MIP Data Repository
New Client Onboarding

New Client Onboarding Documentation:

Media Impact Project - Data Repository:

MIP offers a simple and scalable framework to help media organizations understand their current audiences and target potential ones. We do this through our data repository. It is for tracking people across their website, e-mail newsletter, event registration, donor databases and other audience data sources.

While this document will focus on their website, providing additional data sources can help media organizations to understand their current audience.

Assessment:

The client provides Google Analytics access to USC Annenberg Analytics account and uscmediaimpact@gmail.com .   MIP will do the necessary customization / improvements for MIP standard metrics.

As part of this assessment, the client site will need to be evaluated for the following items and GTM customized accordingly:

- Membership
- Email Subscriber tracking
- Donations
- Customized Scroll Depth level markers for articles
- Social Actions on an article
- Article vs standard page URLs
- Any other data sources (newsletters, etc)

Internal Setup

After the assessment, MIP will follow the steps below to setup the client in Google Cloud and GA / GTM.

Step 1: Google Cloud Platform Client Setup:

1. **Create Project:** In Cloud Console, create a project (from the drop-down project list at the top right) and give it a sensible name. Make sure to select a billing account. Note the short name of the project; you will need it later.
2. **Enable APIs:** When you create a project, most of the needed APIs are enabled automatically. However, Compute Engine needs to be enabled for Dataflow. Go to API Manager in the Cloud Console left nav and enable Compute Engine. BigQuery, Cloud Storage, and Datastore should already be enabled by default.
3. **Create Storage bucket:** Create a Cloud Storage bucket for temporary file storage during processing operations. Go to Cloud Storage in the Cloud Console left nav and create a new bucket. Note the name of the bucket; you will need it later.
4. **Create BigQuery dataset:** Create the dataset in which the access logs and processed data will be stored. Go to BigQuery in the Cloud Console left nav. Make sure the correct project is selected in BQ and create a dataset. Typically we use the view ID of the corresponding view in GA as the dataset name to match what GAP does. Note the name of the dataset; you will need it later.
    - **Copy access_logs_ table:** In the dataset you just created, you have to copy the access_logs_ template table for the streaming logs. From the luna-crunch-test project, copy this table into the dataset.
5. **Upload default module:** The collector is the default App Engine module.
    a. **Edit settings.yaml:** Put in the project short name for the job, change the time zone to match the time zone of the site, put in the BQ dataset name and the Cloud Storage bucket created above. Edit the custom dimension settings to control whether they are processed only at the hit level or bumped up to the session data.
    b. Upload the code to App Engine:
    /path/to/appcfg.sh -A project-name update /path/to/default/
6. **Update the mip-config.yaml**

    Modify and re-upload the mip-config.yaml within the following folder to allow for the new client. Copy the existing format for the previous clients. Contact Lunametrics regarding any errors.

    https://console.cloud.google.com/storage/browser/mip-dataflow-config/?project=mip-dashboard

7. **Upload dataflow-cron module:** This is the module that handles the nightly processing.

   a. **Edit cron.xml:** If necessary, change the time the nightly processing runs. (We usually run all NA sites at 2:00am Pacific time, but if you have EMEA or APAC sites you will want to change this.)
   b. **Upload the code to App Engine:**

```
/path/to/appcfg.sh -A project-name -M dataflow-cron update /path/to/dataflow-cron/
```

Step 2: GA & GTM Client Setup:

1. **Create property/view:** If it doesn't already exist, create a property and view for the site in GA.
2. **Create a container:** If it doesn't already exist, create a container for the site in GTM.
3. **Set up GTM:** Create tags, triggers, and variables as normal but with the following changes:
   a. **Task tag:** Include the task tag (Custom HTML) that includes the Javascript that sets the task to duplicate hits to GA to your project.
   b. **Collector URL variable:** Set the variable for the URL of your project's collector: project-name.appspot.com/log
   c. **Tracker name:** For all GA tags, set the tracker name (t0 is the default value, but change the variable and the custom HTML tag if you want to change the name).
   d. **Pageview triggers:** For any pageview triggers for GA, rather than firing on gtm.js, make sure they fire on the lm.ready event called in the task tag.

   There's a container in the internal recipe library that includes the task tag, a GA pageview tag, and the necessary variables. Also, see the Luna container for a working example.

      - Each client will need the customization of the GTM level for the MIP custom metrics and html tags. This needs to be done prior to the embed code going live with the client. eg:
         - Membership
         - Email Subscriber tracking
         - Donations
         - Customized Scroll Depth level markers for articles
         - Social Actions on an article
         - Article vs standard page URLs
         - Any other data sources (newsletters, etc)

Step 3: Dashboard setup:

1. Add the new client from within the Client Management section.
2. Create the new folders within the Google App Engine App structure (resources > views > data) to reflect the new client's data section with any customizations for ui that are required.
3. Update the Data Controller where appropriate (app > Http > Controllers > DataController.php)
4. Check the database to make sure the new client tables have been automatically created successfully.
5. Set the the new client 'ready' flag to 'true' within the MYSQL database in the clients table.
6. Add New users for the client in the 'Settings' section of the Dashboard.

Onboarding

Step 1: Client adds Google Tag Manager embed code

Google Tag Manager (GTM) allows us to provide a customized experience and MIP metrics. We will provide the media organization with an embed code for their site. This GTM embed code will need to be added to the header of all pages on the site.

This will send data from Google Tag Manager to both Google Analytics and our data repository in an asynchronous manner. This means that the performance of the website not be affected by this additional embed code.

The GTM embed code allows us to make improvements and updates to the audience data collection on the fly. While we do test and debug best practices before all changes, for major GTM updates we will coordinate to make sure the media organization's technical team is aware prior the change.

Step 2: Gather additional data sources

The media organization will provide access to additional data sources including website usernames, e-mail newsletter, event registration, donor databases and other audience data sources. This may be simple CSV or Excel documents / exports from various third party sources, and/or shared access to those sources.

We will add these additional data sources to our data repository for the client accessible data repository dashboard.

Data Validation

Step 1. Data Processing with GTM & Query Updates

After the GTM embed code is placed on the client site, we will validate the initial data and make the necessary GTM changes to improve data quality for future analysis. The data should be reviewed on a daily basis. Both GTM changes and BQ query modifications many need to be made in order to make the dashboard function as intended.

Step 2. Final Review with Analyst

After two weeks of validated and quality data and no data exceptions, have a Data Analyst do a final review of the data with the data analyst in preparation for giving the client access to the dashboard.

Step 3. Walkthrough with Client

After the dashboard has 2-3 weeks of valid and approved data, schedule a meeting to walk the client through the dashboard and provide the email/password access for their admin and other dashboard users.