

REDCap © Your SAS Friend: Playing around in the API Playground

Brooke Ellen Delgoffe | Marshfield Clinic Health System

ABSTRACT

In a previously published paper, you were introduced to REDCap © (Harris, et al., 2009) as a tool for SAS programmers (Delgoffe & Roush, 2022). Once you've established which of many use cases REDCap © can help you with (and have a functional database), you'll likely want to extract data from it. In this paper, we'll focus more on what data can be exported from REDCap using the API. At conclusion of this paper you should be able to utilize its knowledge to create successful, dynamic, SAS code. Leveraging the "cleaner than Excel" data to learn, innovate, and optimize is just around the corner!

A few macros will be presented to assist with exporting data from REDCap including: simple exports, reports, limited variable exports, metadata, and logging. Select REDCap concepts, such as the data structure, and best use cases will also be covered to assist the SAS programmer in creating efficient export code.

INTRODUCTION

When faced with a REDCap ® project full of data – the task of exporting data comes with as many options as the front end of the project. When creating the database, you should be thinking ahead to include specific data structure but also about which variables are designated for export. It's possible that there is a variety of variables that simply assist in collecting data or making the forms more visually appealing. For example, including descriptive text fields to organize fields into a prettier view for those entering data.

In this paper you will learn about what comes out of REDCap so you have better plan for an export that meets your needs best and select options for export that limit data being exported to only those that are needed.

THE REDCAP API AND API PLAYGROUND

The method used to deliver data from your project is called an API and it uses PROC HTTP to send an HTTP command to the front end of the REDCap database. Of note, it does not query against the database that your REDCap environment is hosted on – such as SQL Server – which is another and separate option best left to the developers. My opinion is it's extremely dangerous to query directly against the host system because you could inadvertently cause a fatal error to the database. Using the API provides a safe and resource friendly method of obtaining the data.

Since not all people are familiar with HTTP strings, REDCap has created a tool that generates the calls needed to perform the desired functions. The API Playground ® will show the user a point and click interface with drop-downs and produce the strings in a variety of languages (Figure 1). Notice that there is not a specific one for SAS, but that is because SAS can use the cURL call. For more information on that part of the interface, refer to my first paper (Delgoffe & Roush, 2022).

API Playground

The API Playground is an interface that allows experimentation with the REDCap API without actually writing any code. You can explore all the different API methods and their various options to customize a given API request. You may even execute a real API request and see the exact response that REDCap returns from the request. If you are interested in creating an API script, the section at the bottom will provide code samples for various programming languages to give you a head start. For details on the capabilities of the REDCap API and how to use it, please see the [REDCap API documentation](#).

Select an API method from the drop-down list below, after which it will load any other options that are specific to that method.

API Method: Export Project Info

Production - Some APIs have been disabled for the playground because they affect data

Format: JSON

Errors: JSON

Raw Request Parameters

Displayed in the box below are all the POST parameters that would be sent in the API request based on the selections above.

```
token: 0FE803FF499ABCD9264DFF185BFEDA02
content: project
format: json
returnFormat: json
```

Response

Click the Execute Request button to execute a real API request, and it will display the API response in a text box below.

Execute Request

PHP Perl Python Ruby Java R C# cURL

Displayed in the box below is the code you would use to execute this API request in the selected programming language.

```
<?php
$data = array(
```

Figure 1. Point and Click Interface of REDCap

Since these PROC HTTP strings are what “do the work” of defining what data is exported, **I will be touching on the differences in this string for each of the covered options.** Downloadable programs that define macros for performing these actions are distributed with this publication. The full code will export a data dictionary, download data, perform data cleaning, and imports/formats the data into a SAS dataset. See Code and Materials for instructions. These examples will show the implications, but all strings can be generated by the Playground and the raw strings maybe copy and pasted into the PROC HTTP in= string. One downside of that method is not being able to fill values dynamically, which is why you will see I’ve embedded macro variables into the strings for the solutions.

UNDERSTANDING DATA STRUCTURE BEFORE THE EXPORT

Using the API playground you can select a variety of output formats, but the data are generally exported as flat files (columns and rows in a single file instead of different tables with merging keys). For the purpose of the paper we will focus on only what would be produced in a CSV file using any one or multiple of the options here.

In the “full data export” without using limits on variables, there will be one row for each record id (identifier field in REDCap) and also one for each repeating instance form (Figure 2). Some forms/surveys in REDCap can repeat. Each repeat is called an instance. Each form or survey has a name There are 3 variables that will be automatically included – record_id, redcap_repeat_instrument, and xzredcap_repeat_instance. All other columns will be fields that come from the project and will be in the order they appear in the project. All forms will be combined in one form so there will be missing data in cells that do not align.

	Automatic Variables			Project Fields (by Order in REDCap)				
	record_id	redcap_repeat_instrument	redcap_repeat_instance	abstractor	abstractor	date_of_abstraction	document_date	birth_record
- Each row has record_id	42007			donnerbl	donnerbl	11/14/2023	7/14/1969	0
- Non-repeat data on first row w/ repeat instrument value missing	42007	contact_information	1					
- New Row for Each Instance	42007	emergency_contacts	1					
							Fields not in that form have all missing values	

Figure 2. View of Standard REDCap Export Structure

REPORTS FOR SIMPLE DATA EXPORTS

Inside the front end of REDCap you have the option to create “Reports” that will list values for records meeting selection criteria. As part of that process you will select what variables you want to display values for and define ordering of records.

STEP 1

User Access: Choose who can edit and view this report

View Access

All users

Edit Access

All users

STEP 2

Fields to include in report

+ Quick Add

Field 1	record_id "Record ID"
Field 2	where_seen "Where did you see this link/
Field 3	otr_where_seen "Where else did you see
Field 4	paper_counter "Paper Download Count"
Field 5	pp_counter "PowerPoint Counter"
Field 6	code_counter "Code Counter"
Field 7	download_counter "Total Downloads Col
Field 8	Type variable name or field label

STEP 3

Filters (optional)

Filter 1: obtain_presentation_materials_comp

AND

Filter 2: Type variable name or field label

Get all fields from selected instrument: -- choose instrument --

tain Presentation Materials	X
tain Presentation Materials	X
tain Presentation Materials	X

Additional report options (optional)

☒ Include the survey identifier field and survey timestamp field(s)?

☒ Combine checkbox options into single column of only the checked-off options (will be formatted as a text field when exported to stats packages)

☒ Remove line breaks/carriage returns from all text data values (only applicable for CSV Raw and CSV Label data exports)

In the report header, display the field label, variable, or both (not applicable for exports)? Both

In the report's data, display the field label, raw data value, or both for multiple choice fields (not applicable for exports)? Both

Figure 3. Preview of the "Create Report" Interface

In the case where you want to download only a subset of data and want to be able to quickly confirm the data in REDCap and/or be able to click into the records to make changes, it may be a good idea to create a report in REDCap first. While the API Playground can help you find an HTTP string that will allow you to do the same actions – it will require you to change it if the report does. For this reason, if you're seeking data that has an operational value using the options for REPORT_ID = will be a better go to.

In the Appendix, see an excerpt of code under Report Downloader C that calls the defined macro %RedCapAPIReport (See Code and Materials for downloadable content) . Below see the PROC HTTP step that is defining what data to extract. Each report is given its own identifier that is viewable in the “Data Exports, Reports, and Stats” area of REDCap (**Error! Reference source not found.**). The Report ID will be used in the call (See highlighted section) to select the report as the source of data to be exported.

My Reports & Exports				
	Report name	Management Options	Report ID (auto-generated)	Unique report name (auto-generated)
A	All data (all records and fields)			
B	Selected instruments (all records)			
1	Who Saw It and Where	Edit Copy Delete	3679	R-389W4A7CXY
+ Create New Report				

Figure 4. Report Listing Denoting Identifier for that Report

```
proc http
    in=&data_string.
    out= my_out
    url =
"token=&secured_token.%NRSTR(&content=report&format=csv&report_id=) &report_id.%NRSTR(&csvDelimiter=&rawOrLabel=raw&rawOrLabelHeaders=raw&exportCheckboxLabel=false&returnFormat=csv) "
;
method="POST";
debug level=1;
run;
```

The green highlighted macro variable `&secured_token.` represents the API Token that will be used like a username and password to access data granted to accounts for the project. For more information on this concept, see REDCap: Your SAS Friend for EHR Manual Abstraction (Delgoffe & Roush, 2022).

LIMITED VARIABLE EXPORTS

While reports are good for getting to the relevant information in existing front end reports, there are also options to do variable and record selection in the HTTP statements. To perform limited variable exports you will add a string for `&fields =` and then list the variables you're interested in using comma delimited format. You do NOT need to have quotes around the variable names.

```
proc http
    in=
"token=&mytoken.%NRSTR(&content=record&type=flat&format=csv&fields=) &fieldlist.%NRSTR(&)"
    out= my_out
    url = "&instance_url."
    method="POST"
;
debug level=1;
run;
```

To find the variable names, you will reference the Data Dictionary or Codebook inside REDCap. You can download the data dictionary using the methods in Metadata Exports below. The Codebook in REDCap

will list variable names along with coded values for those fields (Figure 5). Note that for checkbox fields there are separate binary fields created by the options and so the variable name will be listed by the value and those names will need to appear in the call instead of the checkbox field name.

#	Variable / Field Name	Field Label Field Note	Field Attributes (Field Type, Validation, Choices, Calculations, etc.)
Instrument: Obtain Presentation Materials (obtain_presentation_materials) Enabled as survey Collapse			
1	{record_id}	Record ID	text Checkbox Variable Name
2	where_seen	Where did you see this link/paper/presentation?	checkbox, Required WILSU where_seen__wilsu Wisconsin Illinois SAS Users Group

Figure 5. Project Definition Methods

METADATA EXPORTS

In order to dynamically format your REDCap data you may want to leverage the formatting assigned to the field in the REDCap itself. To do so you will download the “metadata” and place that information in a CSV in a similar manner. Note that field headers and values can contain HTML tags to make them look nice so parsing to remove them may be needed for proper labeling of the field.

In the code snip below, see that we specify the downloaded content as metadata.

```
proc http
  in=
  "token=&mytoken.%NRStr(&content=metadata&type=flat&format=csv&)"
  out= my_dict
  url = "&instance_url."
  method="POST"
;
run;
```

Variable / Field Name	Form Name	Section Header	Field Type
record_id	obtain_presentation_materials		text
where_seen	obtain_presentation_materials		checkbox
email_address_respon	obtain_presentation_materials		text
othr_where_seen	obtain_presentation_materials		text
paper_download_ses	obtain_presentation_materials		descriptive

Figure 6. View of Downloaded Data Dictionary File

This file can also be downloaded directly from REDCap in the Project Design area by selecting the Data Dictionary.

LOGGING EXPORTS

Logging allows you to see the “Who, What, Where, When” of changes to the data. When looking for changes in the data to calculate abstraction time spent adding data to a record or when looking to correct a mistake – this is a life saver.

Time / Date	Username	Action	List of Data Changes OR File Record	
8/24/2024 9:05	delgoffb	Manage/Design	Download data dictionary	
8/23/2024 13:59	delgoffb	Manage/Design	Edit report (report: "Who Saw It and Where	

Figure 7 View of Downloaded Logging File

Note the use of not only the &CONTENT. = log, but also the designation of the period of interest. By leaving calls blank we are indicating to include all valid values.

```
proc http
  in=
  "token=&secured_token.&content=log&logtype=&user=&record=&beginTime=20
  22-10-17 10:14&endTime=&format=csv&returnFormat=csv"

  out= my_dict
  url = "&instance_url."
  method="POST"

;
run;
```

PREPARING FOR AND EXECUTING AN UPLOAD.

Not only can REDCap take download requests, it can also accept files for upload. This is done exactly backwards, by providing a CSV following the structure of the downloads.

Of note, there are instructions in the Import Data area of REDCap (Figure 8) that help you understand requirements for the import file and you can manually import the file to preview a data comparison report that both shows changes and checks syntax for you.

Here are a couple considerations:

- Values must match exactly (case sensitive, using specific names and only previously defined or valid values for fields).
- For repeating instances you can auto number the repeating instance by providing the value "new"
- All multiple choice fields (e.g., dropdown, radio) must have the raw coded value (rather than the choice label) entered in those cells, or else it cannot be processed. These coded values can be found in the codebook.

Figure 8. View of Data Import Utility

CONCLUSION

There are a variety of data conversations that can be had between REDCap and SAS – each providing structured data. The REDCap API and API Playground define ways to export data in many ways: report downloading, limited variable export, and data dictionary (metadata) export being a few. The ability to apply formats, options, and data quality rules make REDCap a great tool, but it's great ability to make that data readily available make it a must have.

REFERENCES

- Delgoffe, B. E., & Roush, S. (2022). REDCap: Your SAS Friend for EHR Manual Abstraction. *Southeast SAS Users Group Conference*. Mobile, AL: SESUG.
- Harris, P. A., Taylor, R., Minor, B., Elliott, V., Fernandez, M., O'Neal, L., . . . Duda, S. (2019, July). The REDCap consortium: Building an international community of software platform partners. *Journal of Biomedical Informatics*, 95, 103208. doi:10.1016
- Harris, P. A., Taylor, R., Thielke, R., Payne, J., Gonzalez, N., & Conde, J. G. (2009, Apr). Research electronic data capture (REDCap) – A metadata-driven methodology and workflow process for providing translational research informatics support. *J Biomed Inform*, 42(2), 377-81. doi:10.1016

CODE AND MATERIALS

I use the following REDCap survey to distribute my presentations, papers, and code for REDCap. This assures that the most recent versions are available and any bugs or shortcomings of the code are posted. Take the survey to download content. You can also elect to provide your email address to receive new versions or materials automatically.

<https://redcap.link/REDCapIsMySASFriend>

ACKNOWLEDGMENTS

I would like to recognize my coworkers Leila Deering and Chris Kadolph for bringing REDCap into our institution and becoming REDCap ® Most Valuable Persons in the worldwide community of REDCap users!

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Brooke Ellen Delgoffe
Marshfield Clinic Health System
Delgoffe.brooke@marshfieldresearch.org
Brooke_degoffe@hotmail.com



SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.

APPENDIX

REPORT DOWNLOADER CALL

```
%RedCapAPIReport(External=0 /*Instance of REDcap */  
  ,RC_PID=%str() /*REDCap Project ID*/  
  ,Report_ID=%bquote() /*REDCap Report ID*/  
  ,BaseDir=%bquote() /*Location you want raw data in*/  
  ,CSVName=LIMITED_FIELDS_RPT/*Name for Raw Data CSV*/  
  ,ReplaceCRLF=1 /*0=Do not remove line feeds 1=remove line feeds*/  
  ,UseProcImport=1 /*0=Only CSV, 1=CSV+SAS dataset*/  
  ,SASDataName=RC_REPORT_DATA /*Name for SAS Data Set*/  
  ,NoLabels=0/*Supresses Labels in SAS Data Set*/  
  ,format_library=0 /*0=no format library 1=format library created*/  
  ,formatlibname= /*name of format library*/  
  ,FormatDataName=RC_REPORT_FORMATTED /*Name of formatted SAS  
dataset*/  
  ,dict_string_path=%str() /*Location of HTTPS string for Dictionary*/  
  ,data_string_path=%str() /*Location of HTTPS string for Data*/  
  )
```