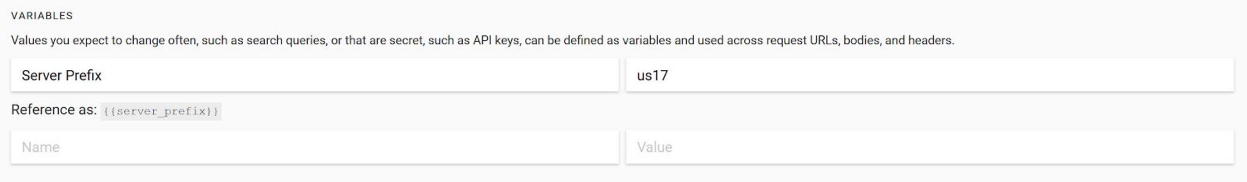
Precog Universal API Connector

Precog's Universal API Connector provides a webform for configuring access to any REST API.

To demonstrate how to configure a new API Connector, we'll walk through each of the configuration steps.

Variables

User-defined *Variables* can be used anywhere in the connector configuration. We recommend defining a *Variable* for any value you'll need to use more than once.



VARIABLES

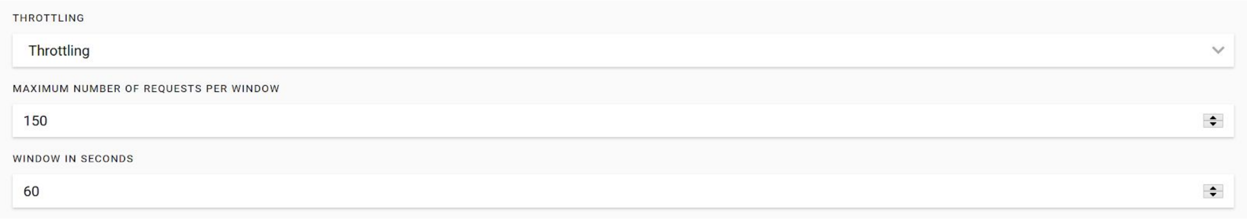
Values you expect to change often, such as search queries, or that are secret, such as API keys, can be defined as variables and used across request URLs, bodies, and headers.

Server Prefix: us17

Reference as: `{{server_prefix}}`

Throttling

Precog will automatically throttle requests so as not to exceed the rate limit prescribed by the API. If the API you're configuring enforces a rate limit, enter that information here.



THROTTLING: Throttling

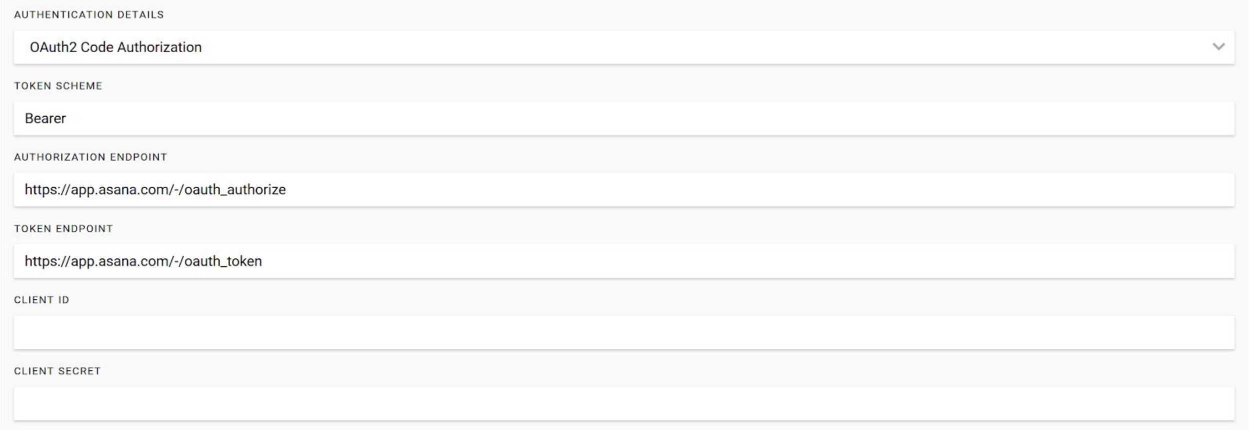
MAXIMUM NUMBER OF REQUESTS PER WINDOW: 150

WINDOW IN SECONDS: 60

Authentication

Precog supports authentication via Basic Authentication, OAuth 2.0, and Request Token Authentication. In this example, we consider the OAuth 2.0 authentication flow.

Enter the *Authorization Endpoint* and *Token Endpoint*, as specified by the API documentation. Then enter your *Client Id* and *Client Secret*, which are generally provided when you create a new application in the API's developer portal.



AUTHENTICATION DETAILS

OAuth2 Code Authorization

TOKEN SCHEME

Bearer

AUTHORIZATION ENDPOINT

https://app.asana.com/-/oauth_authorize

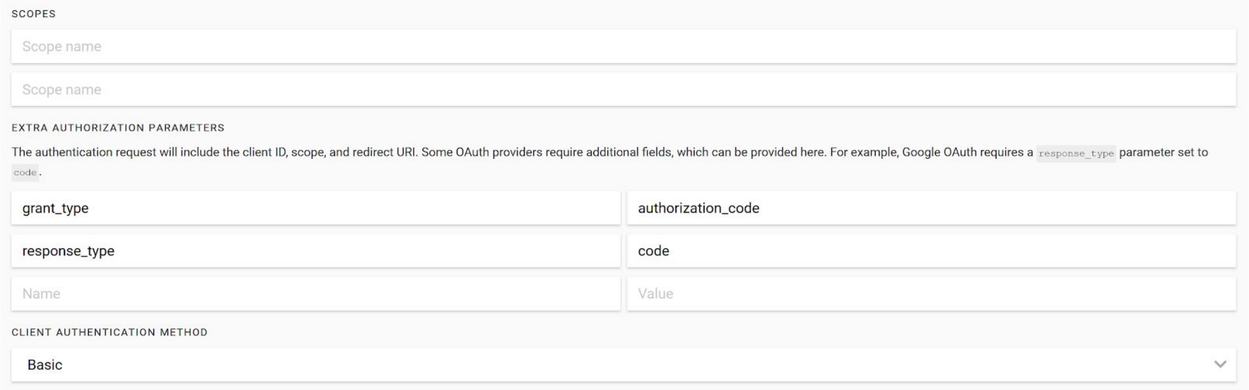
TOKEN ENDPOINT

https://app.asana.com/-/oauth_token

CLIENT ID

CLIENT SECRET

Then enter any additional *Scopes* and *Authorization Parameters*, as required per the API documentation. Select the *Client Authentication Method* that will be used during the authentication flow.



SCOPES

Scope name

Scope name

EXTRA AUTHORIZATION PARAMETERS

The authentication request will include the client ID, scope, and redirect URI. Some OAuth providers require additional fields, which can be provided here. For example, Google OAuth requires a `response_type` parameter set to `code`.

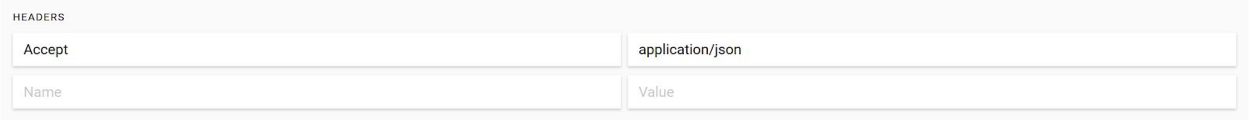
grant_type	authorization_code
response_type	code
Name	Value

CLIENT AUTHENTICATION METHOD

Basic

Headers

Request headers can be defined globally and per dataset. In this section, define the request headers that will be applied globally (i.e. to each API request).



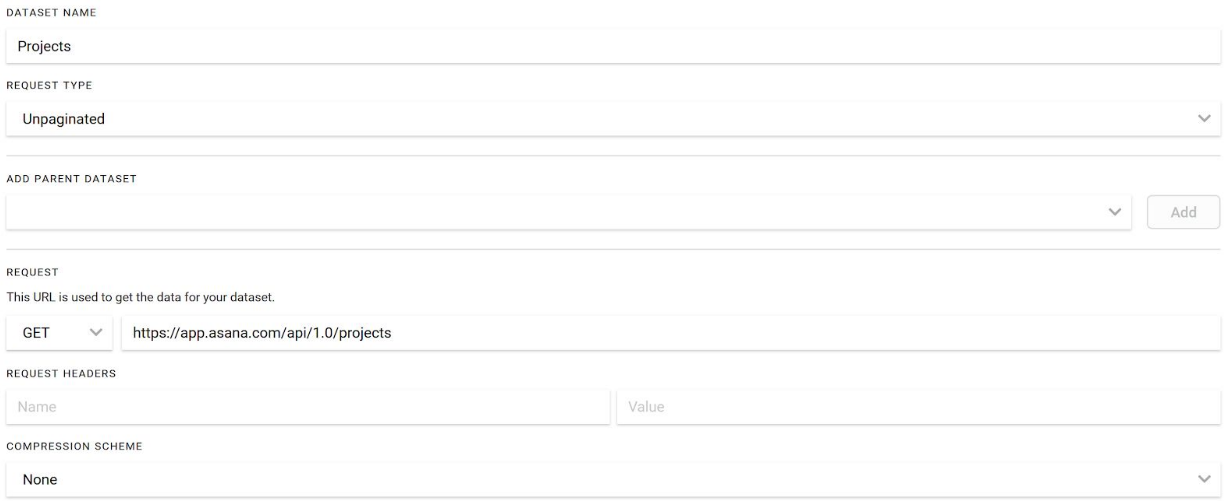
HEADERS

Accept	application/json
Name	Value

Datasets - Parent Datasets

Next, we configure the datasets. In this example, each dataset connects to an Asana API endpoint.

Here we create the Projects dataset, which provides data about all the Asana Projects available.



DATASET NAME

Projects

REQUEST TYPE

Unpaginated

ADD PARENT DATASET

REQUEST

This URL is used to get the data for your dataset.

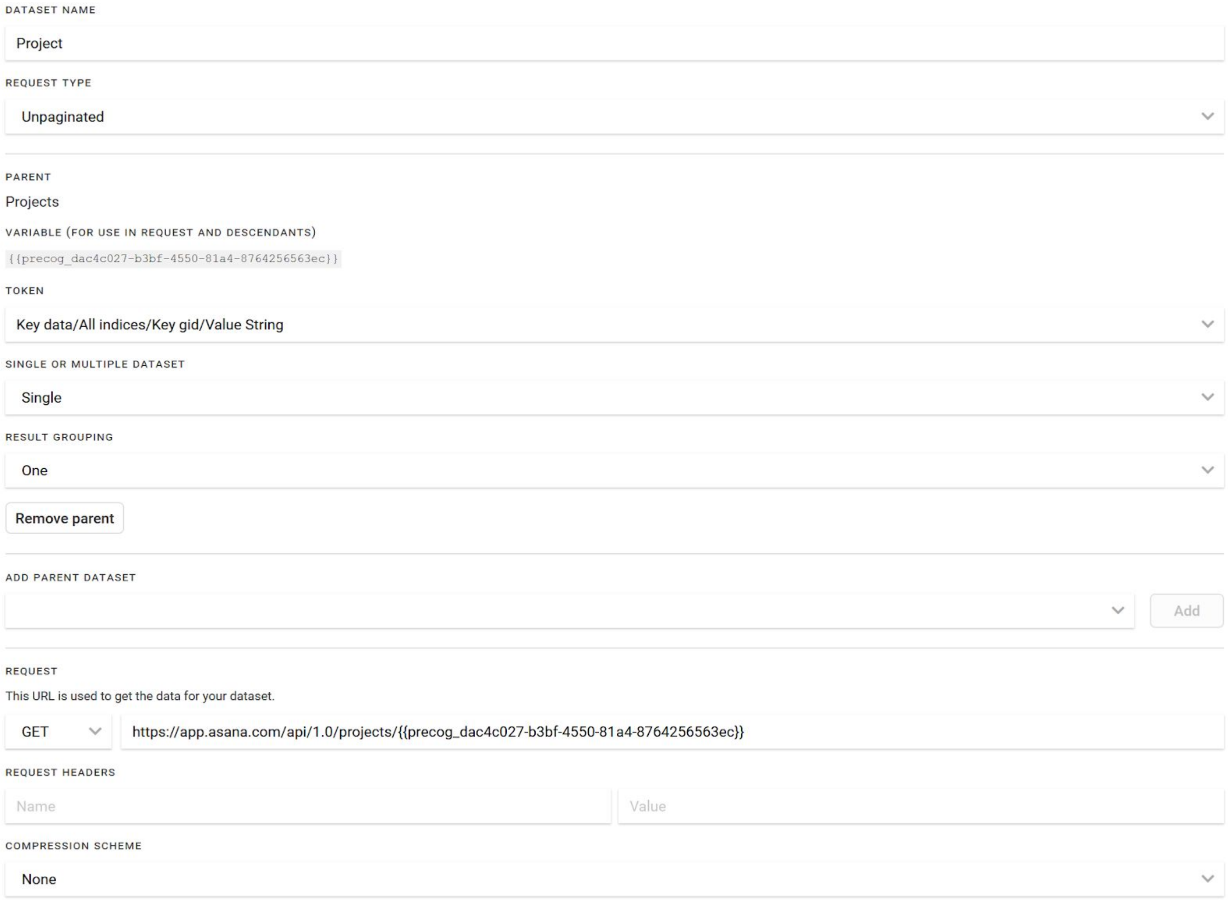
GET https://app.asana.com/api/1.0/projects

REQUEST HEADERS

COMPRESSION SCHEME

None

Now that we have access to all the Asana Projects, we'd like to gather information about each of the Projects individually. To achieve this, we can use the Projects dataset as a *Parent Dataset* for a subsequent API request.



DATASET NAME
Project

REQUEST TYPE
Unpaginated

PARENT
Projects

VARIABLE (FOR USE IN REQUEST AND DESCENDANTS)
{{precog_dac4c027-b3bf-4550-81a4-8764256563ec}}

TOKEN
Key data/All indices/Key gid/Value String

SINGLE OR MULTIPLE DATASET
Single

RESULT GROUPING
One

ADD PARENT DATASET

REQUEST
This URL is used to get the data for your dataset.
GET https://app.asana.com/api/1.0/projects/{{precog_dac4c027-b3bf-4550-81a4-8764256563ec}}

REQUEST HEADERS

COMPRESSION SCHEME
None

Here we see that the Project dataset is configured with the Projects dataset as its *Parent Dataset*. This configuration specifies that each Project Id value (which Asana calls a *gid*) will be passed individually into the endpoint

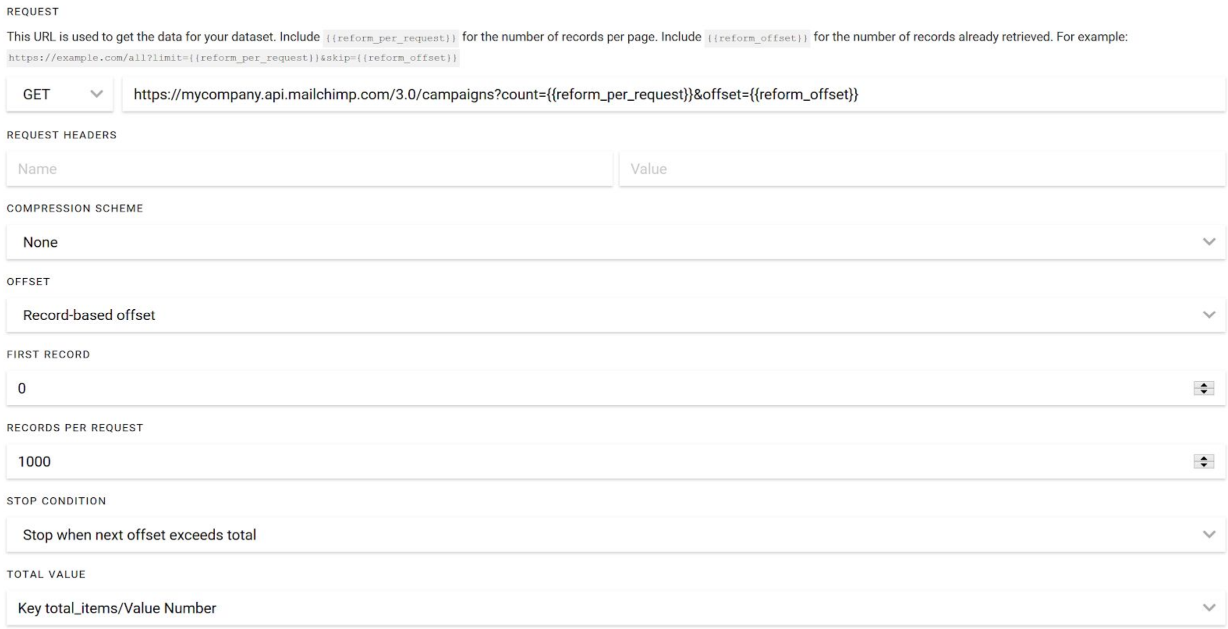
https://app.asana.com/api/1.0/projects/{{precog_dac4c027-b3bf-4550-81a4-8764256563ec}}

in place of the variable {{precog_dac4c027-b3bf-4550-81a4-8764256563ec}}. This will produce one dataset that contains the detailed information about all the Asana projects.

Datasets - Pagination

API endpoints often return results spread across multiple pages. To support these APIs, Precog supports various kinds of pagination. In particular, Precog supports: *Token Pagination*, *Offset Pagination*, and *Header Pagination*, each of which include various configurations of their own.

Here we see the *Offset Pagination* configuration for Precog's MailChimp connector. Precog will automatically make one request per page until there are no more pages available.



REQUEST

This URL is used to get the data for your dataset. Include `{{reform_per_request}}` for the number of records per page. Include `{{reform_offset}}` for the number of records already retrieved. For example: `https://example.com/all?limit={{reform_per_request}}&skip={{reform_offset}}`

GET

https://mycompany.api.mailchimp.com/3.0/campaigns?count={{reform_per_request}}&offset={{reform_offset}}

REQUEST HEADERS

COMPRESSION SCHEME

None

OFFSET

Record-based offset

FIRST RECORD

0

RECORDS PER REQUEST

1000

STOP CONDITION

Stop when next offset exceeds total

TOTAL VALUE

Key total_items/Value Number

Support for New Connectors

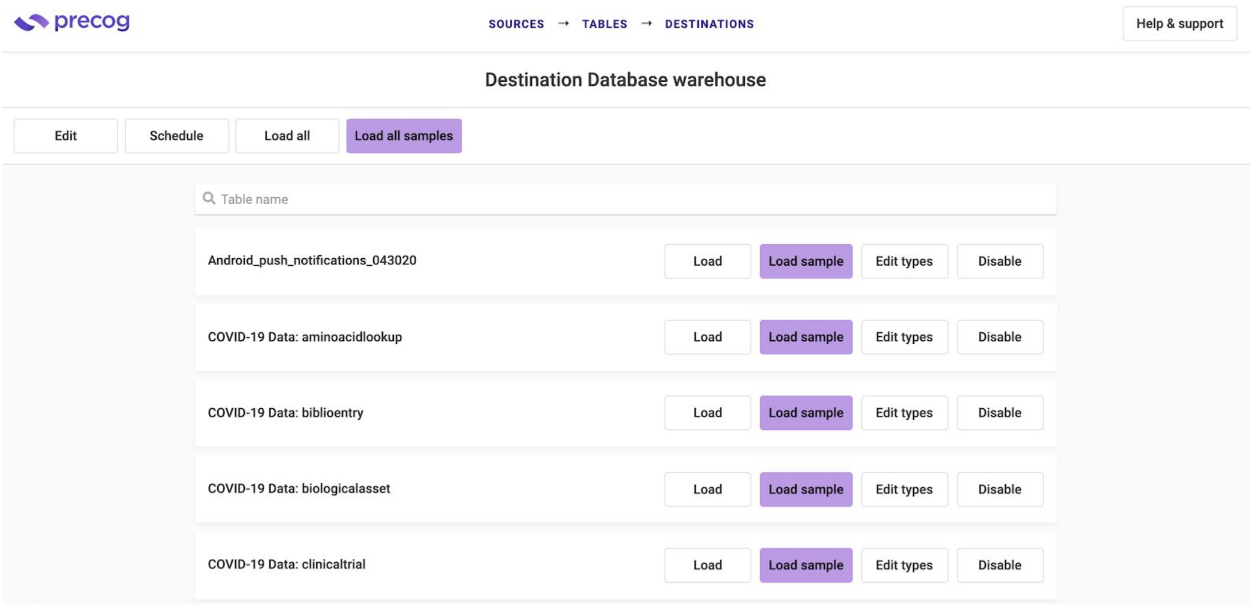
We love supporting customers as they configure new API Connectors. If you're working on configuring your own API Connector and you'd like some help, we'd love to jump on a call with you.

Additionally, we're always adding new pre-configured API Connectors as well as adding new features, so let us know if there's a Connector or feature you'd like to see added!

Summary

Once a new connector is configured Precog will automatically create tables from the source data. These are fully SQL ready tables you can use in any BI or ML tool, load into a Data Warehouse or simply download as a CSV for offline use.

The “magic” of Precog is you don’t need to do any manual mapping or parsing of the API data, simply pick from the list of tables presented in the Precog UI and start using it with your favorite tool. And one of the best features is these tables are all fully customizable. You can add or remove new columns easily thru the UI to get the exact data you need in minutes.



Destination Database warehouse

Android_push_notifications_043020

COVID-19 Data: aminoacidlookup

COVID-19 Data: biblioentry

COVID-19 Data: biologicalasset

COVID-19 Data: clinicaltrial

Precog can manage all your tables and load them on a schedule from every 30 seconds to once per day and everything in between, or you can choose to load them anytime by simply choosing “Load All” from the menu. When you load the data Precog will automatically fetch the most recently updated data from the API.

Now anyone can access any API and get analytic ready tables in minutes that work with your favorite BI tool, Machine Learning tool or any Data Warehouse or Database.