need not only a tool, but in-depth system knowledge for the applications you use.

3. One-time project vs. continuous process

Deduplication often ends up being a one-time project. This is usually because of the complications and efforts required to handle the surviving logic and merge steps.

Unfortunately, when you don't have the right technology and vendor for the job, these two steps frequently end up being manual, making it feasible only as a one-time project. The benefits of making deduplication a continuous process are apparent.

Making it a reality is quite doable as well. Pick the right solution and vendor for the job, put the effort into the first bulk deduplication project, and properly capture and implement your business logic and process. Then, simply keep the automated processes running. But suppose you don't put the necessary effort into automating the process and decide to take the shortcut of relying on a manual resolution. In that case, you're simply putting in a short-term fix. Unfortunately, you'll then see your data deteriorate, and you'll have to endure another large deduplication project down the road.

4. People > process > data > technology—in that order

Duplicate records are the result of gaps and poor alignment between people, process, and technology. To solve the duplication problem, you must first understand the root causes thoroughly to design and implement the appropriate remediation. Otherwise, you'll fail to solve the problem, if not worsen it.

5. It involves more than one system

Most readers of this paper are likely working with CRM and MAPs. These systems are joined at the hip. Data is synchronized at some level, and interactions between these systems can be complex. For such integrated systems, deduplication is fundamentally a cross-system problem. You must have stakeholders from both systems buy in, and your solution must address both systems simultaneously, taking into account the constraints from both sides and understanding the interactions and side effects.

OPENPRISE™

Contact Openprise:
info@Openprisetech.com
(888) 810-7774
www.openprisetech.com

About Openprise

The Openprise RevOps Data Automation Platform fuels company growth by automating hundreds of sales and marketing processes, helping RevOps teams realize the value promised from their RevTech investments. Openprise is a single, no-code platform that can help to simplify even the most complex RevTech stack. For more information, please visit www.openprisetech.com.

WP July 2022