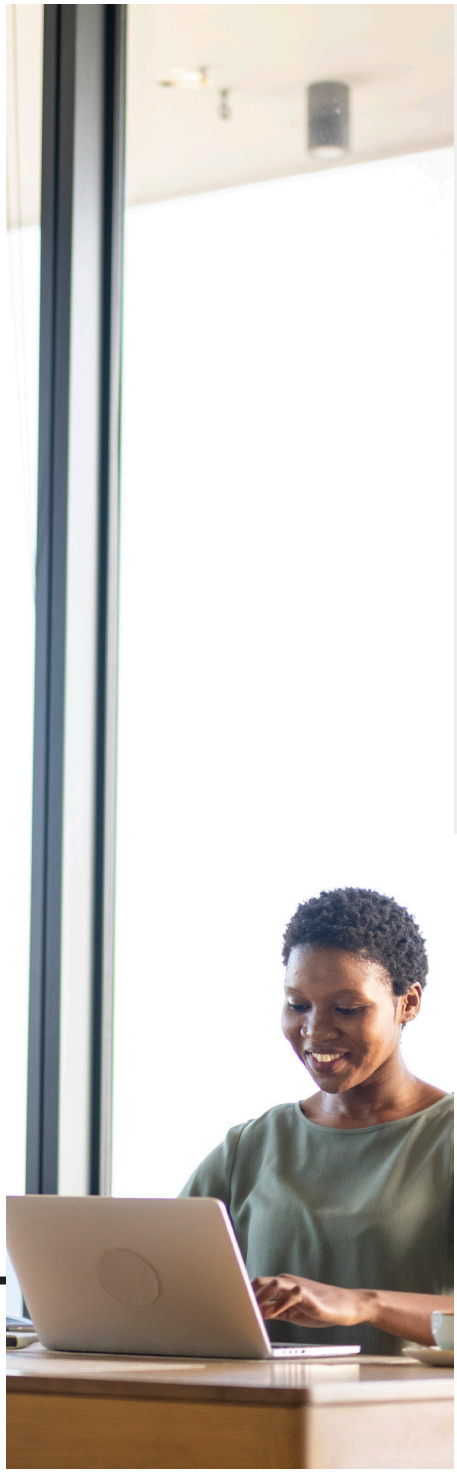
redpath

Raiser's Edge to Salesforce Migrations: Tricky Tables

January 2023

When an organization makes the decision to embark on a journey of digital transformation, they are fostering new and exciting opportunities to help work towards their vision. Moving into a new system, like Salesforce, can open the door for innovative and increasingly effective fundraising strategies. But to use your new system to its fullest potential, your journey of digital transformation will first bring you to an important crossroads in how you manage your data migration. While your old systems might no longer work well for your organization, the data within them still has value even if it needs a little attention first.

Our focus is specifically on preparing organizations migrating from Raiser's Edge (RE) to Salesforce for Nonprofits (including NPSP). Data is what brings your fundraising systems to life. No matter how much effort or money you put into developing a new system, it would be of little use if all the information related to your existing and prospective donors vanished along the way. **Data migration can also be a costly and lengthy part of a new system implementation if you are not well prepared.**



This whitepaper will outline some of the initial steps you can take to ensure you are in the best position to migrate from Raiser's Edge to Salesforce, including identifying:

- What data you have
- How your data is used
- What data needs to be moved to Salesforce
- What data needs to be cleaned or transformed
- What data can be left behind

Understanding the scope and use of your data early on not only helps your organization identify what is most valuable to retain in the move from Raiser's Edge but will also help your implementation partner better understand how to construct and optimize your new system.

Depending on your own role within the organization, these steps might require that you set up meetings with internal teams to understand their current and historical processes and their accompanying data needs.

So, if you are considering a move from Raiser's Edge, now is the time to get started.

GETTING THE LIST OF TRICKY TABLES

Getting started can be overwhelming. You have a lot of data to sift through before you can go-live in Salesforce. Raiser's Edge stores data in tables. Some tables have a simple path to Salesforce and others bring a host of challenges. To keep your project on-track and make sure you don't bring clutter over to your new Salesforce org, we recommend starting with those more challenging tables, which we've nicknamed "Tricky Tables."

While many aspects of Raiser's Edge are used in a similar way from one organization to another, the "Tricky Tables", such as attributes, are used as a way to customize the system for each organization and need some extra time to understand and map the data.

There are two ways to grab the "Tricky Tables" from Raiser's Edge: the back end via SQL, or by reviewing them in the front end. The back end option requires a technical resource on your team, while the front end method is a process any end user can follow. Either way, the end result should be similar to the table below that identifies the data's values and purpose. Let's review two options to to get your list along with SQL queries for the Tricky Tables.

From the Back End

If you have SQL access to your RE instance, you can grab the tricky table info by running the queries in Appendix A. (Set up SQL access by following Blackbaud's instructions here.) Run each query separately to make it easier to organize the values.

Not every query will return data, which just means you're not using that attribute or information type in RE. For each query with results, copy the values into an Excel sheet and note the query it came from.

From the Front End

If you don't have SQL access, you can do this process manually by opening a handful of records in RE. For example, if you're reviewing Constituent Attributes, open a Constituent record, and navigate to the Attributes tab. From there, review the different attribute types. These are unique to your organization and may be different across constituents. Note which values are being used and which ones aren't (more on this in the next section).

Redpath's recommended attribute types to review:

- Constituent Attributes
- Gift Attributes
- Action Attributes
- Appeal Attributes
- Campaign Attributes
- Constituent Address Attributes
- Constituent Bank Attributes
- Education Attributes
- Event Attributes
- Fund Attributes

There are a few other values in RE Constituent records that can be tricky to migrate. To pull values for these tables, navigate to a Constituent record and find the relevant section, then note the possible types of data that can be recorded.

For example, if you're reviewing Phone Types, open a Constituent record and navigate to the contact information. Note each type of contact information a constituent can have, such as "Home Phone," "Work Email," "Mobile Phone," etc.

Sample list of Constituent Information to document:

- Phone, Email, Fax, and other contact types
- Aliases
- Constituent Codes
- Salutations

Now that you've got a list of the tricky tables it's time to start the analysis of your data.

IDENTIFY THE DATA'S PURPOSE

By utilizing a simple strategy aimed at identifying the historical purpose and future value of these data points, you'll be gaining valuable insight into what information is truly important to your end users.

At first glance, you might notice that some data points are showing up in multiple places. Some users may have historically utilized constituent codes, whereas others may have opted to select constituent attributes or even duplicate attribute values that exist within the same category. During this phase, resist the urge to determine the importance of a specific data point: **your main goal should be to identify its purpose, whether historical or current.** Utilizing a simple table, like the following template, work with your end users to jot down anything and everything you can about attribute types and values. It will come in handy in the next phase of this process.

List of Attributes & Codes

Type	Value(s)	# Of Records *(SQL only)*	Purpose/ Definition of Type *(how used)*	Currently Used?
Constituent Attribute: Giving Tier	Major Donor Silver Donor	5000	Defines donor greater than $5,000 all time	X
Constituent Attribute: Membership	Board Member	2	Who are current board members	X
Special Mail Codes	Only 1 Solicitation	157	Used to restrict solicitations	
Salutation	Publication Acknowl-edgement	3000	Sent out in EOY letter	X

KEEP, KILL, COMBINE

Now that you've identified the purposes of the custom data points within your organization, it's time to consider if they have a place within your new database, and whether they need some form of transformation. Throughout this phase of the process, there will likely be voices in your ear letting you know that every single piece of data is important and impossible to do without. But realistically, you'll have to make decisions about which data points to keep, which to combine, and which to "kill".

It's important to make these decisions so that you aren't forced to contend with data volume limitations early on in your move to a new system, and your business processes can be streamlined to help foster user adoption. As an example, bringing over constituent data that is no longer accurate or useful will only serve to confuse and overwhelm users because records and reports will be bloated with duplicate or unnecessary data.

Start by asking questions of yourself and your key end users that can help promote discussion about the value of data and whether it's worth retaining:

- Did this data relate to a specific initiative in the past that is no longer relevant? How recently and how often was this data point utilized?
- Is there confusion amongst users about the purpose of a specific data point?
- Is the type or value of the data point so specific that it creates a long and repetitive data set?
- Has the data point been used consistently over time?
- Can these attributes be combined and renamed to provide greater clarity?
- What process does this data help our organization to facilitate?
- What decisions will you make with this data?

Once you've prompted your users to think about these questions, work with your teams to make decisions to correct any data problems you've stumbled upon. Utilize a table, such as the table below, to start to place the different data types into different categories of importance:

Current Attribute/ Constituent Types	Types to Keep	Types to Combine	Types to "Kill"

The earlier you can begin this process, the better. If possible, you should act within Raiser's Edge to clean up data before providing it for data migration. Here are a few actions you can take to make this happen:

- Delete attribute types and values that are not used or no longer serve a purpose
- Combine attributes that are currently separate but can be combined into a single category
- Rename attributes that serve a purpose but lack clarity

IN CONCLUSION

Deciding to set sail on your digital transformation journey opens the door to various exciting opportunities for your organization, and it is key to take the necessary steps in order for a successful migration from Raiser's Edge to Salesforce. Getting the data conversations started early prepares your team for a smoother technical transition.

Visit our Raiser's Edge migrations webpage for more information on Raiser's Edge to Salesforce migrations.

APPENDIX A

TRICKY TABLES SQL QUERIES

Constituent Attributes

```
select distinct description as attribute, count(description) as numrecords from
ConstituentAttributes c (nolock)
join AttributeTypes a (nolock) on c.ATTRIBUTETYPESID=a.ATTRIBUTETYPESID
group by description
order by DESCRIPTION
```

Gift Attributes

```
select distinct description as attribute, count(description) as numrecords
from GiftAttributes c  (nolock)
join AttributeTypes a (nolock) on c.ATTRIBUTETYPESID=a.ATTRIBUTETYPESID
group by description
order by DESCRIPTION
```

Action Attributes

```
select distinct description as attribute, count(description) as numrecords
from ActionAttributes c  (nolock)
join AttributeTypes a (nolock) on c.ATTRIBUTETYPESID=a.ATTRIBUTETYPESID
group by description
order by DESCRIPTION
```

Appeal Attributes

```
select distinct description as attribute, count(description) as numrecords
from AppealAttributes c  (nolock)
join AttributeTypes a  (nolock) on c.ATTRIBUTETYPESID=a.ATTRIBUTETYPESID
group by description
order by DESCRIPTION
```

Campaign Attributes

```
select distinct description as attribute, count(description) as numrecords
from CampaignAttributes c  (nolock)
join AttributeTypes a (nolock) on c.ATTRIBUTETYPESID=a.ATTRIBUTETYPESID
group by description
order by DESCRIPTION
```

Constit Address Attributes

```
select distinct description as attribute, count(description) as numrecords
from ConstitAddressAttributes c (nolock)
join AttributeTypes a (nolock) on c.ATTRIBUTETYPESID=a.ATTRIBUTETYPESID
group by description
order by DESCRIPTION
```

CONSTITUENT BANK ATTRIBUTES

```
select distinct description as attribute, count(description) as numrecords
from CONSTITUENT_BANKATTRIBUTES c  (nolock)
join AttributeTypes a (nolock) on c.ATTRIBUTETYPESID=a.ATTRIBUTETYPESID
group by description
order by DESCRIPTION
```

Education Attributes

```
select distinct description as attribute, count(description) as numrecords
from EducationAttributes c  (nolock)
join AttributeTypes a (nolock) on c.ATTRIBUTETYPESID=a.ATTRIBUTETYPESID
group by description
order by DESCRIPTION
```

Event Attributes

```
select distinct description as attribute, count(description) as numrecords
from EventAttributes c  (nolock)
join AttributeTypes a (nolock) on c.ATTRIBUTETYPESID=a.ATTRIBUTETYPESID
group by description
order by DESCRIPTION
```

Fund Attributes

```
select distinct description as attribute, count(description) as numrecords
from FundAttributes c  (nolock)
join AttributeTypes a (nolock) on c.ATTRIBUTETYPESID=a.ATTRIBUTETYPESID
group by description
order by DESCRIPTION
```

Phone Types

```
select LONGDESCRIPTION as phonetype, count(phonesid) as numrecords
        from PHONES p  (nolock)
        join TABLEENTRIES t (nolock) on p.PHONETYPEID=t.TABLEENTRIESID
        group by LONGDESCRIPTION
        order by LONGDESCRIPTION
```

Alias Types

```
select LONGDESCRIPTION as aliastype, count(id) as numrecords
        from ALIASNAME a (nolock)
        join TABLEENTRIES t (nolock) on a.ALIAS_TYPE=t.TABLEENTRIESID
        group by LONGDESCRIPTION
        order by LONGDESCRIPTION
```

Constituent Codes

```
select LONGDESCRIPTION as constituentcode, count(id) as numrecords
        from CONSTITUENT_CODES c (nolock)
        join TABLEENTRIES t (nolock) on c.code=t.TABLEENTRIESID
        group by LONGDESCRIPTION
        order by LONGDESCRIPTION
```

Salutations

```
select LONGDESCRIPTION as salutation, count(id) as numrecords
        from CONSTITUENT_SALUTATION s (nolock)
        join TABLEENTRIES t (nolock) on s.SAL_TYPE=t.TABLEENTRIESID
        group by LONGDESCRIPTION
        order by LONGDESCRIPTION
```