

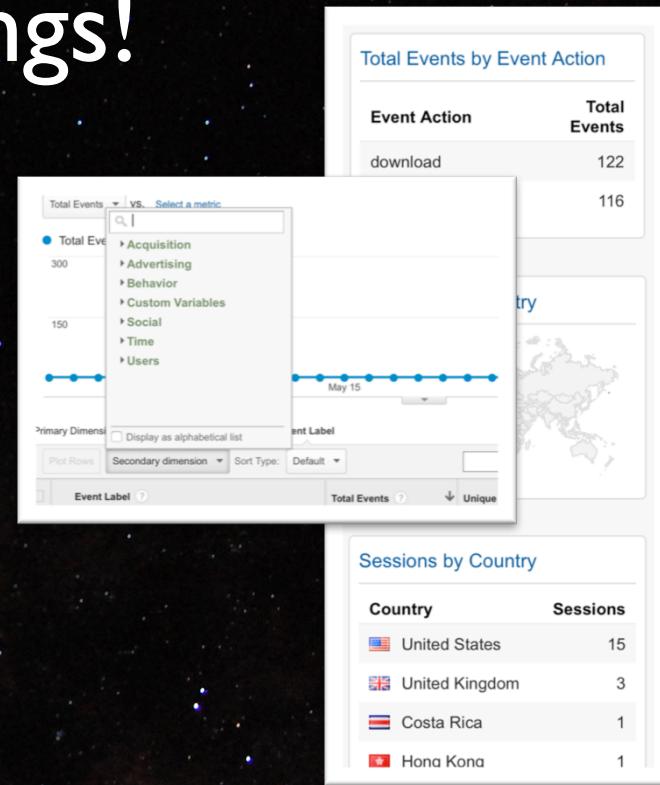


Beyond the Dashboard
Customized Analytics
Reporting with Google APIs
Open Repositories, June 2016

Liz Krznarich, Software Engineer/UI Design, ORCID
e.krznarich@orcid.org <http://orcid.org/0000-0001-6622-4910>

Google Analytics UI is great for lots of things!

- Ooo, pretty!
- No coding needed!
- Integrations aplenty



...but it has limitations

- No data for items with no interactions
- I want to add data from other sources!
- I have some crazy use cases!

Fortunately, there's an API for that!



analytics + drive + sheets = infinite possibilities

In this demo we'll:

- Query the Analytics API
- Set up API credentials
- Upload Analytics data to Drive (in a spreadsheet)
- Add data from other sources

I. Build a website

<http://orcid.github.io/or2016-ga>

The Annals of ORCID

Search...

Home

Analytics

Get the code!

Repository Stuff

ORCID: a system to uniquely identify researchers



Type: Publication (Journal article)
Date: 1 October 2012
Authors: Laurel L. Haak, Martin Fenner, Laura Paglione, Ed Pentz, Howard Ratner
Identifier: 10.1087/20120404

[Download](#) [View](#)

ORCID Public Data File 2015



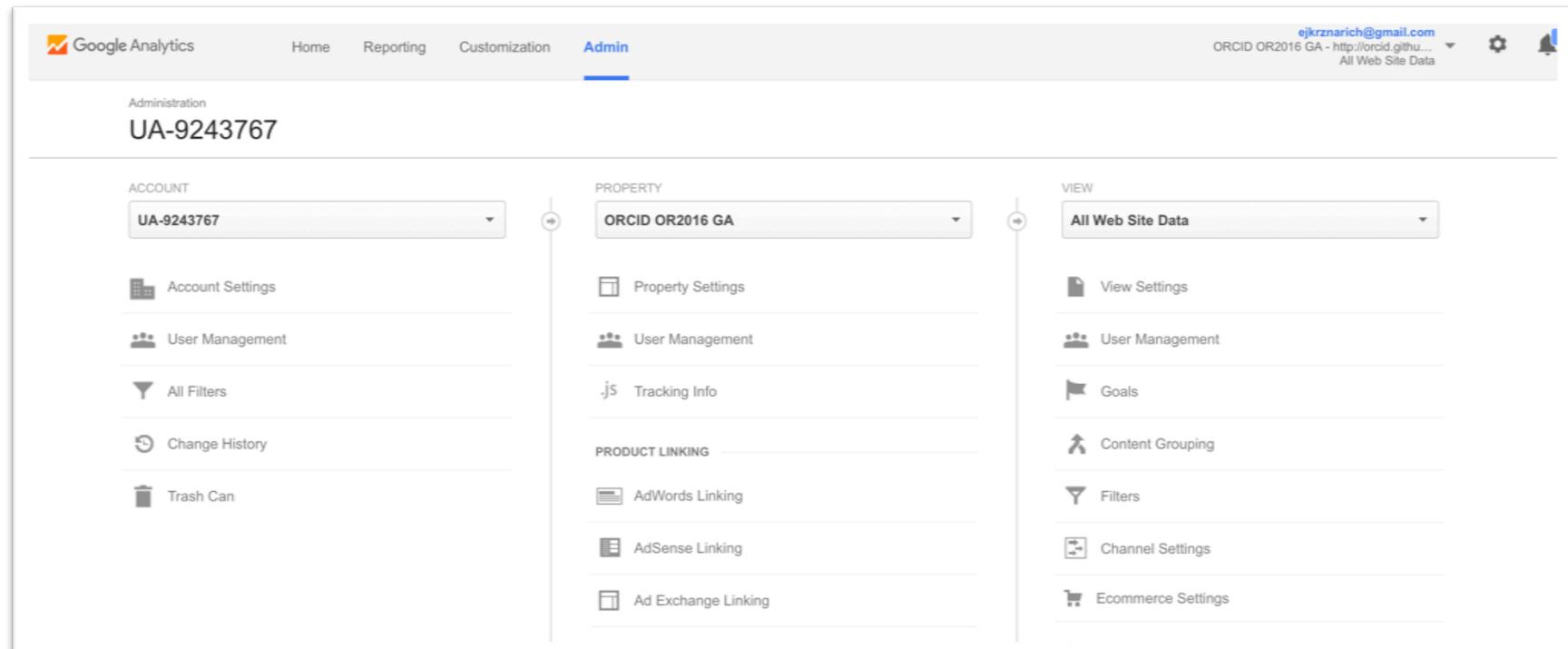
Type: Dataset
Date: 23 October 2015
Authors: Paglione, Laura; Peters, Robert; Wilmers, Catalina; Simpson, Will; Montenegro, Angel; Ramírez Monge, Fran; Tyagi, Shobhit; Krznarich, Elizabeth; Demeranville, Tom; Brown, Josh; Miyairi, Nobuko; Buys, Matthew; Cardoso, Ana; Sethate, Cheryl; Haak, Laurel
Identifier: 10.6084/m9.figshare.1582705.v1

[Download](#) [View](#)

ORCID is the standard for unique researcher identifiers. Learn more. The ORCID Conference (ATC) Call

2. Get a Google Analytics account, create a new project & set up tracking

<https://analytics.google.com>



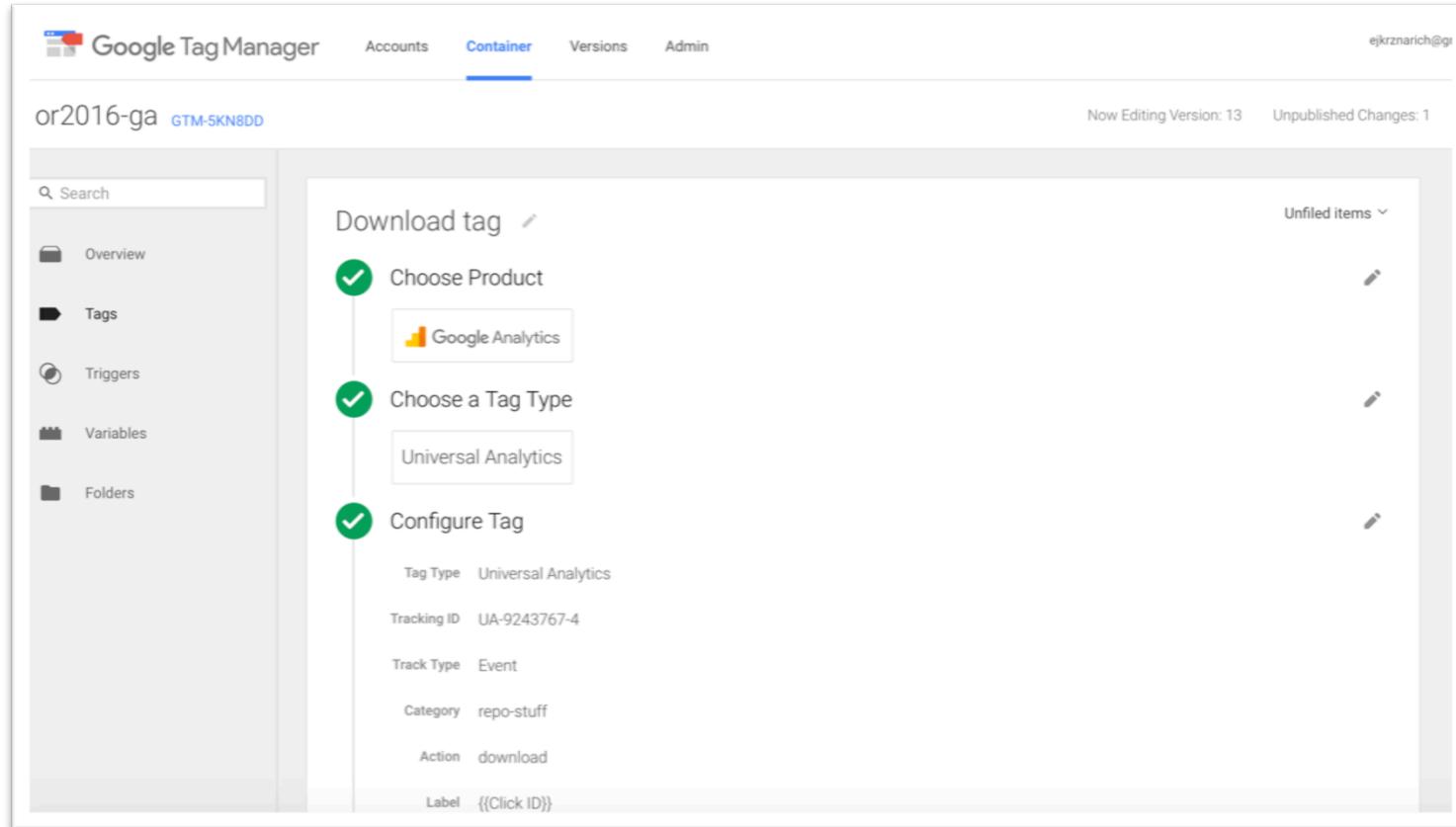
The screenshot shows the Google Analytics Admin interface. At the top, the navigation bar includes 'Google Analytics', 'Home', 'Reporting', 'Customization', and 'Admin' (which is underlined, indicating it is the active tab). The top right corner shows the user's email (ejkrznarich@gmail.com), the account name (ORCID OR2016 GA - http://orcid.github...), and links for 'All Web Site Data', a gear icon for settings, and a bell icon for notifications.

The main content area is titled 'Administration' and shows the account ID 'UA-9243767'. It is divided into three columns: 'ACCOUNT', 'PROPERTY', and 'VIEW'.

- ACCOUNT:** UA-9243767
 - Account Settings
 - User Management
 - All Filters
 - Change History
 - Trash Can
- PROPERTY:** ORCID OR2016 GA
 - Property Settings
 - User Management
 - .js Tracking Info
 - PRODUCT LINKING
 - AdWords Linking
 - AdSense Linking
 - Ad Exchange Linking
- VIEW:** All Web Site Data
 - View Settings
 - User Management
 - Goals
 - Content Grouping
 - Filters
 - Channel Settings
 - Ecommerce Settings

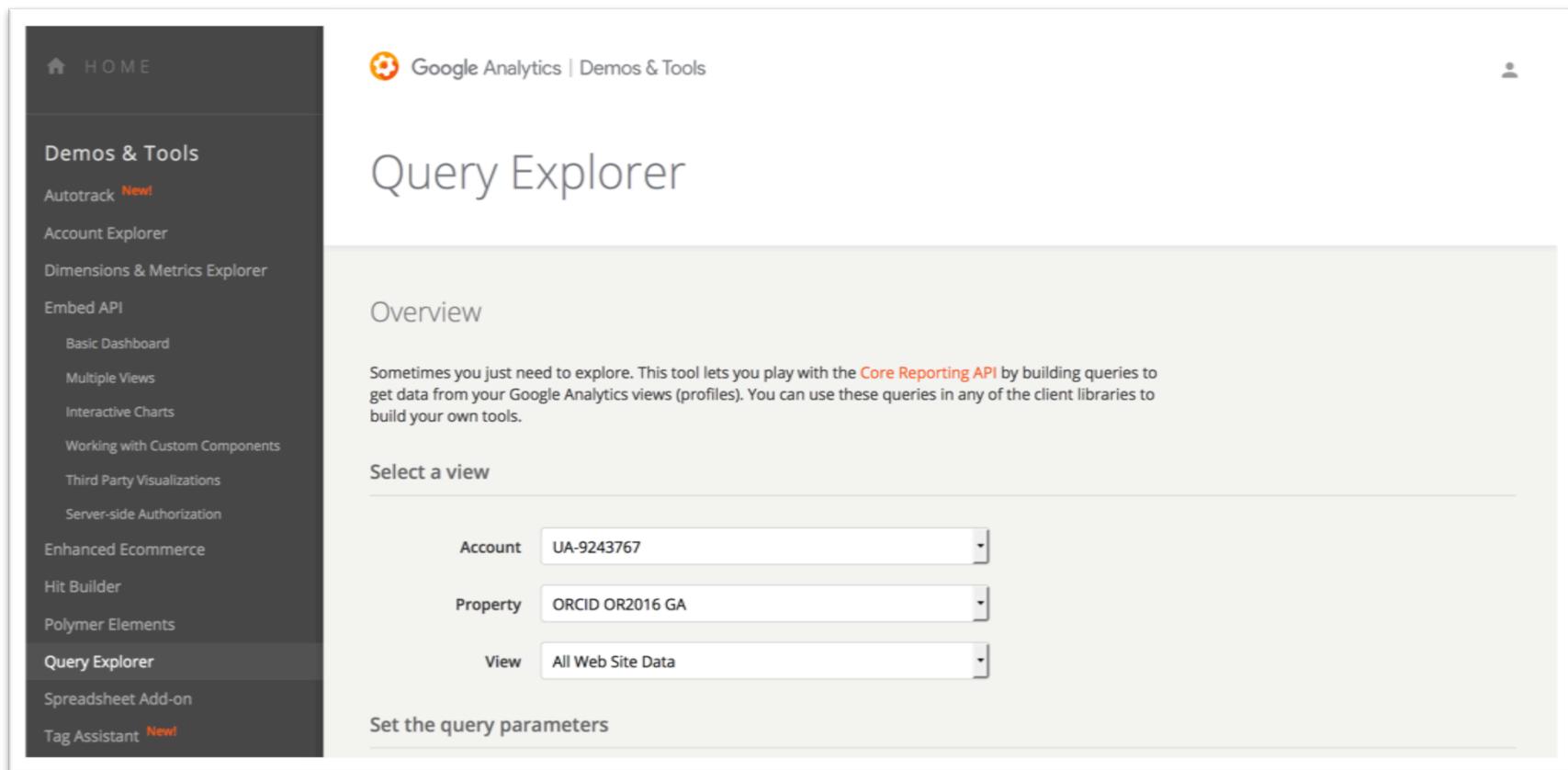
Google Tag Manager = easy customized tracking

<https://tagmanager.google.com>



3. Now for some fun with queries!

<https://ga-dev-tools.appspot.com/query-explorer/>



The screenshot shows the Google Analytics Query Explorer interface. On the left, a dark sidebar menu lists various tools: HOME, Demos & Tools (Autotrack, Account Explorer, Dimensions & Metrics Explorer, Embed API, Basic Dashboard, Multiple Views, Interactive Charts, Working with Custom Components, Third Party Visualizations, Server-side Authorization, Enhanced Ecommerce, Hit Builder, Polymer Elements, Query Explorer, Spreadsheet Add-on, Tag Assistant), and Demos & Tools (New!). The 'Query Explorer' item is highlighted. The main content area has a header 'Query Explorer' and 'Google Analytics | Demos & Tools'. Below the header is a 'Overview' section with text about using the Core Reporting API to build queries. A 'Select a view' section contains dropdown menus for 'Account' (UA-9243767), 'Property' (ORCID OR2016 GA), and 'View' (All Web Site Data). At the bottom, there is a 'Set the query parameters' section.

Dimension & metrics & filters, oh my!

- **Dimensions:** How to break down the data (city, device)
- **Metrics:** What you're counting (clicks, views, etc)
- **Filters:** Limit data by specific criteria

Analytics API Reference:

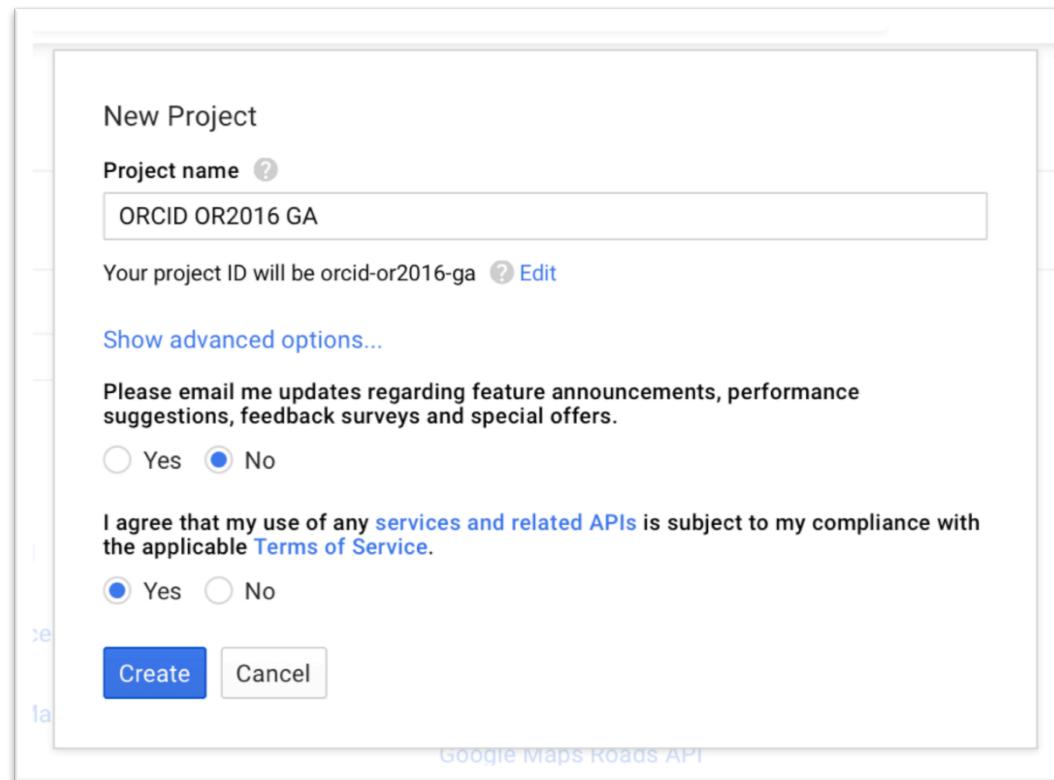
<https://developers.google.com/analytics/devguides/reporting/core/v3/reference>

Dimensions & Metrics Reference:

<https://developers.google.com/analytics/devguides/reporting/core/dimsmets>

4. Create a new Google Developer project

<https://console.developers.google.com/project>



The screenshot shows the 'New Project' dialog box. The 'Project name' field contains 'ORCID OR2016 GA'. The 'Your project ID will be' field shows 'orcid-or2016-ga'. A link to 'Edit' the project ID is available. Below the project ID, there is a link to 'Show advanced options...'. A checkbox for receiving updates is checked, and the 'Yes' radio button is selected. A statement about agreeing to the 'Terms of Service' is present, with the 'Yes' radio button selected. At the bottom, there are 'Create' and 'Cancel' buttons, and a note about the 'Google Maps Roads API'.

New Project

Project name [?](#)

ORCID OR2016 GA

Your project ID will be orcid-or2016-ga [?](#) [Edit](#)

[Show advanced options...](#)

Please email me updates regarding feature announcements, performance suggestions, feedback surveys and special offers.

Yes No

I agree that my use of any [services and related APIs](#) is subject to my compliance with the applicable [Terms of Service](#).

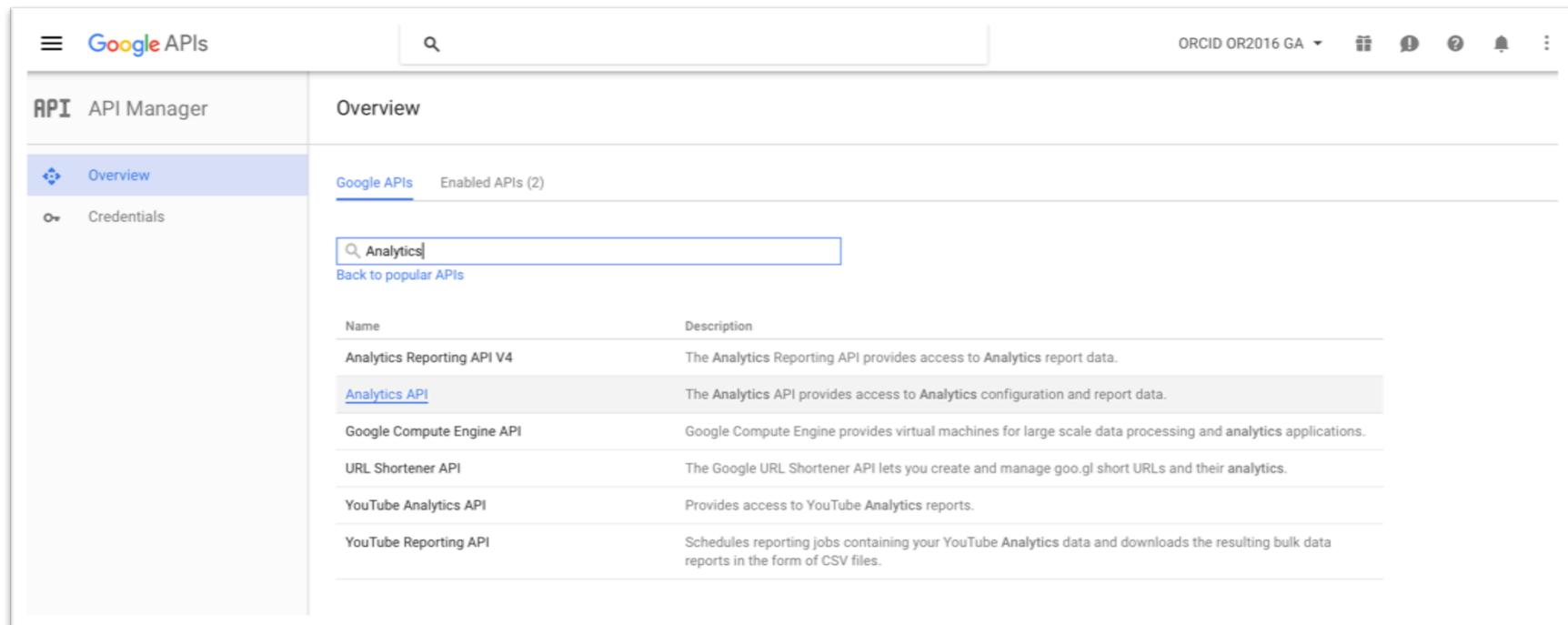
Yes No

[Create](#) [Cancel](#)

Google Maps Roads API

5. Enable APIs (Analytics & Drive)

<https://console.developers.google.com/apis/library>

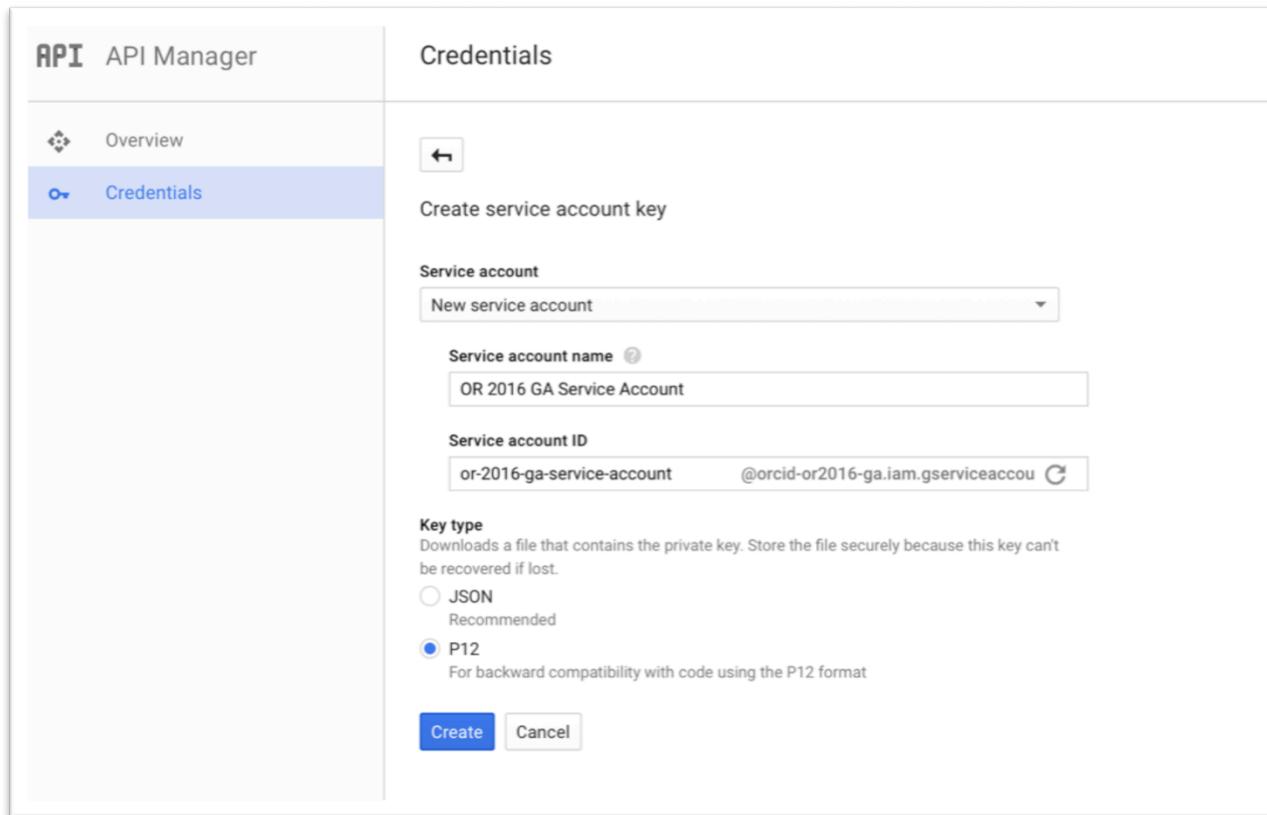


The screenshot shows the Google API Manager interface. The left sidebar has 'API Manager' selected under 'API'. The main area is titled 'Overview' and shows 'Google APIs' selected. A search bar at the top has 'Analytics' typed into it. Below the search bar, there are two tabs: 'Google APIs' (selected) and 'Enabled APIs (2)'. A table lists several Google APIs with their names and descriptions. The table has two columns: 'Name' and 'Description'.

Name	Description
Analytics Reporting API V4	The Analytics Reporting API provides access to Analytics report data.
Analytics API	The Analytics API provides access to Analytics configuration and report data.
Google Compute Engine API	Google Compute Engine provides virtual machines for large scale data processing and analytics applications.
URL Shortener API	The Google URL Shortener API lets you create and manage goo.gl short URLs and their analytics.
YouTube Analytics API	Provides access to YouTube Analytics reports.
YouTube Reporting API	Schedules reporting jobs containing your YouTube Analytics data and downloads the resulting bulk data reports in the form of CSV files.

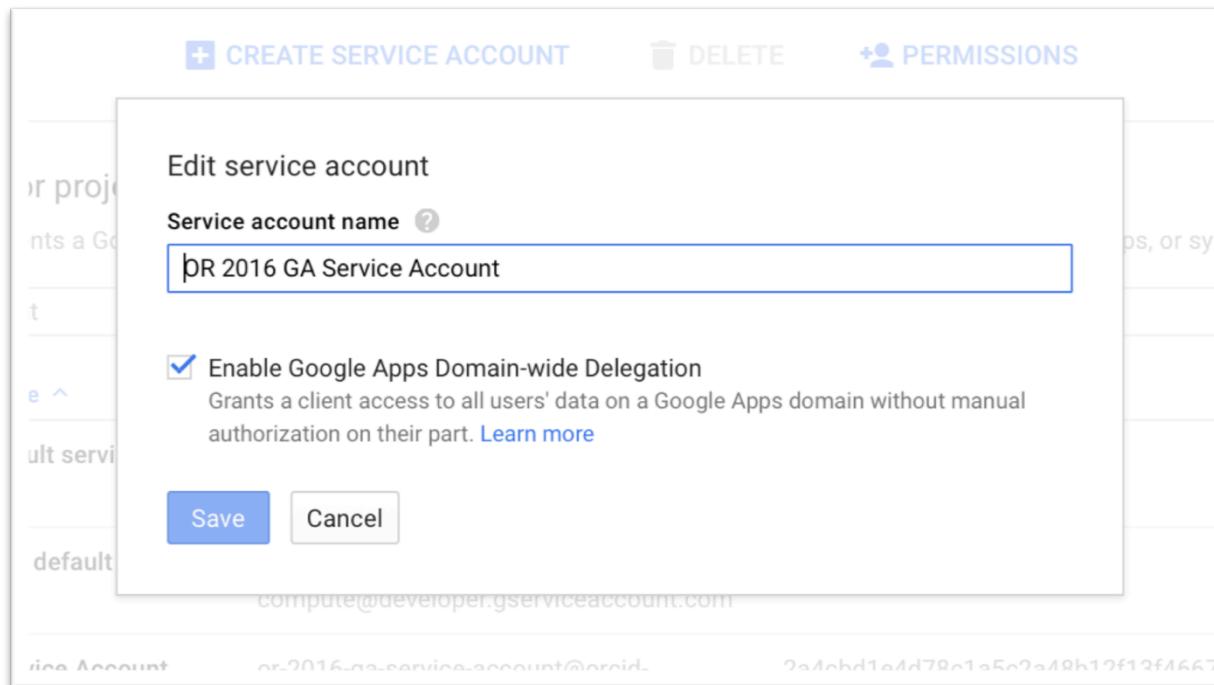
6. Create Service Account & download P12 key

<https://console.developers.google.com/apis/credentials>



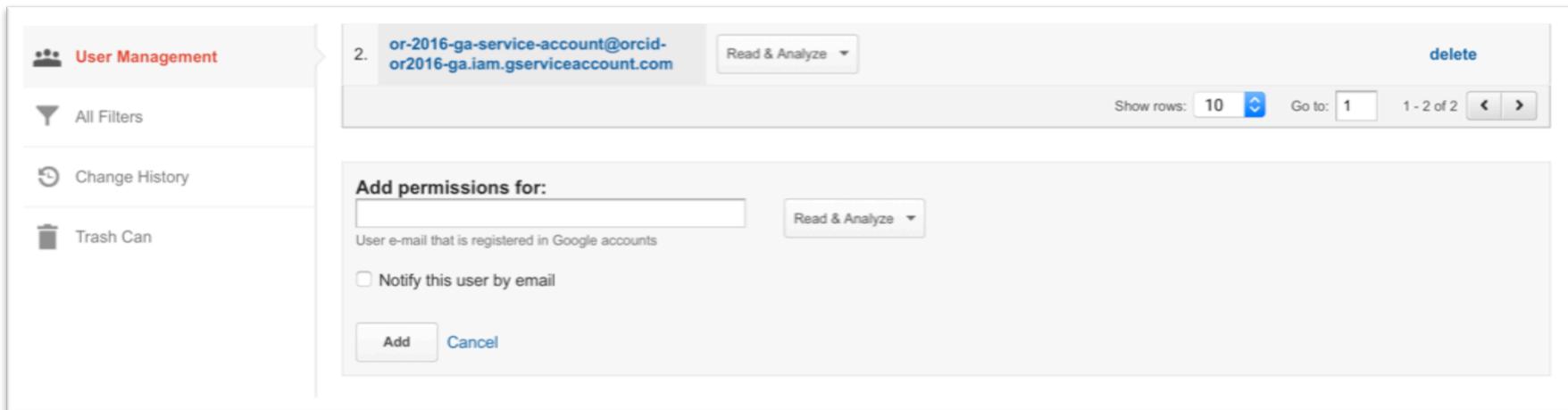
7. Enable domain-wide access

<https://console.developers.google.com/iam-admin/serviceaccounts>



8. Add Service Account to Analytics

<https://analytics.google.com/analytics/web/#management/Settings> > User Management

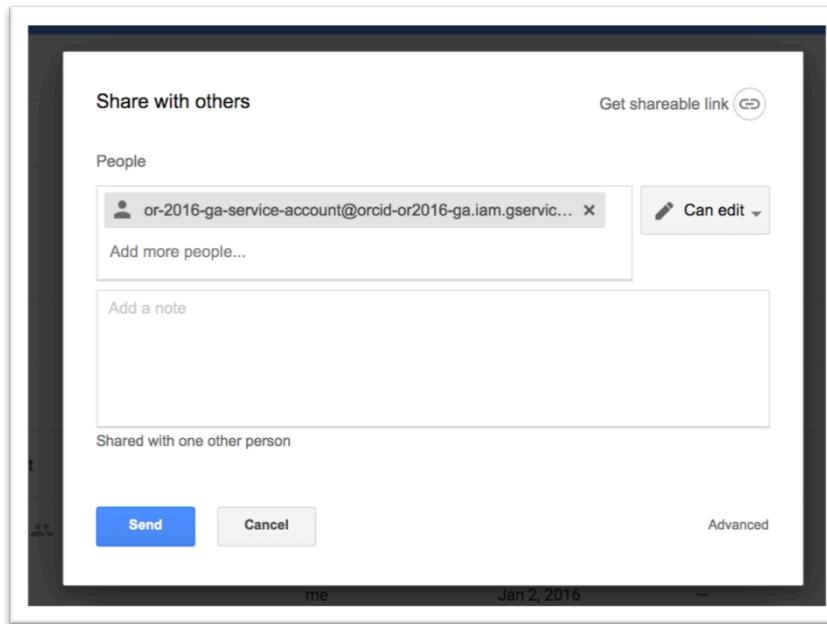


The screenshot shows the Google Analytics User Management interface. On the left, a sidebar has 'User Management' selected. The main area shows a list of users with one entry: 'or-2016-ga-service-account@orcid-or2016-ga.iam.gserviceaccount.com' with a 'Read & Analyze' permission level. Below this, a modal dialog is open titled 'Add permissions for:' with the email 'or-2016-ga-service-account@orcid-or2016-ga.iam.gserviceaccount.com' entered. The dialog includes a 'Read & Analyze' permission level, a 'User e-mail that is registered in Google accounts' note, an unchecked 'Notify this user by email' checkbox, and 'Add' and 'Cancel' buttons.

Guess what?! If you're using the DSpace Google Analytics module, you've already done steps 4-8, and you can use the same service account credentials for your custom applications!

9. Create a new Drive folder & share it with Service Account

<https://drive.google.com/drive/my-drive>





Whew, that was tedious

Probably time for a beer.

Then, on to the code!

Our tasks

1. Authenticate to Analytics, Drive & Sheets APIs
2. Get Analytics data
3. Upload Analytics data to Drive
4. Add data from other sources using Sheets API

API Authentication (Aaargh!!!)

Oauth2 SignedJwtAssertionCredentials

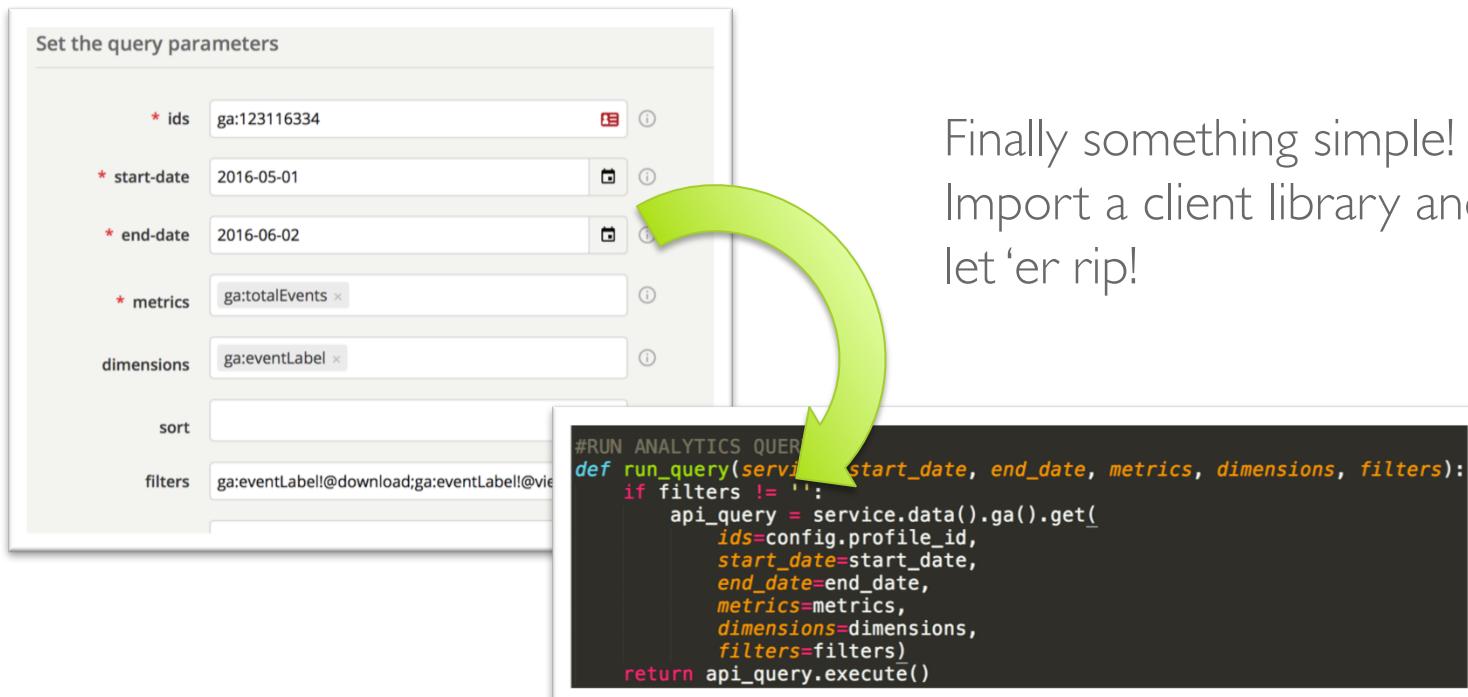
You could read this:

<https://developers.google.com/identity/protocols/OAuth2ServiceAccount#authorizingrequests>

Or just borrow someone else's code.

Get Analytics Data

<https://developers.google.com/analytics/devguides/reporting/core/v3/coreDevguide>



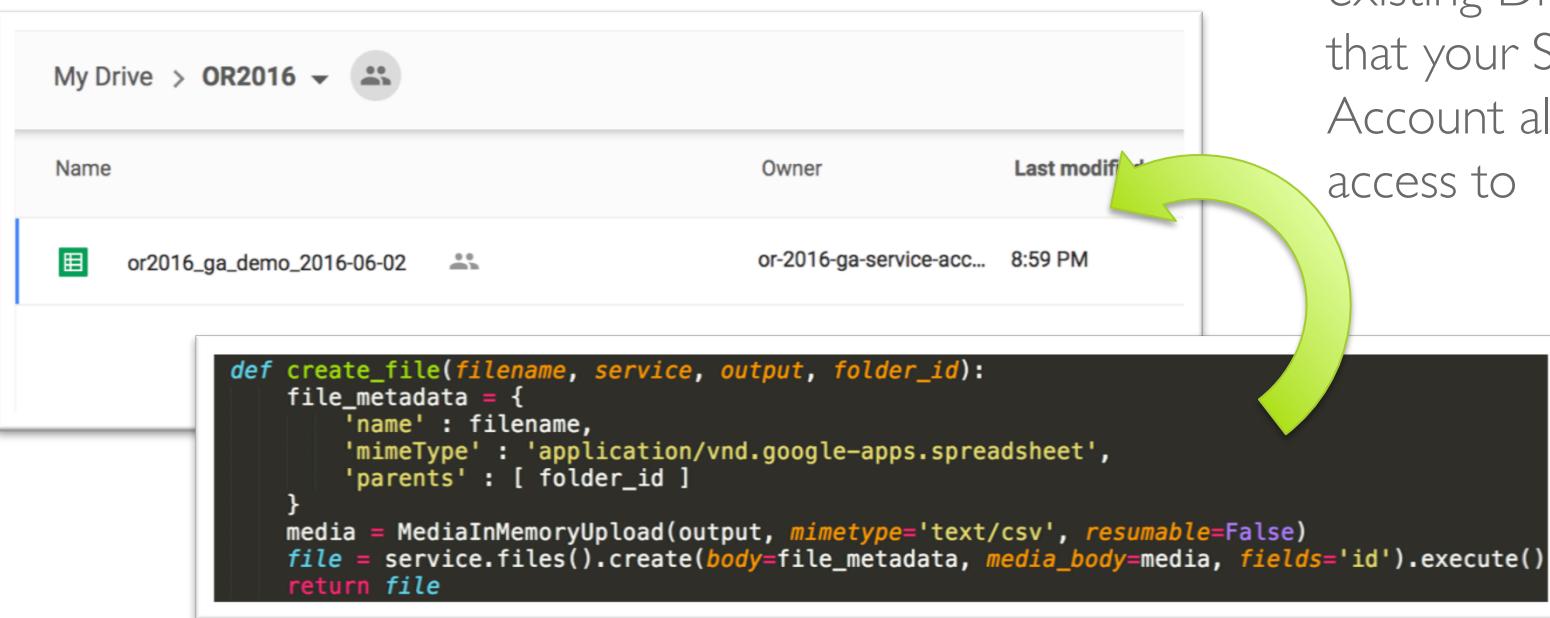
The image shows a comparison between a user interface for querying Google Analytics data and the underlying API code. On the left, a screenshot of the 'Set the query parameters' interface is displayed. It includes fields for 'ids' (set to 'ga:123116334'), 'start-date' (set to '2016-05-01'), 'end-date' (set to '2016-06-02'), 'metrics' (set to 'ga:totalEvents'), 'dimensions' (set to 'ga:eventLabel'), and 'filters' (set to 'ga:eventLabel@download;ga:eventLabel@view'). On the right, a Python code snippet is shown, with a green arrow pointing from the interface to the code. The code defines a function 'run_query' that takes parameters for service, start_date, end_date, metrics, dimensions, and filters, and returns an API query object.

```
#RUN ANALYTICS QUERY
def run_query(service, start_date, end_date, metrics, dimensions, filters):
    if filters != '':
        api_query = service.data().ga().get(
            ids=config.profile_id,
            start_date=start_date,
            end_date=end_date,
            metrics=metrics,
            dimensions=dimensions,
            filters=filters)
    return api_query.execute()
```

Finally something simple!
Import a client library and
let 'er rip!

Upload data to Drive

<https://developers.google.com/drive/v3/reference/files/create>



My Drive > OR2016

Name	Owner	Last modified
or2016_ga_demo_2016-06-02	or-2016-ga-service-acc...	8:59 PM

```
def create_file(filename, service, output, folder_id):
    file_metadata = {
        'name' : filename,
        'mimeType' : 'application/vnd.google-apps.spreadsheet',
        'parents' : [ folder_id ]
    }
    media = MediaInMemoryUpload(output, mimetype='text/csv', resumable=False)
    file = service.files().create(body=file_metadata, media_body=media, fields='id').execute()
    return file
```

Tip: Start with an existing Drive folder that your Service Account already has access to

Get some other data

In the example, we loop through a list of DOIs, checking to see which ones are linked to ORCID iDs

```
curl -H "Content-Type: application/orcid+xml" -H "Accept: application/orcid+json" "https://pub.orcid.org/v1.2/search/orcid-bio/?q=digital-object-ids%2210.1087%2F20120404%22"
```

Hey! You can use the ORCID Public API to do this...it's just an HTTP request away!

Edit the new sheet to add your data

<https://developers.google.com/sheets>

<http://gspread.readthedocs.io>

- Python gspread library makes life easy
- Add/edit rows, columns, cells, sheets, etc
- Find content in a sheet
- Sheet API v3 was frightening – v4 might be able to do w/out Gspread

Edit the new sheet to add your data

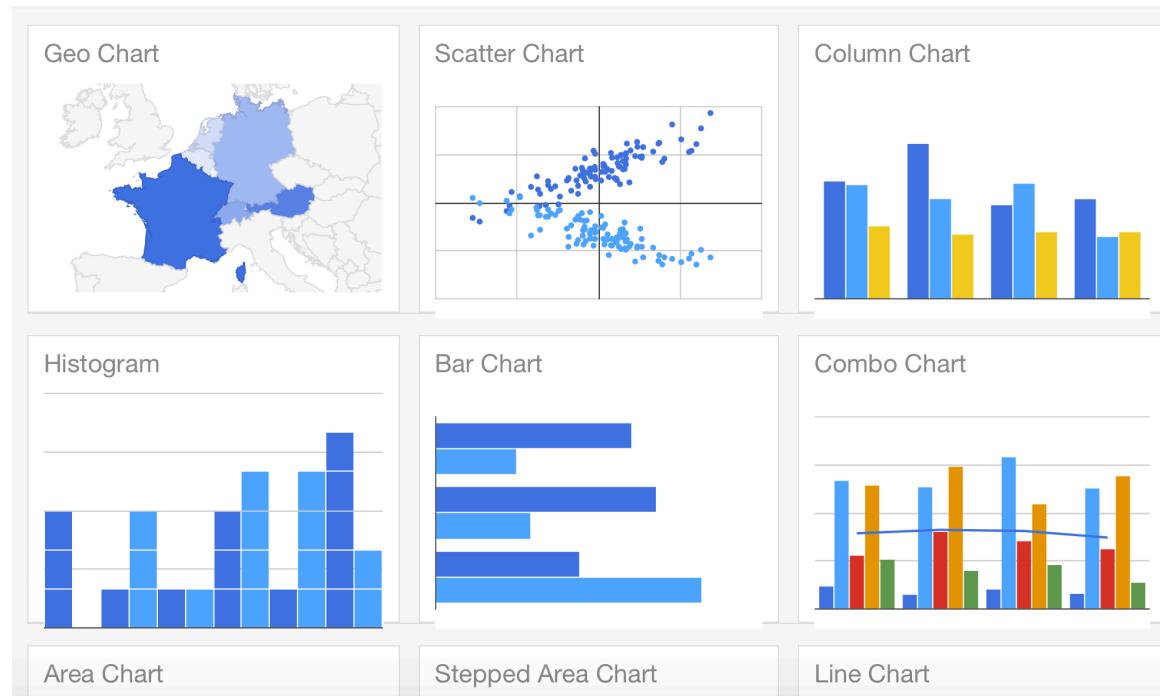
1. Get the Drive file id
2. Open the file
3. Figure out which cells to edit & send the update request

My Fancy Analytics Report		
	A	B
1	My Fancy Analytics Report!	
2	Generated on 2016-06-02	
3	Data for 2016-05-01 to 2016-06-02	
4		
5	Aggregate Data	
6	Items in repository	7
7	Items downloaded at least once	10
8	Items linked to at least 1 ORCID iD	3
9		
10	Item Data	
11	DOI	Downloads
12	10.1087/20120404	5
13	10.6084/m9.figshare.1582705.v1	4
14	10.6084/m9.figshare.3172084	4
15	Download label test	11
16	download-10.1087/20120404	1
17	download-10.6084/m9.figshare.1582705.v1	1
		2
		1
	05.v1	2
	34	1

```
def edit_spreadsheet(sheets_client, drive_file, orcid_data):  
    drive_file_fileId = drive_file.get('id')  
    drive_file_worksheet = sheets_client.open_by_key(drive_file_fileId).sheet1  
    orcid_record_col = drive_file_worksheet.find('ORCID Records with this DOI').col  
    doi_orcid_count = 0  
    for doi in orcid_data:  
        try:  
            doi_match_row = drive_file_worksheet.find(doi[0]).row  
            drive_file_worksheet.update_cell(doi_match_row, orcid_record_col, doi[1])  
            doi_orcid_count += 1  
        except:  
            pass  
  
    total_linked_orcid_row = drive_file_worksheet.find('Items linked to at least 1 ORCID iD').row  
    drive_file_worksheet.update_cell(total_linked_orcid_row, 2, doi_orcid_count)
```

Now, take that data and do cool stuff!
(like charts with Google Charts API)

<https://developers.google.com/chart>



Hot Tips!

- Tokens expire in 1hr (no refresh token...just get a new one)
- Analytics API is twitchy – expect sporadic errors when running lots of queries
- Sheets API is sloooooow – try to combine as many actions as possible into 1 request

Get the code

Demo site

<http://orcid.github.io/or2016-ga>

Code

<https://github.com/lizkrznarich/OR2016>

THANK YOU!

Questions? e.krznarich@orcid.org

<https://github.com/lizkrznarich>

<http://orcid.org/0000-0001-6622-4910>