Dynamic Data Pull (DDP)

from an External Source System via Web Services

(Web Service Technical Specifications)

The Dynamic Data Pull (DDP) service in REDCap 5.9.0+ requires the use of two (or optionally three) web services. These web services are not included in REDCap but are to be built by the host institution, in which REDCap will communicate directly with the web services when performing various actions for the DDP. Allowing the host institution to create their own web services provides more control and more flexibility than if REDCap were forced to make direct connections to a database, for instance. REDCap knows how to talk with the web services because the communication from REDCap to the web services (and vice versa) is standardized. Described below are the detailed technical specifications for each web service, including the HTTP request method type, parameters (and their format) sent by REDCap to the service, as well as the format/content of the response body received by REDCap from the service. Once these web services are set up and the DDP has been enabled in the REDCap Control Center, REDCap will then be able to communicate to the services successfully to perform the various functions for the DDP in any DDP-enabled project.

How to secure the communication between REDCap and your DDP web services:

There are two methods for securing the transactions between REDCap and the web services: 1) using an IP whitelist so that only your REDCap server(s) can access the web services, and/or 2) by passing a secret hash in the query string to the web service (e.g., https://domain.org/redcap/data_webservice.php?secret=E32apCs211e8s), which would be defined when you set the URLs of the web services on the DDP setup page in the REDCap Control Center. Using both 1 and 2 together would provide the highest level of ensuring the authenticity of the request as having originated from REDCap.

List of the common Post parameters sent to all 3 web services:

Please note that **all three web services will be called via an HTTP(S) Post request, and all will be sent the three Post parameters listed below**. Please note that requests to the user access web service and metadata web service will be sent as multipart form data POST requests (content-type=application/x-www-form-urlencoded), while requests to the data web service will instead be sent as JSON-encoded POST requests (content-type=application/json) and will also contain the additional parameters “id” (string) and “fields” (array).

1. “**user**” (string) – The current user’s REDCap username. *Note: If the REDCap cron job is fetching the data from the data web service (rather than an authenticated user), then this value will be blank.*
2. “**project_id**” (string) – The unique ID number of the REDCap project (i.e. the “pid” value found in the URL when accessing the project in REDCap).
3. “**redcap_url**” (string) – The base web address to REDCap (URL of REDCap's home page).

Specs for the Metadata Web Service

- Description: REDCap will make requests to this web service when end-users are in the DDP mapping/setup module when they are mapping fields from the external source system. This web service provides a list of all fields (and their attributes) that are available from the source system. No input is sent to this web service from REDCap.
- Request method: Post
- URL: doesn't matter
- Post parameters sent: “**user**”, “**project_id**”, “**redcap_url**” (see details above)
- **Return format:** JSON-encoded string of items with each item returned having the following attributes
  - Unique field name called “field” (must be only lower or capital letters, numbers, underscores, and dashes) - REQUIRED
  - “label” (short text describing the field) – default *blank/null*
  - “description” (extra info in addition to label) – default *blank/null*
  - “temporal” (“1” if the field has multiple values repeated over time, “0” if not) – default “0”
  - “category” (general group, e.g. Demographics, Vitals, Labs) – default *blank/null*

    - “subcategory” (smaller group within a given category) – default *blank/null*
    - “identifier” (“1” if the field is the record identifier field for the source system – e.g., MRN) – default *blank/null*
- Example of response text:
```
[{"field":"mrn","label":"Medical Record Number","description":"Patient's MRN from
StarPanel","temporal":0,"category":"Demograghics",”identifier”:”1”},
{"field":"dob","label":"Date of Birth","description":"Patient's date of birth (Y-M-D
format)","temporal":0,"category":"Demograghics"},
{"field":"GluWB","label":"GLUCOSE WHOLE BLOOD","description":"Glucose whole blood yada yada
yada","temporal":1,"category":"Labs",”subcategory”:”Glucose Tests”}]
```

Specs for the Data Web Service

- Description: REDCap will make requests to this web service when pulling data from the source system for an individual record. REDCap will specify in a standardized JSON format the data requested from the source system, and the web service will then return the data available for the record from the source system in a standardized JSON format, which will then be interpreted by REDCap. Specifically, REDCap will send the “id” value (i.e., the value of the external source system’s ID field) and the unique name of the source system field for which it is requesting data. For any source fields that are defined in the Metadata Web Service as “temporal” fields, REDCap will additionally send a “timestamp_min” and “timestamp_max” value to note the relevant window of time in which the data will occur in the source system. NOTE: While the requests sent from REDCap to the metadata web service and user access web service are a multipart form data POST requests (content-type=application/x-www-form-urlencoded), contrarily requests sent from REDCap to the data web service will instead be JSON-encoded POST requests (content-type=application/json).
- URL: doesn't matter
- JSON data (top-level fields in the JSON-encoded content sent in the Post request):
    - “**user**” (see details above)
    - “**project_id**” (see details above)
    - “**redcap_url**” (see details above)
    - **"id"** (string) – The value of the external source system’s ID field (e.g., MRN)
    - **“fields”** (array) – List of fields to search for in the external source system. Each will have the following attributes:
        - “field” – The unique name of the source field, as defined in the metadata web service.
        - “timestamp_min” – (attribute is only present if the field is temporal, as defined in the metadata web service) The lower limit for the time range to search within in the source system. The value will always be sent in YYYY-MM-DD HH:MM:SS format. This value is calculated automatically by REDCap by subtracting the day offset value from the temporal field’s timestamp.
        - “timestamp_max” – (attribute is only present if the field is temporal, as defined in the metadata web service) The upper limit for the time range to search within in the source system. The value will always be sent in YYYY-MM-DD HH:MM:SS format. This value is calculated automatically by REDCap by adding the day offset value to the temporal field’s timestamp.
    - Example of JSON-encoded content sent by REDCap to data web service:
```
{"user":"taylorr4",
"project_id":394,
"redcap_url":"http:\/\/10.151.18.250\/redcap_trunk_active\/redcap\/",
"id":"123456",
"fields":[
   {"field":"dob"},
   {"field":"gender"},
   {"field":"glucoseTolerance",
   "timestamp_min":"2013-09-03 10:51:00", "timestamp_max":"2013-09-07 10:51:00"},
   {"field":"glucoseTolerance", "timestamp_min":"2013-09-05 00:00:00",
   "timestamp_max":"2013-09-09 00:00:00"}
]}
```
- **Return format:** JSON-encoded string of items with each item returned having the following attributes
    - “field” – The unique name of the external source field, as defined in the metadata web service.
    - “value” – The data value for the field from the source system.

        - "timestamp" – (attribute is only present if the field is temporal, as defined in the metadata web service) The date or timestamp value denoting the time when the data value was recorded in the source system. The timestamp can be in YYYY-MM-DD, YYYY-MM-DD HH:MM, or YYYY-MM-DD HH:MM:SS format.
    - Example of response text:

```
[{"field":"gender","value":"0"},
{"field":"dob","value":"1994-09-09"},
{"field":"race","value":"3"},
{"field":"firstName","value":"Rob"},
{"field":"lastName","value":"Taylor"},
{"field":"weight","value":"190.3","timestamp":"2013-09-05"},
{"field":"glucoseTolerance","value":"181","timestamp":"2013-09-01 14:32"},
{"field":"glucoseTolerance","value":"124","timestamp":"2013-09-04 06:55"},
{"field":"glucoseTolerance","value":"105","timestamp":"2013-09-05 08:23"},
{"field":"glucoseTolerance","value":"91","timestamp":"2013-09-05 10:09"},
{"field":"otherglucose1","value":"101","timestamp":"2013-09-05"}]
```

- Other notes: When parsing the response from the data web service, REDCap will automatically ignore any values returned that are outside of the date/time range for a field, given its timestamp value returned and the day offset value. It is recommended that the data web service only returns values within the appropriate date/time range for a given temporal field (i.e., between timestamp_min and timestamp_max), but if it does return any outside the range, REDCap will filter them out automatically.

Specs for the User Access Web Service (optional)

- Description: REDCap will make requests to this web service when validating if the current user has proper authorization for adjudicating DDP data and/or accessing data from the source system. This web service will only be called once per project per session. If the user is determined by the service not to have access, then they will be given an error message if they attempt to adjudicate any data from the DDP. Note: This service is not mandatory but is only optional for extra security.
- Request type: Post
- URL: doesn't matter
- Post parameters sent: "**user**", "**project_id**", "**redcap_url**" (see details above)
- **Return format:** Boolean, either "1" if they have access to the source system (e.g., Epic, StarPanel) or "0" if not.