

Developer Quickstart

Suggest Edits

Introduction

Welcome to the Lytics developer tier! This guide will walk you through the steps to get started with Lytics and leverage

Before You Begin

Before diving into the setup process, make sure you have the following:

Site Access / Management Permission: To install Lytics, you need permission to install JavaScript either via a tag manager or install Lytics via a Drupal module.

Active Lytics Account: Verify that you can access an active Lytics account. If you don't have one yet, you can claim your own.

Lytics Dev Tools Chrome Extension: Install our developer tools Chrome extension to streamline the development and testing process.

Getting Started Checklist

Getting started with Lytics is quick and easy! In just a few minutes, you'll be able to set up Lytics and start personalizing your website. Follow these steps to ensure a positive experience for you and your customers:

1. Install the Lytics tag on your site.
2. Ensuring site content and Lytics are syncing.
3. Create your first personalized message.

Digging Deeper

After completing the initial checklist outlined above, it's time to explore further avenues for enhancing and utilizing your Lytics personalization guides into two core focuses:

Building Profiles

Here, we'll gain a comprehensive understanding of all available out-of-the-box attributes. Discover how to tag your site and create user profiles. This section is divided into:

Default Attributes & Segments:

Profile Attributes

Audience Segments

Content Collections

Site Activity & Conversion Tagging:

Capturing Website Activity (coming soon)

Capture Website Conversion Activity (coming soon)

Using Profiles

Here, we'll explore leveraging out-of-the-box personalization SDKs and APIs to deliver optimal user experiences. Discover how to create personalized experiences that resonate with your audience. This section covers:

Guides & Inspiration

Surface a lead capture form only to unknown visitors.

Surface content recommendations based on interests.

Surface a promotional message to high-momentum visitors. (coming soon)

Sync profiles & audiences to GA4 or meta. (coming soon)

Personalize your site based on behaviors and stored attributes. (coming soon)

SDK Documentation

Web

JavaScript SDK

Personalization SDK

Mobile

iOS SDK

Android SDK

React Native SDK

Updated 5 months ago

What's Next

1. Account Setup

1. Install the Lytics Tag

Suggest Edits

Installing the Tag

In the Web SDK, Lytics provides a tag that can be placed on your site to collect behavioral data and surface the materialization instructions are available from within the app at Data Pipeline > SDK > Web SDK.

If your site is a Drupal or Wordpress site, or you use Google Tag Manager we recommend using one of these turnkeys:

Drupal

WordPress

Google Tag Manager

Testing the Lytics Tag Installation

Once the tag has been installed, validate a successful installation via one of the three following methods:

Lytics Dev Tools Chrome Extension (Recommended)

In-app Verification Assistant

Manually

Testing the Current Visitor's Profile via JavaScript SDK

The final step to verify installation is ensuring you can access your visitor profile. This profile is built and delivered in

Lytics Dev Tools Chrome Extension (Recommended)

Manually

Updated 4 months ago

What's Next

2. Content Setup

2. Content Setup

Suggest Edits

Lytics' Interest Engines effectively analyze your content with minimal effort required. However, to guarantee that the results of a quick test is advisable, as poor metadata can sometimes lead to less-than-optimal outcomes. The most reliable method is to classify a selection of your URLs manually.

Use the left-hand navigation to select Content > Classification.

Click on the section labeled Classify, then paste a URL from your website and press Classify.

After a moment or two, a set of preliminary classification details will come back. Most notably, ensure the Title, Primary Category, and Meta Tags are populated. For more information, please refer to our full documentation for configuring content and meta tags.

Updated 4 months ago

What's Next

3. Surface Personalized Message

3. Surface Personalized Message

Suggest Edits

The Lytics Personalization Engine profiles real-time access to a comprehensive visitor profile. Before we jump into the next steps, let's create our first experience.

Surface a Simple Message

Lytics comes with our Personalization SDK called Pathfora. Pathfora allows you to easily surface simple lead capture forms. If a form for Pathfora is available, but initially, let's surface a welcome message to our anonymous visitor audience.

javascript

```
// jstag profiles a helper function to ensure that the Pathfora library
```

```
// has been loaded before triggering the experience.
```

```
jstag.on('pathfora.publish.done', function(topic, event){
```

```
  // here we initialize a new Pathfora "Message" experience
```

```
  var module = new pathfora.Message({
```

```
    id: 'sample-message-campaign', // this value will be collected along side all interactions and used in reporting
```

```
    layout: 'slideout', // for layout we'll use a small slide out
```

```
    position: 'bottom-left', // the model will enter and sit at the bottom left
```

```
    theme: 'dark', // css can be customized to brand but we'll use the default dark theme
```

```
    headline: 'Hello world!', // this will be the headline of our message
```

```
    msg: 'Congratulations on setting up your first targetted campaign using the Lytics Personalization Engine!', // the body of the message
```

```
});

var modules = {
  target: [{
    segment: "anonymous_profiles", // target only visitors with the anonymous_profile attribute
    widgets: [module]
  }]
};

pathfora.initializeWidgets(modules); // initialize the campaign
});
Alter the Pathfora configuration to your liking.
```

Install the Pathfora configuration onto your site via your preferred tag management method.

Refresh the page and be greeted with your new welcome message targeted at anonymous visitors!

Lytics and Pathfora provide a great deal of flexibility. If you are ready to dive deeper, please explore some of our other

Surface a Promotional Message to High Momentum Visitors (coming soon)

Surface a Lead Capture form Only to Unknown Visitors

Surface Content Recommendations Based on Interests (coming soon)

Sync Profiles & Audiences to GA4 or Meta (coming soon)

Personalize Your Site Based on Behaviors and Stored Attributes (coming soon)

Updated 5 months ago

Building Profiles

Suggest Edits

Welcome to the world of building profiles with Lytics! Here, you can unlock the full potential of understanding your audience. x:

Default Profile Attributes: Lytics offers a robust set of pre-packaged user attributes, including automatically generated machine learning algorithms, these attributes enable you to gain deeper insights into your audience.

Default Segments: Explore predefined user segments that categorize your audience based on behavioral and demographic data. These segments offer quick insights for tailored marketing strategies.

Ready to dive in? Click on each section to explore the details and enhance your profiling efforts effortlessly. Looking for all the available attributes.

Updated 10 months ago

Default Attributes

Suggest Edits

Lytics offers a wide range of pre-packaged user attributes, including automatically generated and customizable ones. Machine learning algorithms provide insights and scores, allowing users to gain a deeper understanding of their audience. These attributes are used in various ways to enhance your profiling efforts.

-

If you haven't already reviewed our documentation on collecting events via our JavaScript SDK, we highly recommend reviewing it first. Data collection works at a high level before delving into the specifics of what can be collected.

Available Attributes

The following attributes are all available out of the box with no customization necessary in all Lytics pricing tiers. Some are collected directly but are computed based on various factors, including other non-computed attributes.

,

For a more comprehensive example of how any of the following attributes can be collected and used for your visitor

w.

Identifiers

Default attributes that are used to stitch profiles together. For instance, if you pass an email along with the `_uid`, all email addresses will be merged into a single comprehensive profile.

Name	Slug	Description	Computed	Example
Lytics ID	<code>_id</code>	A unique ID that represents the materialized profile in Lytics.	Yes	4fab5b3-b199-58f2-a68b-4b266b363dd1
Current Lytics Cookie	<code>_uid</code>	The current cookie id for the user.	No	50b772f5-a0be-42f2-8828-84b8db5d5a23
All Lytics Cookies	<code>_uids</code>	All cookies that are associated with the user.	No	["50b772f5-a0be-42f2-8828-84b8db5d5a23", "50b772f5-a0be-42f2-8828-84b8db5d5a23"]
Email	<code>email</code>	The email address of the user.	No	example@lytics.com
Unique User ID	<code>uuid</code>	A UUID for the user.	No	4fab5b3-b199-58f2-a68b-4b266b363dd1

Details

Details encompass all default attributes related to user demographics and general information, including name, phone number, and email address, as well as attributes specifically tied to interactions or behaviors.

Name	Slug	Description	Computed	Example
Name	<code>name</code>	The full name of the user.	No	John Doe
First Name	<code>first_name</code>	The first name of the user.	No	John
Last Name	<code>last_name</code>	The last name of the user.	No	Doe
Title	<code>title</code>	The title of the user.	No	President
Phone	<code>phone</code>	The phone number of the user.	No	555-555-5555
Cell	<code>cell</code>	The cell phone number of the user.	No	555-555-5555
Origin	<code>origin</code>	The origin of the user.	No	loyalty_2022
Language	<code>language</code>	The language of the user.	No	en-us
Age	<code>age</code>	The age of the user.	No	25
Companies	<code>companies</code>	The companies the user is associated with.	No	["Lytics", "Pantheon"]
Gender	<code>gender</code>	The gender of the user.	No	M
Status	<code>status</code>	The status of the user.	No	active
User Attributes	<code>user_attributes</code>	A map of custom attributes associated with the user.	No	{"role": "member", "bonus": 100}
Timezone	<code>timezone</code>	The timezone of the user.	No	-7
City	<code>city</code>	The city of the user.	No	Denver
Country	<code>country</code>	The country of the user.	No	US
State	<code>state</code>	The state of the user.	No	CO

Meta

Meta encompasses all system-level information that provides insights into the health and breadth of the profile. This includes information about the profile's source information, and other metadata associated with the profile's management and maintenance. Metadata offers a high-level overview of the profile's history and performance.

Name	Slug	Description	Computed	Example
Created	<code>_created</code>	The date the user was created.	Yes	2023-12-12T21:09:11.625960142Z
Last Scored	<code>_last_scored</code>	The date the user was last scored.	Yes	2024-02-28T02:45:51.377423153Z
Modified	<code>_modified</code>	The date the user was last modified.	Yes	2024-02-28T02:45:51.377423473Z
Number of Aliases	<code>_num_aliases</code>	The number of aliases for the user.	Yes	1
Number of Days	<code>_num_days</code>	The number of days the profile has existed.	Yes	38
Number of Events	<code>_num_events</code>	The number of events the user has been associated with.	Yes	2425
Number of Streams	<code>_num_streams</code>	The number of streams the user has been associated with.	Yes	2
Stream Names	<code>_streamnames</code>	The names of the streams the user has been associated with.	Yes	["default", "ios"]
User is Bot	<code>is_bot</code>	Whether the user has been flagged as a bot or not.	Yes	f

Behavior

Behavioral attributes typically cannot be directly managed but represent a set of insights derived from a user's behavior over time. These attributes are calculated based on changes in behavior or behaviors indicative of high likelihood. For instance, you might want to track a user's engagement with your content in a usual. Behavioral attributes enable targeted and timely interventions tailored to user actions and patterns.

Name	Slug	Description	Computed	Example
Consistency	<code>score_consistency</code>	Score representing how consistent their activity patterns are.	Yes	99
Frequency	<code>score_frequency</code>	A score representing how frequently the user is active.	Yes	63
Intensity	<code>score_intensity</code>	A score representing how intense the user's activity is.	Yes	94
Maturity	<code>score_maturity</code>	A score representing how mature the user's activity is.	Yes	34
Momentum	<code>score_momentum</code>	A score representing how much momentum the user currently has.	Yes	54

Propensity score_propensity A score representing how likely the user is to engage again. Yes 1
Quantity score_quantity A score representing how much activity the user has. Yes 99
Recency score_recency A score representing how recent the user's activity is. Yes 99
Volatility score_volatility A score representing the degree of variability in behavior. Yes 99
Interests

Interests entail understanding the topics a user is interested in based on their interactions, cross-referenced by deep learning models to provide insights for tailored content recommendations and targeted messaging aligned with the user's preferences and engagement patterns.

Name Slug Description Computed Example

Lytics Content lytics_content A map of topic-level interests for the user. Yes {"Baking": 0.26418695138978837}
Intelligence

Attributes classified as intelligence encompass diverse, highly valuable information to facilitate relevant and high-value marketing strategies, cover real-time segment membership, values crucial for split testing and experimentation, and direct correlation to user behavior for dynamic and data-driven decision-making, enhancing the efficacy of personalized marketing strategies.

Name Slug Description Computed Example

Segment Membership _segments The segments the user is associated with. Yes ["all", "anonymous_profiles", "smm"]
Split _split A random value that is evenly distributed across users. Yes 74

Split 2 _split2 A random secondary value that is evenly distributed across users. Yes 58

Needs Message needs_message Stream-specific score that represents the relative distance between now and the next expected event. Yes 0.5

Next Event next_event Stream-specific prediction for the next expected event. Yes {"default": "2024-03-01T03:00:00Z"}

Lookalike Model Predictions segment_prediction Scores from Lytics Lookalike and SegmentML models. Yes {"likelihood": 0.85}

Lookalike Model Percentiles segment_prediction_percentile Percentiles from Lytics Lookalike and SegmentML models. Yes [0.1, 0.5, 0.9]
Activity

Activity encompasses the user's engagement across different channels and campaigns, including clicks and conversions, used for campaign optimization and channel effectiveness assessment.

General

Name Slug Description Computed Example

First Seen event_first_seen The first time the user was seen for a specific event. No {"click": "2023-12-12T21:09:11.625Z"}

Last Seen event_last_seen The last time the user was seen for a specific event. No {"click": "2024-02-28T02:45:49.784Z"}

Channels channels The channels the user has been active on. No ["web", "email"]

Devices devices The devices the user has been active on. No {"desktop": 123}

Hourly hourly The number of events per hour for the user. Yes {"0": 17, "1": 69, "2": 262, "3": 97}

Hour of Week hourofweek The number of events per hour of the week for the user. Yes {"3": 2, "4": 2, "5": 1, "11": 1}

Last Active last_active_ts The last time the user was active. No 2024-02-28T02:45:50.784Z

Last Channel Activities last_channel_activities The last time the user was active on a specific channel. No {"web": "2024-02-28T02:45:50.784Z"}
Web

Name Slug Description Computed Example

Domains domains The domains the user has been active on. No ["umami.lytics.com"]

First Visit Timestamp firstvisit_ts The first time the user visited the site. No 2023-12-12T21:09:11.625Z

Last Visit Timestamp lastvisit_ts The last time the user visited the site. No 2024-02-28T02:45:50.784Z

Pageview Count pageviewct The number of pageviews the user has had. Yes 234

Referring Domain refdomain The referring domain for the user. No ["umami.lytics.com"]

User Agent user_agent The user agent for the user. Yes Chrome

Visit Count visitct The number of visits the user has had. Yes 145

Visit City visit_city The city the user visited from. Yes Denver

Visit Country visit_country The country the user visited from. Yes US

Visit Region visit_region The region the user visited from. Yes CO

Form Data form_data The form data the user has submitted. No {"first_name": "John"}

Forms Submitted forms_submitted The forms the user has submitted. No ["newsletter", "contact"]

UTM Campaign Last utm_campaign_last The last UTM campaign referred from. No holiday

UTM Campaigns utm_campaigns The UTM campaigns the user has interacted with. No ["holiday", "summer"]

UTM Content Last utm_content_last The last UTM content referred from. No recipe-1

UTM Contents utm_contents The UTM contents the user has interacted with. No ["recipe-1", "recipe-2"]

UTM Medium Last utm_medium_last The last UTM medium referred from. No article

UTM Mediums utm_mediums The UTM mediums the user has interacted with. No ["article", "recipe"]

UTM Source Last utm_source_last The last UTM source referred from. No google_ads

UTM Sources utm_sources The UTM sources the user has interacted with. No ["google_ads", "meta_ads"]

UTM Term Last utm_term_last The last UTM term referred from. No example

UTM Terms `utm_terms` The UTM terms the user has interacted with. No ["example"]

Campaign

Name Slug Description Computed Example

Hover `ly_hover` The number of times the user hovered over a specific campaign. No {"content-rec-modal": 5}

Impressions `ly_impressions` The number of times the user saw a specific campaign. No {"content-rec-modal": 1}

Closes `ly_closes` The number of times the user closed a specific campaign. No {"content-rec-modal": 10}

Conversions `ly_conversions` The number of times the user converted on a specific campaign. No {"content-rec-modal": 1}

Milestones `ly_milestones` The number of times the user reached a milestone on a campaign. No {"engaged-donation": 1}

Goals `ly_goals` The number of times the user reached a goal on a campaign. No {"made-donation": 1}

Examples

Identifiers

Lytics ID (`_id`)

This is an automatically generated canonical ID managed by Lytics. It refers to the materialized profile and cannot be changed.

Current Lytics Cookie (`_uid`) and All Lytics Cookies (`_uids`)

`_uid` represents the Lytics anonymous 1st party cookie. This value is automatically captured with every `jstag.send()` call. The only way to explicitly set the value of `_uid`, which we do not recommend.

JavaScript

```
jstag.setid("somecustomvalue");
```

```
jstag.send();
```

Email (`email`)

JavaScript

```
jstag.send({
  email:"example@lytics.com"
});
```

Unique User ID (`uuid`)

JavaScript

```
jstag.send({
  uuid:"someuniqueuserid"
});
```

Details

First Name (`first_name`)

JavaScript

```
jstag.send({
  first_name:"John",
});
```

Last Name (`last_name`)

JavaScript

```
jstag.send({
  last_name:"Doe",
});
```

Title (`title`)

JavaScript

```
jstag.send({
  title:"President",
});
```

Phone (`phone`)

JavaScript

```
jstag.send({
  phone:"555-555-5555",
});
```

Cell (`cell`)

JavaScript

```
jstag.send({  
  cell:"555-555-5555",  
});  
Origin (origin)  
JavaScript
```

```
jstag.send({  
  origin:"loyalty_2022",  
});  
Language (language)
```

By default, the Lytics JavaScript SDK will collect language information based on the browser, but this can be overridden.

JavaScript

```
jstag.send({  
  _ul:"en-us",  
});  
Age (age)  
JavaScript
```

```
jstag.send({  
  age:25,  
});  
Companies (companies)  
JavaScript
```

```
jstag.send({  
  companies:["Lytics", "Pantheon"],  
});  
Gender (gender)  
JavaScript
```

```
jstag.send({  
  gender: "N/A",  
})  
Meta
```

Created (_created)

Lytics automatically generate this and represents the oldest event associated with the user.

Modified (_modified)

This is automatically generated by Lytics and represents the last time the user was modified.

Last Scored (_last_scored)

This is automatically generated by Lytics and represents the last time the users scores were updated.

Number of Aliases (_num_aliases)

This is automatically generated by Lytics and represents the number of aliases associated with the user.

Number of Days (_num_days)

This is automatically generated by Lytics and represents the number of days the user has existed.

Number of Events (_num_events)

This is automatically generated by Lytics and represents the number of events associated with the user.

Number of Streams (_num_streams)

This is automatically generated by Lytics and represents the number of streams associated with the user.

Stream Names (_streamnames)

This is automatically generated by Lytics and represents the names of the streams associated with the user.

User is Bot (is_bot)

This is automatically generated by Lytics and represents whether the user has been flagged as a bot or not.

Behavior

The following attributes are all computed in real-time as the profile evolves. Each of the behavioral attributes are sur aggregate summary of the user's behavior across various dimensions: Consistency, Frequency, Intensity, Maturity, M

Interests

Lytics Content (lytics_content)

The interest attributes are computed in real-time and represent the user's interest in various topics. These topics are rest Engine and then associated with the user based upon their interaction with content on your site.

Intelligence

Segment Membership (_segments)

This attribute displays an array of all segments the user is currently a member of. It updates in real-time based on va gments out of the box, requiring no additional setup. For detailed information on these audiences, refer to our Deve

Split & Split2 (_split & _split2)

These attributes are automatically generated by Lytics and represent a random value evenly distributed across users

Needs Message (needs_message)

This attribute is computed in real-time and represents the relative distance between now and the next predicted eve ser is likely to engage again.

Next Event (next_event)

This attribute is computed in real-time and represents the next expected event. It is stream specific and is useful for

Lookalike Model Predictions & Lookalike Model Percentiles (segment_prediction & segment_prediction_percentile)

This attribute is computed in real-time and represents the scores resulting from Lytics Lookalike and SegmentML mo ring no additional setup. For detailed information on these models, refer to our Developer Tier > Models documenta

Activity

General

First Seen & Last Seen (event_first_seen & event_last_seen)

Both of these attributes are automatically populated based upon the _e value in the jstag.send payload. By default L automatically populate the first_seen and last_seen attributes. Below is an example of collecting a custom event that w

JavaScript

```
jstag.send({
  _e:"custom_event"
});````
```

Channels (channels) [needs update]

JavaScript

```
jstag.send({
  _channel:"web",
});
```

Devices (devices)

JavaScript

```
jstag.send({
  _device:"desktop",
});
```

Hourly (hourly)

This attribute is automatically populated with a count of events per hour for the user.

Hour of Week (hourofweek)

This attribute is automatically populated with a count of events per hour of the week for the user.

Last Active Timestamp (last_active_ts)

This attribute is automatically populated with the last time an event was received in any stream for the user.

Last Channel Activities (last_channel_activities) [needs update]

JavaScript

```
jstag.send({  
  _channel:"web",  
});
```

Web

Domains (domains)

This attribute is automatically populated with the domains the user has been active on.

First Visit Timestamp (firstvisit_ts)

This attribute is automatically populated with the first time the user visited the site and sends data to the default stream.

Last Visit Timestamp (lastvisit_ts)

This attribute is automatically populated with the last time the user visited the site and sends data to the default stream.

Pageview Count (pageviewct)

This attribute is automatically populated with the number of _pv events received for the user.

JavaScript

```
jstag.send({  
  _e:"pv"  
});
```

Referring Domain (refdomain)

This attribute is automatically populated with the referring domain for the user.

JavaScript

```
jstag.send({  
  _ref:"umami.lytics.com",  
});
```

User Agent (user_agent)

This attribute is automatically populated based on the user agent of the browser. This attribute must be turned on in the configuration.

Visit Count (visitct)

This attribute is automatically populated with the number of visits the user has had based on presence of the _sesst attribute.

JavaScript

```
jstag.send({  
  _sesstart:1  
});
```

Visit City (visit_city)

This attribute is automatically populated with the city the user visited from based upon GeoIP.

Visit Country (visit_country)

This attribute is automatically populated with the country the user visited from based upon GeoIP.

Visit Region (visit_region)

This attribute is automatically populated with the region the user visited from based upon GeoIP.

Form Data (form_data)

Form data is a wildcard attribute that allows you to pass a number of key value pairs that all get stored under the form_data attribute.

S.

JavaScript

```
jstag.send({  
  formdata_fn:"John",  
  formdata_ln:"Doe",  
  formdata_someotherkey:"somevalue"  
});
```

Forms Submitted (forms_submitted)

JavaScript

```
jstag.send({  
  form_name:"newsletter"  
});
```

UTM Campaign Last (utm_campaign_last)

JavaScript

```
jstag.send({  
  utm_campaign:"holiday"  
});
```

UTM Campaigns (utm_campaigns)

JavaScript

```
jstag.send({  
  utm_campaign:"holiday"  
});
```

UTM Content Last (utm_content_last)

JavaScript

```
jstag.send({  
  utm_content:"recipe-1"  
});
```

UTM Contents (utm_contents)

JavaScript

```
jstag.send({  
  utm_content:"recipe-1"  
});
```

UTM Medium Last (utm_medium_last)

JavaScript

```
jstag.send({  
  utm_medium:"article"  
});
```

UTM Mediums (utm_mediums)

JavaScript

```
jstag.send({  
  utm_medium:"article"  
});
```

UTM Source Last (utm_source_last)

JavaScript

```
jstag.send({  
  utm_source:"google_ads"  
});
```

UTM Sources (utm_sources)

JavaScript

```
jstag.send({  
  utm_source:"google_ads"  
});  
UTM Term Last (utm_term_last)  
JavaScript
```

```
jstag.send({  
  utm_term:"example"  
});  
UTM Terms (utm_terms)  
JavaScript
```

```
jstag.send({  
  utm_term:"example"  
});  
Campaign  
Hover (ly_hover) [needs update]  
JavaScript
```

```
jstag.send({  
  pf_widget_id: "content-rec-modal",  
  pf-widget-event: "hover"  
});  
Impressions (ly_impressions) [needs update]  
JavaScript
```

```
jstag.send({  
  pf_widget_id: "content-rec-modal",  
  pf-widget-event: "show"  
});  
Closes (ly_closes) [needs update]  
JavaScript
```

```
jstag.send({  
  pf_widget_id: "content-rec-modal",  
  pf-widget-event: "close"  
});  
Conversions (ly_conversions) [needs update]  
JavaScript
```

```
jstag.send({  
  pf_widget_id: "content-rec-modal",  
  pf-widget-event: "conversion"  
});  
Milestones (ly_milestones) [needs update]  
JavaScript
```

```
jstag.send({  
  pf_widget_id: "engaged-donation-page",  
  pf-widget-event: "milestone"  
});  
Goals (ly_goals) [needs update]  
JavaScript
```

```
jstag.send({  
  pf_widget_id: "made-donation",  
  pf-widget-event: "goal"  
});  
Updated 10 months ago
```

Default Segments

Suggest Edits

Categorizing users based on their behaviors and characteristics is pivotal for effective audience targeting and personalization. A powerful tool for organizing users into meaningful groups based on shared attributes or behaviors. This section provides a list of default out-of-the-box.

Available Segments (Audiences)

-

Lytics audience segments apply predefined rules to each user profile as they update. Membership in these segments changes as users enter or exit the segment.

The following audience segments are all available out of the box, with no customization necessary in all Lytics pricing plans.

Name	Slug	Description	Definition
------	------	-------------	------------

All	all	Your entire user base, both anonymous and known.	<code>FILTER *</code>
-----	-----	--------------------------------------------------	-----------------------

Anonymous Profiles - 30 days	anonymous_profiles_30_days	Anonymous profiles older than 30 days	<code>FILTER AND (_num_aliases > 0, EXISTS 'uids') FROM anonymous_profiles_30_days</code>
------------------------------	----------------------------	---------------------------------------	------------------------------------------------------------------------------------------------

Anonymous Profiles - 60 days	anonymous_profiles_60_days	Anonymous profiles older than 60 days	<code>FILTER AND (_num_aliases > 0, EXISTS 'uids') FROM anonymous_profiles_60_days</code>
------------------------------	----------------------------	---------------------------------------	------------------------------------------------------------------------------------------------

Anonymous Profiles - 90 days	anonymous_profiles_90_days	Anonymous profiles older than 90 days	<code>FILTER AND (_num_aliases > 0, EXISTS 'uids') FROM anonymous_profiles_90_days</code>
------------------------------	----------------------------	---------------------------------------	------------------------------------------------------------------------------------------------

Anonymous Profiles	anonymous_profiles	Anonymous Profiles	<code>FILTER AND (_num_aliases = 1, EXISTS 'uids') FROM anonymous_profiles</code>
--------------------	--------------------	--------------------	-------------------------------------------------------------------------------------

Connected Customers	connected_customer_segment	Connected Customer Segment: Users who are active on 2 or more customer segments	<code>FILTER AND (_num_aliases > 0, EXISTS 'uids') FROM connected_customer_segment</code>
---------------------	----------------------------	---------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------

Unhealthy Profiles	default_unhealthy_profiles	Any user profile that is in an unhealthy state. Commonly caused by missing data or failed interactions.	<code>filter (_num_aliases > 0, EXISTS 'uids') FROM default_unhealthy_profiles</code>
--------------------	----------------------------	---------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------

Behavior: At Risk Users	ly_at_risk	People whose interaction behavior is changing for the worse.	<code>FILTER AND (score_frequency < 3) FROM ly_at_risk</code>
-------------------------	------------	--------------------------------------------------------------	--------------------------------------------------------------------

Behavior: Binge Users	ly_binge_user	People who show a lot of activity when they do interact with your brand.	<code>FILTER AND (score_frequency > 6) FROM ly_binge_user</code>
-----------------------	---------------	--------------------------------------------------------------------------	-----------------------------------------------------------------------

Engagement: Casual Visitors	ly_casual_visitor	People who show little activity when they do interact with your brand.	<code>FILTER AND (score_frequency < 3) FROM ly_casual_visitor</code>
-----------------------------	-------------------	------------------------------------------------------------------------	---------------------------------------------------------------------------

Engagement: Deeply Engaged Users	ly_deeply_engaged_users	People who show a lot of activity when they do interact with your brand.	<code>FILTER AND (score_frequency > 6) FROM ly_deeply_engaged_users</code>
----------------------------------	-------------------------	--------------------------------------------------------------------------	---------------------------------------------------------------------------------

Engagement: First-time Visitors	ly_first_time_visitor	People who are visited from the first time.	<code>FILTER visitct = 1 FROM ly_first_time_visitor</code>
---------------------------------	-----------------------	---------------------------------------------	------------------------------------------------------------

Behavior: Frequent Users	ly_frequent_user	People consistently interacting with your brand.	<code>FILTER score_frequency > 6 FROM ly_frequent_user</code>
--------------------------	------------------	--------------------------------------------------	------------------------------------------------------------------

Campaign Referral Interactions: Email	ly_from_email	People referred from email.	<code>FILTER utm_mediums INTERSECTS ("email") FROM ly_from_email</code>
---------------------------------------	---------------	-----------------------------	---------------------------------------------------------------------------

Campaign Referral Interactions: Paid	ly_from_paid	People referred from paid media.	<code>FILTER utm_mediums INTERSECTS ("cpc", "ppc") FROM ly_from_paid</code>
--------------------------------------	--------------	----------------------------------	-------------------------------------------------------------------------------

Campaign Referral Interactions: Social	ly_from_social	People referred from social media.	<code>FILTER utm_mediums INTERSECTS ("social", "twitter", "facebook") FROM ly_from_social</code>
----------------------------------------	----------------	------------------------------------	----------------------------------------------------------------------------------------------------

Web Activity: Has Visited Mobile Web	ly_has_visited_mobile_web	People who have visited on mobile web.	<code>FILTER EXISTS 'mobile_web' FROM ly_has_visited_mobile_web</code>
--------------------------------------	---------------------------	----------------------------------------	------------------------------------------------------------------------

Web Activity: Has Visited Web	ly_has_visited_web	People who have visited on web.	<code>FILTER channels INTERSECTS ("web") FROM ly_has_visited_web</code>
-------------------------------	--------------------	---------------------------------	---------------------------------------------------------------------------

Behavior: Infrequent Users	ly_infrequent_user	People not interacting with your brand.	<code>FILTER score_frequency < 3 FROM ly_infrequent_user</code>
----------------------------	--------------------	-----------------------------------------	--------------------------------------------------------------------

Location: International Visitors	ly_international_visitor	People who have visited outside the US.	<code>FILTER AND (EXISTS 'international') FROM ly_international_visitor</code>
----------------------------------	--------------------------	-----------------------------------------	----------------------------------------------------------------------------------

Email Capture Status: Known Email	ly_known_email	People who have a known email address.	<code>FILTER EXISTS email FROM ly_known_email</code>
-----------------------------------	----------------	----------------------------------------	------------------------------------------------------

Engagement: Moderately Engaged Visitors	ly_moderately_engaged_visitor	People who show average activity when they do interact with your brand.	<code>FILTER AND (score_frequency > 3, intensity < 76) FROM ly_moderately_engaged_visitor</code>
-----------------------------------------	-------------------------------	-------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------

Behavior: Moderately Frequent Users	ly_moderately_frequent_user	People occasionally interacting with your brand.	<code>FILTER AND (score_frequency > 3, intensity < 76) FROM ly_moderately_frequent_user</code>
-------------------------------------	-----------------------------	--------------------------------------------------	--------------------------------------------------------------------------------------------------------

Web Activity: Multi Session Visitor	ly_multi_session_visitor	People who have visited multiple times.	<code>FILTER visitct > 1 FROM ly_multi_session_visitor</code>
-------------------------------------	--------------------------	-----------------------------------------	------------------------------------------------------------------

Behavior: Perusers	ly_peruser	People who visit often but rarely interact deeply with your brand.	<code>FILTER AND (score_frequency < 3, intensity < 76) FROM ly_peruser</code>
--------------------	------------	--------------------------------------------------------------------	---------------------------------------------------------------------------------------

Engagement: Repeat Visitors	ly_repeat_visitor	People who have visited multiple times.	<code>FILTER visitct > 1 FROM ly_repeat_visitor</code>
-----------------------------	-------------------	-----------------------------------------	-----------------------------------------------------------

Casual Visitors	ly_reporting_casual_visitors	People who show little activity when they do interact with your brand.	<code>FILTER AND (score_frequency < 3) FROM ly_reporting_casual_visitors</code>
-----------------	------------------------------	------------------------------------------------------------------------	--------------------------------------------------------------------------------------

Deeply Engaged Users	ly_reporting_deeply_engaged_users	People who show a lot of activity when they do interact with your brand.	<code>FILTER AND (score_frequency > 6) FROM ly_reporting_deeply_engaged_users</code>
----------------------	-----------------------------------	--------------------------------------------------------------------------	-------------------------------------------------------------------------------------------

Frequent Users	ly_reporting_frequent_users	People who interact with your brand a lot.	<code>FILTER score_frequency > 6 FROM ly_reporting_frequent_users</code>
----------------	-----------------------------	--------------------------------------------	-----------------------------------------------------------------------------

Email	ly_reporting_from_email	People referred from email.	<code>FILTER utm_mediums intersects ("email") FROM ly_reporting_from_email</code>
-------	-------------------------	-----------------------------	-------------------------------------------------------------------------------------

Facebook	ly_reporting_from_facebook	People referred from Facebook.	<code>FILTER utm_sources intersects ("Facebook") FROM ly_reporting_from_facebook</code>
----------	----------------------------	--------------------------------	-------------------------------------------------------------------------------------------

Google	ly_reporting_from_google	People referred from Google search.	<code>FILTER utm_sources intersects ("Google", "Google Search") FROM ly_reporting_from_google</code>
--------	--------------------------	-------------------------------------	--------------------------------------------------------------------------------------------------------

Paid	ly_reporting_from_paid	People referred from paid media.	<code>FILTER utm_mediums intersects ("cpc", "ppc") FROM ly_reporting_from_paid</code>
------	------------------------	----------------------------------	-----------------------------------------------------------------------------------------

Social	ly_reporting_from_social	People referred from social media.	<code>FILTER utm_mediums intersects ("social", "twitter", "facebook") FROM ly_reporting_from_social</code>
--------	--------------------------	------------------------------------	--------------------------------------------------------------------------------------------------------------

Has Visited Mobile Web	ly_reporting_has_visited_mobile_web	People who have visited on mobile web.	<code>FILTER EXISTS 'mobile_web' FROM ly_reporting_has_visited_mobile_web</code>
------------------------	-------------------------------------	----------------------------------------	----------------------------------------------------------------------------------

Has Visited Web	ly_reporting_has_visited_web	People who have visited on web.	<code>FILTER channels INTERSECTS ("web") FROM ly_reporting_has_visited_web</code>
-----------------	------------------------------	---------------------------------	-------------------------------------------------------------------------------------

Infrequent Users	ly_reporting_infrequent_users	People who interact with your brand occasionally.	<code>FILTER score_frequency < 3 FROM ly_reporting_infrequent_users</code>
------------------	-------------------------------	---------------------------------------------------	-------------------------------------------------------------------------------

Last Visit Within 3 Months	ly_reporting_last_visit_within_3_months	People who have visited within the last 3 months.	<code>FILTER lastvisitct < 3 FROM ly_reporting_last_visit_within_3_months</code>
----------------------------	-----------------------------------------	---------------------------------------------------	-------------------------------------------------------------------------------------

Last Visit Within A Day	ly_reporting_last_visit_within_day	People who have visited within the last day.	<code>FILTER lastvisitct < 1 FROM ly_reporting_last_visit_within_day</code>
-------------------------	------------------------------------	----------------------------------------------	--------------------------------------------------------------------------------

Last Visit Within A Month ly_reporting_last_visit_within_month People who have visited within the last month. FILTER ly_reporting_last_visit_within_month < "now-1m") FROM user
 Last Visit Within A Week ly_reporting_last_visit_within_week People who have visited within the last week. FILTER ly_reporting_last_visit_within_week < "now-1w") FROM user
 Multi Session Visitor ly_reporting_multi_session_visitor People who have visited multiple times. FILTER visitct > 1
 Single Page Visitor ly_reporting_single_page_visitor People who have only visited one time. FILTER pageviewct = 1
 Web Activity: Single Page Visitor ly_single_page_visitor People who have only visited one time. FILTER pageviewct = 1
 Email Capture Status: Unknown Email ly_unknown_email People who do not have a known email address. FILTER ly_email_status = "unknown_email") FROM user
 Location: US Visitors ly_us_visitor People who have visited from the US. FILTER visit_country IN ("US")
 Browser / OS: Android ly_uses_android People who have used Android. FILTER devices INTERSECTS ("Android")
 Browser / OS: Desktop ly_uses_desktop People who used a desktop. FILTER devices INTERSECTS ("desktop")
 Browser / OS: iOS ly_uses_ios People who have used iOS. FILTER devices INTERSECTS ("IOS")
 Browser / OS: Mobile ly_uses_mobile People who have used a mobile device. FILTER devices INTERSECTS ("Android", "IOS", "Blackberry")
 Browser / OS: Other ly_uses_other People who have used other devices. FILTER devices INTERSECTS ("Blackberry", "Other")
 Lytics Currently Engaged smt_active Users who are currently engaging with your brand. FILTER AND (score_momentum > 0, score_frequency <= 5, score_intensity == 0, score_momentum == 0, score_quantity <= 3), NOT AND (score_quantity >= 50, score_momentum > 10) FROM user
 Lytics Disengaged smt_dormant Users who show minimal or no activity for a prolonged period of time. FILTER AND (score_momentum < 0, score_quantity <= 3), EXISTS score_momentum, _created < "now-1w") FROM user
 Lytics Previously Engaged smt_inactive Users who are currently disengaged with your brand, but had been previously engaged. FILTER AND (score_momentum < 0, score_frequency <= 5, score_intensity == 0, score_momentum == 0, score_quantity <= 3), NOT AND (score_quantity >= 50, score_momentum > 10) FROM user
 Lytics New smt_new Users who are new to your audience within the last week. FILTER _created >= "now-1w" FROM user
 Lytics Highly Engaged smt_power Users who engage most frequently and consistently of your users. FILTER AND (score_momentum > 40, score_frequency <= 5, score_intensity == 0, score_momentum == 0, score_quantity <= 3), EXISTS score_momentum, _created < "now-1w", NOT AND (score_quantity >= 50, score_momentum > 10) FROM user
 Lytics Unscored smt_unscored Users who have not registered enough activity to be scored by our behavioral algorithm. FILTER ly_score = 0
 Updated 6 months ago

Lead Capture

Suggest Edits

TL;DR

This document outlines how to leverage Lytics' real-time personalization engine to identify visitors who lack the strong engagement with other activation channels such as email.

Background

What is lead capture?

Traditionally, lead capture involves gathering information from individuals interested in your products or services, such as when people visit your website, subscribe to your newsletter, download resources, or interact with your content. However, as the consumer engagement strategy has undergone significant transformation.

Why is it important?

Accurate measurement of attribution and conversions is paramount to assessing the effectiveness of your marketing strategy. With the challenges posed by the deprecation of third-party cookies, particularly impacting top-of-funnel ad campaign strategy.

In an era of short attention spans, every interaction must be hyper-relevant and efficient, making lead capture instrumental. As marketing budgets face constraints, each dollar spent must yield maximum potential, and lead capture aids in optimizing. Building comprehensive consumer profiles representing behavioral patterns over time is essential to get the most out of your data. See it in action.

Lytics' real-time personalization engine lets you first identify the current web visitor, programmatically review their profile, and then red to their browser to close gaps in that profile.

Instructions

Below, you'll find a detailed guide on how to execute this use case. At the end, you'll find additional reading, which will take you further.

Requirements

Have access to an active Lytics account.

Ability to add JavaScript to your website via tag manager or CMS.

Entry-level knowledge of JavaScript. (Don't worry; it is mostly "copy and paste" here.)

Install Lytics Core SDK

Before executing this use case, your website must be properly configured with the core Lytics SDK (JavaScript tag). PL is present before continuing.

Activate Required Attributes

As an admin user of Lytics, you can enable or disable any attributes from being surfaced to the web via our JavaScript. You need to allow some of the required attributes for this particular use case.

Required Attributes:

Attribute

`anonymous_profiles` This computed attribute, available by default in all accounts, represents visitors who do not yet

`known_profiles` This computed attribute, available by default in all accounts, represents visitors with strong identifier

Configure Web Experience

To execute this use case, we will launch a simple lead capture form at the bottom left of your site. The lead capture will use a generic "dark" theme along with some boilerplate messaging. For best results, we recommend leveraging an offer or incentive to "sign up" and identify themselves. In return, you'll gain much knowledge about this visitor, strengthen your consumer relationship, and improve the overall experience of just the web.

JavaScript

```
<script type="text/javascript">
  jstag.on('pathfora.publish.done', function(topic, event){

    var module = new pathfora.Form({
      id: 'lead-capture-form',
      layout: 'slideout',
      theme: 'dark',
      headline: 'Sign up!',
      msg: 'Submit this form to get updates',
      formElements: [
        {
          "type": "email",
          "required": true,
          "label": "Email",
          "name": "email"
        }
      ]
    });

    var modules = {
      target: [{
        segment: "anonymous_profiles",
        widgets: [module]
      }]
    };

    pathfora.initializeWidgets(modules);
  });
</script>
```

Bonus Idea

You can extend this use case to progressively improve your visitor profiles by surfacing data capture forms specific to your audience. For example, if you have a known visitor, you can ask for more information. If you have an unknown visitor, only ask for email. As soon as you have an email, perhaps you want to know more about their preferences or lead source.

Configure Reporting

The Lytics platform has a pre-defined schema to facilitate this use case. As such, we recommend building a custom report to track the impact of this use case. You can measure the impact of this use case by comparing the number of unknown or anonymous visitors to known visitors with a strong identifier. There are many ways to configure this type of report to measure impact.

Create New Report

Add Anonymous vs. Known "Size" Component

Step 1: Once you have created your new report, add a component using the "+ Add New Component" button at the top of the report configuration page.

Step 2: Select "Size" as your component type.

Step 3: Name your component. We recommend something simple like "Anonymous vs. Known".

Step 4: Add an optional description to explain the goal of this report to ease the consumption for others later.

Step 5: Select the audiences you'd like to highlight. In this use case, we'll select "Anonymous Profiles" and "Known Profiles".

Step 6: Save your component. Please note it may take some time for the report to populate once you have configured the report. Once the report is populated, the data is good to go.

Add Impression "Composition" Component

Step 1: Following steps similar to the previous component, we'll add another component, but this time, we use the "Composition" component type.

Step 2: Name your component "Campaign Impressions," as this component will focus on showcasing impression information for a specific campaign.

Step 3: Select the "All" audience to ensure all visitors, regardless if they are known or unknown, are included in the report.

Step 4: Select the field "ly_impressions." This field represents an array of all campaigns each visitor has interacted with. The campaign ID from campaigns will match the "ID" you set above in the campaign configuration.

Step 5: Finally, replace "my_campaign_id" in the example below with the ID of your campaign above to highlight only that campaign.

Add Conversion "Composition" Component

Following the same steps as above, configure another "Composition" component, but this time, use the field "ly_conversions" instead of the impressions used previously.

That's it. Your report has been configured. As you begin to gain impressions and conversions, this report will populate with data.

Bonus!

Get additional guidance on how to pass interaction information related to your campaign to the following common systems.

Google Analytics [coming soon]

Custom BI via Data Warehouse [coming soon]

Execute Your Experience

Once you have configured and tested your Experience, activation is as simple as publishing the JavaScript code to your website or using another mechanism such as Drupal blocks or WordPress.

Updated 10 months ago

Content Recommendations

Suggest Edits

TL;DR

This document details the process of utilizing Lytics' real-time personalization engine to showcase content that resonates with your audience, either inline or through a pop-up modal on your website to boost engagement.

Background

How does Lytics make content recommendations?

Lytics Interest Engines provides a sophisticated approach for businesses to gain and leverage insights into which content

is achieved through a detailed enrichment and affinity generation process in real-time. While full details are available, the process can be distilled into four key steps:

Tracking Interactions: Monitoring visitor behavior in real-time to understand their engagement with the site.

Content Analysis & Topic Extraction: Using Google NLP and proprietary algorithms, Lytics automatically analyzes all content on the site to build a topic graph.

Interest Scoring: Assigning scores to reflect a visitor's level of interest in different topics, updated dynamically as new content is added.

Content Recommendation: Suggesting content that aligns with the visitor's demonstrated interests, leveraging real-time data.

This streamlined framework enables real-time personalization, enhancing user experience by continuously adapting content to individual users.

Why is it important?

Relevant content for individual users boosts marketing effectiveness and ROI by increasing engagement and conversion rates, optimizing resource utilization and higher customer retention. This targeted strategy enhances sales opportunities and brand loyalty.

See it in action.

Lytics' real-time personalization engine lets you first identify the current web visitor, programmatically review their profile, and then recommend their unique interests to maximize engagement.

Instructions

Below, you'll find a detailed guide on how to execute this use case. At the end, you'll find additional reading, which we recommend you explore further.

Requirements

Have access to an active Lytics account.

Lytics account must have had enough time to build the topic graph. Typically, 24 to 48 hours.

Ability to add JavaScript to your website via tag manager or CMS.

Entry-level knowledge of JavaScript. (Don't worry; it is mostly "copy and paste" here.)

Install Lytics Core SDK

Before executing this use case, your website must be properly configured with the core Lytics SDK (JavaScript tag). Please ensure the SDK is present before continuing.

Interest Scores & Content Collections

Your content must be classified before Lytics can offer content recommendations or insights. Typically, this classification process is automated, but if your content catalog is particularly large. To ensure the health of your content graph and interest engines, there are a few key areas to verify:

Verify Interest Scores on Profile

Ensure the Lytics Developer Tools Chrome extension is installed and Enabled.

Visit the domain(s) that have the Lytics JavaScript SDK installed.

Open the Chrome extension and navigate to the Profile section at the bottom.

From the Summary tab, you will see a section labeled Interests. Here, we will list any topics and their associated level of interest. If the data is not "available," proceed to the next step to ensure content has been classified.

Verify Content has Been Classified

If you have not already verified the quality of classification and metadata, please refer to our quickstart documentation.

From the Lytics interface, navigate to Decision Engine from the product switcher at the top left.

Using the left-hand menu, navigate to Content > Collections.

From the list of collections, there is likely only one; look for All Content in the list and the rows Size. This refers to the total number of items in the catalog. If you have more than one and reflect your total catalog.

For greater detail, you can also visit Content > Web Classification.

At the top of that page will be a summary dashboard of all content that has been successfully classified and cataloged. This provides a high-level overview of your content and its classification status.

Troubleshooting Common Problems

I have no lytics_content_enrich stream. (coming soon)

My content is not showing up at all. (coming soon)

My content is being classified, but none of my documents have images. (coming soon)

Select a content collection.

We will use the default collection outlined below for this exercise, but you may also create a custom collection with a

Collection Description

all_content This collection contains all classified documents. It comes out of the box in all Lytics accounts.

Ready to go a step farther and build a custom collection of your content to ensure recommendations are selected from on.

Activate Required Attributes & Segments

As an admin user of Lytics, you can enable or disable any attributes from being surfaced to the web via our JavaScript. You need to allow some of the required attributes for this particular use case.

Required Attributes:

Segment Description

all We will use the out-of-the-box segment containing all your site's users for demonstration. We recommend using all visitors.

Configure Web Experience

To execute this use case, we will demonstrate two examples. The first will be a single content recommendation in a page, and the second will be a set of recommendations inline to your website.

Example 1: Content Recommendation Modal

JavaScript

HTML

```
jstag.on('pathfora.publish.done', function(topic, event){
  var module = new pathfora.Message({
    id: 'content-rec-sample',
    layout: 'slideout',
    theme: 'dark',
    headline: 'Yummy content!',
    msg: 'We suspect you are going to want to check this out.',
    recommend: {
      collection: 'all_content'
    },
    cancelShow: false,
    okShow: false,
    variant: 3
  });

  var modules = {
    target: [{
      segment: "all",
      widgets: [module]
    }]
  };

  pathfora.initializeWidgets(modules);
});
```

Example 2: Inline Content Recommendation

In this example, we'll use a generic Drupal Block with pure HTML and CSS, but the same approach can be taken using

HTML

```
<style>
.rec-container {
  display: flex;
  justify-content: space-between;
  width: 100%;
}

.rec-item {
  box-sizing: border-box;
  flex: 1;
  padding: 10px;
}

.rec-title,
.rec-description {
  font-size: 14px;
  margin-bottom: 5px;
  word-wrap: break-word;
}

.rec-description {
  font-size: 12px;
}

.rec-img {
  height: auto;
  margin-bottom: 5px;
  max-width: 100%;
}
</style>
<div class="rec-container">
  <div class="rec-item" data-pfblock="lytics-recommend" data-pfrecommend="all_content">
    <div class="rec-title">
      <a data-pftype="URL"><strong data-pftype="title">Recommendation title loading...</strong></a>
    </div>
    <p>
      <img class="rec-img" alt="Recommendation related image">
    </p>
    <p class="rec-description" data-pftype="description">
      The recommendation description is loading...
    </p>
  </div>
  <div class="rec-item" data-pfblock="lytics-recommend" data-pfrecommend="all_content">
    <div class="rec-title">
      <a data-pftype="URL"><strong data-pftype="title">Recommendation title loading...</strong></a>
    </div>
    <p>
      <img class="rec-img" alt="Recommendation related image">
    </p>
    <p class="rec-description" data-pftype="description">
      The recommendation description is loading...
    </p>
  </div>
  <div class="rec-item" data-pfblock="lytics-recommend" data-pfrecommend="all_content">
    <div class="rec-title">
      <a data-pftype="URL"><strong data-pftype="title">Recommendation title loading...</strong></a>
    </div>
    <p>
```


</p>
<p class="rec-description" data-pftype="description">
The recommendation description is loading...
</p>
</div>

</div>

Updated 10 months ago

What is Vault?

Suggest Edits

Introduction

Vault is a centralized hub for managing security and other administrative features in Lytics. It provides a single place for all administrative features and user permissions, regardless of product. This means that the controls set in Vault apply to Conductor, Designer, and all other Lytics products with your Lytics account.

Take a quick tour of Lytics Vault.

Navigating Vault

Vault is comprised of the following sections:

Account Usage - an overview of your general account quota usage related to event consumption.

Account Settings - view and change necessary administrative settings and details of your account. These are sections that are available to all users.

Account Details

JavaScript Tag

Lytics API

Content

Security

Schema

Users - view a list of users accessing your Lytics account, change their details or permissions, remove a user, or invite new users.
Security

Access Tokens - view and manage a list of Lytics API access tokens. New tokens can be provisioned with specific permissions.

Authorizations - create, edit, view, and delete authorizations within Lytics. Authorizations are credentials to third party services that allow you to import and export jobs to run.

Account Setup

JavaScript Tag - learn how to install the Lytics JavaScript tag and validate the installation.

Who can access Vault?

Vault is focused primarily on account admins or those that have permission to administer settings, user access, etc. If you are not an account admin, not all sections outlined above may not be available.

If you are unable to access a section please reach out to your account administrator to request those adjusted permissions.

In general, Admins will be able to access all areas. Data Managers, Campaign Managers, Experience Managers, and Content Managers will have access to their respective sections. Finally, all Lytics users, regardless of role, should have access to view the JavaScript Tag and Schema sections. All users should be able to view their user profile where they can change details such as their Name, Email Address, Phone Number, etc. Your user profile will be accessible under the main Lytics navigation under "Manage My Profile."

Notable Changes

For existing customers, slight changes will impact your day-to-day management activities.

Product Switcher

Vault will be available directly from the primary product switcher at the top left of your Lytics interface. This will be your main point of entry into the Vault.

Account Usage

Our account usage data and quota meters received a much-needed facelift. The usage metrics act as the Vault "dashboard" for your account.

Account Usage Chart

Account Settings

Account setting sections are now accessible through the main navigation. These settings have received a minor facelift and have been recently updated for a more standard user experience. In addition, users are now prompted to save or discard their changes.

Users

The user list is now sortable and filterable based on name, email, who invited them, and how long they've been a Lytics user. In Vault, in addition to assigning new roles, an administrator can edit any user's name and email address.

The user invite form is now on its own page to improve the flow and experience of inviting users. Roles are now sorted in a way that gives users access to everything, while custom roles give a finer-grain definition of what the user can access. The details on each role are available in the documentation.

Access Tokens

The access tokens list is now sortable and filterable based on the name, description, the creator of the token, or when it was created. You can also view the access tokens you've already created, such as the roles assigned, when it was made, and when it will expire. In addition, you can create a new token with any combination of more granular access roles.

Authorizations

Authorizations are now to be created and updated only in Vault. You will still be able to utilize the auths you create in Conductor or Decision Engine, if you don't see an authorization you want to use for that job, you will be linked to the Vault page to create a new one.

JavaScript Tag Installation

This page remains unchanged, with simple styling changes and updated links to our documentation for troubleshooting.

Updated over 1 year ago

Accessing Accounts

Suggest Edits

Introduction

Under the account menu in the main navigation, find the Account Settings option.

You should see your account information, such as the name of your account, domain, the account owner's email, and the account number.

You can edit your account name, domain, and primary contact email from this page anytime. Once you change one of these, you will be prompted to save or discard your changes.

Account ID and account number are never editable. They are assigned at account creation and are permanent identifiers for your account.

Navigating Accounts

For users who have access to multiple accounts, you can quickly and securely navigate between Accounts using our new navigation menu.

1. Click "Switch Account"

2. Select Account

A menu will appear outlining the accounts you have access to. Simply select an account and a new tab will open with the account's data.

Updated over 1 year ago

Monitoring Metrics and Alerts

How to monitor Lytics, and alert when help is needed.

Suggest Edits

Monitoring

Monitoring and alerting is available on every job and every authorization within Lytics.

To set up alerting on your jobs or authorizations, you can set up a monitoring job from either the Job API or the Lytics UI.

If a source or destination job has failed, Lytics will show the latest error message on the Conductor Diagnostics Dashboard interface, and allow the job to be restarted if needed. The most detailed information for troubleshooting can be accessed via the Job API.

Additional generic monitoring on the Lytics system is available on our status page at lytics.statuspage.io.

Updated 10 months ago

Job Alerts

Monitor and alert on your import and export workflows

Suggest Edits

As systems are updated, permissions change, and networks are reconnected, sometimes situations will be encountered that are not built into Lytics Jobs. Lytics provides alerting through Slack, Microsoft Teams and email on import and export jobs.

Job Alerting Channels

Slack

Microsoft Teams

Email Alerting (Lytics Monitoring)

Keep in Mind!

Alerts can be configured for specific workflows, or for all workflows running in the account.

Consider having one general alert to an Operations Channel or Team for all alerts, or alerts for specific workflows managed by your team.

Consider which statuses to include

Alerts can be configured for statuses aside from failures. Workflows like critical batch import workflows can send a critical alert.

Understanding errors

If a source or destination job has failed, Lytics will show the latest error message on the Conductor Diagnostics Dashboard interface, and allow the job to be restarted if needed. The most detailed information for troubleshooting can be accessed via the Job API.

Updated 10 months ago

Monitoring Audit Logs

Suggest Edits

Account Activity Monitoring via Audit Logs API

Audit Logs provide visibility into system changes such as creating segments, updating schema, deleting items, new account data about changes in your account. These events are shown in many places inside Lytics, such as the history of a job, a job authorization, or password changes on the source for OAuth tokens. See our System Events API documentation for more details.

Updated 10 months ago

Exporting Audit Logs or Alert Jobs

Suggest Edits

Lytics System Events: Audit Logs and Alerts

In Lytics, audit logs and system alerts originate from a shared source: Lytics System Events. A System Event is simply an action taken by a user creating a job, segment, or role, or an error message indicating a job failure.

Export vs. Alert

Log Export: This is a job that either streams logs in real time or performs a batch export to an external source, such as a database or S3.

Log Alert: Alerts are triggered based on specified conditions and notify users when certain actions occur within the system.

In both cases, the content of exports or alerts depends on the filters you set up. Filtering is typically done using the System Event type (e.g. job creation, deletion, or failure).

Creating an Audit Log export (aka System Alert) Job via the UI

Creating a job to export your Audit and System Event logs is like creating any other job; for more information, see Data Export Jobs to create an export of your Audit logs or System Events.

Depending on your Provider, you can then select to export System Events.

For example, if you selected Google Cloud and you want to export the events to BigQuery.

Then, you'd want to select the BigQuery: Export System Events Job-Type tile. After choosing the Job-Type, it's like you can do a one-time or continuous export.

Creating an Alert based on System Events

Creating a job to alert on an Audit and System Event logs is like creating any other job; for more information, see Data Export Jobs to create an export of your Audit logs or System Events.

Depending on your Provider, you can then select to export System Events or Email Alerts.

For example, if you select to get an email when an Alert is triggered, then use Lytics as your provider and select Email Alerts.

Then configure what Subject you wanted to listen too and what Event Types (aka Verbs).

Filtering Audit logs

Audit logs can be filtered by

Subject Type: what the event is about, such as work, workflow, user, campaign. See the list of subject types below.

Subject ID: identifier of a subject, such as work ID, workflow ID, campaign ID, etc.

Verb: action described by the event performed on a subject. See the list of available verbs below.

Verb Description Frequency

synccomplete For the completion of one synchronization cycle. Emitted when a work cycle finishes successfully. This occurs once per sync or when there is a sleep cycle. Real-time

update For when work configuration is modified. It may occur multiple times per work. Real-time, batch

created For when works are created. This only occurs only once per work. Real-time, batch

deleted For when a work is deleted. Real-time, batch

synced For the completion of one sync unit (multiple units may happen per sleep cycle). Real-time

completed For the final successful completion of a work. This occurs once per work. Batch

started For the first time, work is started. This occurs once per work. Real-time, batch

failed For the final failure of a work. This occurs once per work unless work is bounced. Real-time, batch

syncing For the start of a series of sync cycles for a work. Real-time

Subject Type Description

account Represents account-related actions or changes.

auth Refers to authentication events, including 3rd party access control activities with service accounts.

user Actions related to user accounts, such as creation, update, or deactivation.

campaign Actions or modifications related to marketing campaigns.

data Covers data management events, such as data ingestion and updates.

entity Refers to actions involving user entities aka Profiles.

experience Actions associated with experience.

journey Pertains to events tracking customer journeys and related updates.

program Events involving the creation or modification of programs within the system.

provider Refers to data provider configurations or integrations.

query Covers actions involving Lytics queries.

report Refers to generation, access, or modification of Lytics reports.

rollup Involves aggregation events, such as data roll-ups for analytics.

schema Actions associated with data schema configuration and updates.

schematable Refers to operations involving schema tables or data structure definitions.

scoring Covers actions involving user scoring mechanisms within the system.

segment Events related to segment creation, modification, or deletion.

segmentcollection Refers to collections of segments and associated actions.

segmentml Involves machine learning (ML) processes related to segment analysis.

stream Actions or updates related to a Lytics stream.

subscription Events involving subscriptions, such as subscription creation or cancellation.

topic-document Refers to topic modeling or document processing events.

variation Refers to A/B testing or variation setup within experiences or campaigns.

work Covers general work-related actions or tasks within the system.

workflow Actions associated with workflow creation, modification, or execution

Job Status Monitoring via Webhooks

Job-status events can be observed by creating a webhook subscription that POSTs data (or JSON) to a specific URL. This can be used as a downstream for your monitoring use cases. Some common examples include listening for audience exports created/updated, or when a given integration fails.

Work related filters

For events related to the subject type work, the following verbs may be emitted:

synccomplete - Emitted when a sync operation is completed.

updated - Emitted when the job configuration is modified.

created - Emitted when a new job is created.

deleted - Emitted when a job is terminated.

synced - Emitted when the job sync operation completes.

completed - Emitted when a job has finished successfully.

started - Emitted when a job begins execution.

failed - Emitted when a job encounters an error.

syncing - Emitted while a job is actively syncing.

Building custom Alerts/Exports using our Webhook Integration

Overview

Lytics provides a webhook integration that allows you to monitor and react to system events in real-time. This integration can be used to trigger actions, and maintain synchronization with your other systems based on events occurring within Lytics.

Configuration Examples

Basic Webhook Setup

The following example shows how to subscribe to multiple event types and send them to a webhook endpoint:

JSON

```
{
  "config": {
    "system_event_multiple_types": ["updated", "created", "failed"],
    "webhook_url": "YOUR_WEBHOOK_ENDPOINT",
    "headers": {
      "custom-header": "header-value"
    }
  },
  "workflow": "webhook_system_events"
}
```

Slack Integration

You can integrate Lytics system events with Slack using a custom JSON template. This example shows how to send failed events to Slack.

JSON

```
{
  "config": {
    "system_event_type": "failed",
    "webhook_url": "YOUR_SLACK_WEBHOOK_URL",
    "json_template": "local note = if std.objectHas(event.data, \"notes\") then event.data.notes else \"A work failed\\n\". See info go to: https://activate.getlytics.com/data/integrations/work/%s?aid=%s\" % [event.data.subject_id, event.data.subject_id]"
  },
  "workflow": "webhook_system_events"
}
```

Advanced Filtering

You can filter events by source type and ID using the system_event_sources configuration:

Filter by Specific IDs

JSON

```
{
  "config": {
    "system_event_multiple_types": ["updated", "created", "failed"],
    "system_event_sources": {
      "work": [
        "work-id-1",
        "work-id-2"
      ]
    },
    "webhook_url": "YOUR_WEBHOOK_ENDPOINT",
    "headers": {
      "custom-header": "header-value"
    }
  },
  "workflow": "webhook_system_events"
}
```

Monitor All Events of a specific Verb Type

To receive events for all works in your account, use an empty array.

JSON

```
{
  "config": {
    "system_event_multiple_types": ["updated", "created", "failed"],
    "system_event_sources": {
      "work": []
    },
    "webhook_url": "YOUR_WEBHOOK_ENDPOINT",
    "headers": {
      "custom-header": "header-value"
    }
  },
  "workflow": "webhook_system_events"
}
```

Best Practices

Ensure your webhook endpoint can handle the expected volume of events

Implement proper error handling and retry logic in your webhook receiver

Use HTTPS endpoints for secure data transmission

Monitor webhook delivery success rates and implement appropriate alerting

Support

For additional assistance or questions about webhook integration, please contact Lytics support or consult our API docs.

Updated 2 months ago

Monitoring Metrics

Suggest Edits

Introduction

Lytics provides a variety of valuable metrics that downstream monitoring tools, such as Google Stackdriver or New Relic, can use for alerting and monitoring, you can have visibility into Lytics within your existing ecosystem.

Consuming Metrics Downstream

Once you have connected Lytics to your monitoring tool, there are various ways you can apply metrics from Lytics to your tool. Here are some examples:

Active monitoring - defined by alerting, requires additional configuration within your downstream tools to monitor Lytics metrics.

On-call distribution lists - control who within your operational teams to inform.

Quiet hours - manage within a tool where you are already doing that for other metrics.

Correlation of metrics - show existing metrics (e.g., website performance) in the context of Lytics metrics.

Operational users - watch for signals without creating a Lytics admin user account.

Anomaly detection - use threshold-based alerts that most monitoring tools have that go beyond the capabilities Lytics Platform Monitoring via Metric API
The Metric API provides access to a variety of metrics that are recorded in the Lytics platform. This API allows you to access any workflow-specific metrics.

Heartbeats: metrics with a value of 1 for "up & healthy" and 0 for "not healthy" (or missing).

Name	Description	Updated
monitoring_heartbeat	A simple 1 (up) for each minute a workflow runs indicates the overall integrations platform.	1
collection_count	A metric for the count (gauge) for the 1-minute window in total events ingressed (web collection or stream_count)	1
stream_count	Metric per stream for a count of events seen this cycle.	Every hour

*Availability of Metrics: the Lytics Metric API and all Lytics export workflows run inside Lytics' work runtime system in which users can move between servers, potentially resulting in 1 or 2-minute gaps in metrics. Therefore, alerts on single heartbeat metrics have a window of 5.

Updated 3 months ago

Usage Metrics

Suggest Edits

Account Usage Metrics

The Account Usage section shows how many inbound and outbound events Lytics has received for your account. If you want to see how much of your inbound event quota you have used for each quota period.

In the Lytics UI, select Vault from the product switcher. Or, if you are already in Vault, select Usage from the main menu.

This view offers a quick way to gauge the quantity of data Lytics has received from your various Sources. Of course, if you want to view more granular details of all connected Sources from the Conductor interface.

Inbound Events

An inbound event is a record of activity collected or uploaded to Lytics from a customer data source, such as a record of a user's purchase or a phone number record. The inbound events graph shows your total inbound events, which can be displayed daily, weekly, or monthly.

The inbound quota meter displays the number of events recorded in the quota period, your quota limit, and the meter's status. For short-term accounts, quota periods begin on the first of each month and end on the last day of the month. For annual term accounts, quotas (unless otherwise stated in your contract) only count for production accounts.

Monitoring the Inbound Event Quota

Automated alerts for Inbound Event Quotas can be configured to sent to Microsoft Teams, Slack, or directly to an email address.

Outbound Events

An outbound event is any record of data exported from Lytics, whether manually or automatically. The outbound events graph shows your total outbound events, which can be displayed daily, weekly, or monthly.

Updated 10 months ago

Managing Users

Suggest Edits

Managing Your User Profile

To access your profile, click "Manage My User" from the account menu at the bottom of the primary navigation.

You'll have access to personal information and usage statistics from the resulting profile page. It doubles as an ID card for your user profile.

user profile

You can edit your name and email address. Both fields are required. A phone number is only required for 2FA.

Click the "Change Password" button to change your password.

Resetting Passwords

The password must be reset by the user whose password is being reset. Passwords cannot be reset on behalf of other users.

To reset your password, go to Manage My Profile in the account menu and click the Change Password button.

A modal will open, prompting for a new password can be entered.

change password

f

Changing a password will immediately invalidate all current sessions across all machines for the user.

Managing Account Users

Click "Manage Users" in the account menu to manage account users.

manage users menu

Modifying User Permissions

User roles define the amount of access and permissions a Lytics user has when logged in and through the Lytics API.

vault roles

A user may have any number of these roles, and the permissions for multiple roles are additive. For example, a user with the Admin and Data Manager roles has access to all permissions granted by both roles.

Users with the Admin role can control other users' roles within your organization via the "Manage Users" option from the account menu.

Managing Personally Identifiable Information

You can indicate any user fields in your account that contain Personally Identifiable Information (PII) via the private field settings. Only users with the Admin, Data Manager, or User Search roles can view or edit PII. You should verify with Lytics Support that the field is marked as PII to ensure these fields are hidden there.

What does a user of each role have access to?

A user's role determines which parts of the Lytics app they can access. Here is a breakdown of what is shown in the table below.

Admin Audience Manager Campaign Manager Content Manager Data Manager Discovery Insights Experience Manager

Dashboard

Dashboard x x x x x x x x x x x

Goals

Goals x x

Experiences

Experiences x x x

Audiences

Audiences x x x x x x x x

Discovery x x

User Search x x x

User Profiles x x x

GDPR Profile Delete x x

Content

Overview x x x x

Topics x x x x

Collections x x x x

Data

Jobs / Auths x x x x x

Data Streams x x

User Fields x x

Queries x x

Schema Audit x x

Personalize x x x x

Look-a-like Models x

For access to look-a-like models you will need to be an account admin or have a combination of Discovery Insight + Admin role.

What tasks can a user of each role perform?

Roles define a set of permissions the user has, which also dictates what actions they can take in the app and through the API by feature.

Personally Identifiable Information (PII)

To shield PI from users who should not have access, you will need to use the private fields account setting to mark the Admin, Data Manager, or User Search roles.

	Admin	Audience Manager	Campaign Manager	Content Manager	Data Manager	Discovery Insights	Experience Manager
View	x	x		x			

Audiences (without PII)

	Admin	Audience Manager	Campaign Manager	Content Manager	Data Manager	Discovery Insights	Experience Manager
View	x	x	x		x	x	x
Create	x	x	x		x	x	x
Edit	x	x	x		x	x	x
Duplicate	x	x	x		x	x	x
Delete	x	x	x		x	x	

Content Topics

	Admin	Audience Manager	Campaign Manager	Content Manager	Data Manager	Discovery Insights	Experience Manager
Blocklist	x	x					

Content Collections

	Admin	Audience Manager	Campaign Manager	Content Manager	Data Manager	Discovery Insights	Experience Manager
View	x	x	x	x			
Create	x	x	x				
Edit	x	x	x				
Duplicate	x	x	x				
Delete	x	x					

Authorizations (for imports & exports)

	Admin	Audience Manager	Campaign Manager	Content Manager	Data Manager	Discovery Insights	Experience Manager
View	x	x	x	x	x		
Create	x	x	x	x	x		
Edit	x	x	x	x	x		
Delete	x	x	x	x	x		

Jobs (imports & exports)

The former "Integrations" tab is now comprised of the "Jobs" and "Authorizations" sections, which allow you to manage

	Admin	Audience Manager	Campaign Manager	Content Manager	Data Manager	Discovery Insights	Experience Manager
View	x	x	x	x	x		
Create	x	x	x	x	x		
Pause	x	x	x	x	x		
Update	x	x	x	x	x		
Delete	x	x	x	x	x		

Accounts & Users

	Admin	Audience Manager	Campaign Manager	Content Manager	Data Manager	Discovery Insights	Experience Manager
Invite User	x						
Manage User Roles		x					
Create Account		x					
Edit Account		x					
View Account Usage		x					

Personalize Campaigns

	Admin	Audience Manager	Campaign Manager	Content Manager	Data Manager	Discovery Insights	Experience Manager
View	x	x		x	x		
Create	x	x		x			
Edit	x	x		x			
Duplicate	x	x		x			
Delete	x	x					

Additional API-Only Features

All roles have API read access to topic rollups, segment collections, and SegmentML. Marked below are the roles with

	Admin	Audience Manager	Campaign Manager	Content Manager	Data Manager	Discovery Insights	Experience Manager
Content Topic Rollups	x	x	x	x			
Content Classification	x	x	x	x			
Queries API	x	x					
Segment Collections	x	x	x	x	x	x	x

Lookalike Models x x

Subscription x x x

Inviting Users

Click the Create New button from the user list to invite a new user to the account.

vault invite user

Inviting a new user will prompt for an email address and roles to select the appropriate level of access the user will have. The new user will get an email with a link that will take them to the Lytics account login screen.

Removing Users

From the user list, select the user you wish to delete. You can remove this user from Lytics by clicking the Delete User button.

Updated over 1 year ago

Single Sign-On

Suggest Edits

f

Lytics supports Single Sign-On (SSO) by using Google Cloud Identity Platform as a service provider using SAML protocol. This is a SP-initiated SSO, meaning that the Service Provider (SP) initiates SSO. IDP-initiated SSO has been deprecated and is no longer supported by Lytics.

Overview

Single Sign-On (SSO) allows employees to safely and securely access a host of Internet tools with a single username and password. You should consider leveraging Single Sign-On:

SSO helps the end user. Implementing SSO means employees are required to commit just one password to memory instead of many. A unique and secure password without writing it down. SSO reduces password fatigue and frustration and makes the employee's life easier.

SSO benefits the company. Fewer passwords to remember equals fewer calls to the helpdesk, resulting in more time for the helpdesk (which is oftentimes outsourced) means a reduction in cost to the company as well.

SSO improves security. When employees are required to remember numerous passwords, it is easy to fall into lazy habits. Passwords are more susceptible to hacking. Or writing passwords down on Post-it notes where they can easily be stolen or fall into the wrong hands.

SSO helps with compliance. SSO helps companies increase control over the user's access to certain information as well as audit user activity. This is important for organizations to comply with HIPAA, SOX, and other regulatory groups.

SP-Initiated SSO

Lytics supports enterprise Single Sign-On (SSO) by using Google Cloud Identity Platform as a service provider using SAML protocol. This is a SP-initiated SSO, meaning that the Service Provider (SP) initiates SSO.

Once implemented, users will log in to Lytics via a special SSO form that only requires an email address. Lytics will redirect the user to complete the login. Once the IdP verifies credentials, the pop-up will close, redirecting the user to a logged-in instance of Lytics. The user's IdP will communicate with the Lytics APIs, which use Google Cloud Identity Platform to validate the login. This document describes how to set up SSO that uses SAML.

Service Provider Configuration

To configure SAML for the Lytics service provider, some information is required about your IdP. If you have a metadata file, that's appropriate, but please ensure that the following information is provided to Lytics Support:

Entity ID

Sign In URL

X509 Signing Certificate

Further configuration details, such as mappings, may need to be provided, but the Lytics implementation only requires the above information. Once it has been received, Lytics can configure the SAML connection in the Google Cloud Identity Platform.

IdP Configuration

After Lytics configures the SAML connection on the service provider, Lytics Support will provide the following key fields that you need to enter in their IdP.

Assertion Consumer Service (ACS) URL (aka postback or callback URL)

Entity ID of the Service Provider

Sign-in URL

With this information, your IdP connection can be configured to complete the SSO integration.

At this time, Lytics does not support providing this information in an XML metadata file. Additional information about

Testing SSO

Once all the information has been configured in both the IdP and the Lytics service provider, you can test and verify SSO as your only sign-in method, please disable any password restriction or expiration settings that may have been

During the testing process, Lytics can be configured to allow both SSO login and regular username and password (or without disrupting the day-to-day usage of the app.

If requested, once the SSO implementation has been tested and verified, Lytics can disable the use of other login types

Troubleshooting SSO

If it's known that SSO will be added to an account, the user email addresses added to the account should match the match, the login will fail as Lytics will not be able to verify that there is a user with that email address.

For instance, if the email listed in the IdP is abc@123 and within Lytics, it is def@456, then there will be potentially mismatch on Lytics.

If the user enters abc@123 into the IdP pop-up, it will immediately fail as we use the Lytics account user to determine the following:

SSO Error State

However, should the user enter def@456 (their Lytics account email), they will be redirected to their IdP, but the verification fails to Lytics.

To remedy this situation, you would need to create a new user within Lytics with the email address abc@123 for the user or an account using the Managing Users guide as a reference.

Lytics Support can assist in the troubleshooting process. When testing for the first time, Lytics can enable logging to Lytics can help debug if you provide information on the login attempt, such as the login time, user, and account.

If you're encountering a verification error, but you've checked that your emails from the IdP and Lytics match, this may be on the SP side. Contact Lytics Support with details of the issue, and our team can coordinate a fix.

Account Structure with SSO

It should also be noted that primary accounts (master accounts) are decided as the first account that a user was added to. Users often have different primary accounts. This is important for SSO as it will also be the account the user is logged into at the time of login on one account and a user attempts to log in using Google OAuth or their username and password, the user is logged into the first account.

Add that user to your IdP.

Add other logging methods (Google OAuth, username/password).

Remove that user from all accounts and then add them back, with the first account being the one you want to be the primary account.

Assigning Roles using SSO

Configuring Role Assertions for SSO in Lytics

The Lytics team can configure Single Sign-On (SSO) to assign roles via SSO. This configuration must be applied to the accounts that users need access. For more information, refer to the account structure with SSO.

Important Note:

Users cannot be assigned to additional accounts via SSO alone; they must be manually invited to each Lytics account.

To set up role assignments via SSO, customers must configure their Identity Provider (IDP) to include a group assertion. The group must be specified in the format:

lytics_<AID>_<Role>

Example of a SAML assertion for a user whose default account is 123 and also has access to account 234:

XML

```
<samlp:Response xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
  xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
  xmlns:samlp="urn:oasis:names:tc:SAML:2.0:protocol" xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    Destination="https://api.lytics.io/api/user/verifyauth">
    Version="2.0">
    <saml:Assertion xmlns:saml="urn:oasis:names:tc:SAML:2.0:assertion"
      ID="id1" IssueInstant="2024-08-29T16:02:42Z"
      Issuer="https://your-idp.example.com">
      Version="2.0">
      <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">123456</ds:Signature>
      <saml:AttributeStatement>
        <saml:Attribute Name="Email">
          <saml:AttributeValue xsi:type="xs:string">test-user@test.com</saml:AttributeValue>
        </saml:Attribute>
        <saml:Attribute Name="groups">
          <saml:AttributeValue
            xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
            xsi:type="xs:string">lytics_123_content_manager</saml:AttributeValue>
          <saml:AttributeValue
            xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
            xsi:type="xs:string">lytics_123_admin</saml:AttributeValue>
          <saml:AttributeValue
            xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
            xsi:type="xs:string">lytics_234_admin</saml:AttributeValue>
          </saml:Attribute>
        </saml:AttributeStatement>
      </saml:Assertion>
    </samlp:Response>
```

Note: This example is simplified to focus only on the group's assertion; a real-world example would be more detailed.

Assignable roles

The following roles can be assigned:

Assignable Role	Name	Description
-----------------	------	-------------

admin	Account Admin	Grants read/write access to all features and management functions.
-------	---------------	--------------------------------------------------------------------

content_manager	Content Manager	Grants read/write access to content classification, collections, and topics.
-----------------	-----------------	------------------------------------------------------------------------------

segment_manager	Audience Manager	Grants read/write access to audiences.
-----------------	------------------	----------------------------------------

marketer	Marketer	Grants read/write access to audiences and campaigns but cannot publish campaigns or export
----------	----------	--------------------------------------------------------------------------------------------

user_search	User Search	It grants the ability to search for users, view their profiles
-------------	-------------	----------------------------------------------------------------

report_viewer	Discovery Insights	Grants access to explore Discovery insights
---------------	--------------------	---------------------------------------------

report_manager	Report Manager	Can view, create, and delete audience insight reports
----------------	----------------	-------------------------------------------------------

observer	Observer	Grants read access to audiences and campaigns.
----------	----------	------------------------------------------------

data_manager	Data Manager	Grants read/write access to schemas, queries, users, content, and integrations, including
--------------	--------------	-------------------------------------------------------------------------------------------

campaign_manager	Campaign Manager	Grants read/write access to campaigns and audiences without PII, including
------------------	------------------	----------------------------------------------------------------------------

orchestrate	Goal Manager	Grants read/write access to Goals and Experiences features.
-------------	--------------	-------------------------------------------------------------

experience_manager	Experience Manager	Grants read/write access to Experiences.
--------------------	--------------------	------------------------------------------

Implementing SSO with Okta

This document will walk you through how to implement Single Sign-On to the Lytics application with Okta as an identity provider application. Still, while the partnership is being established, you can follow these instructions to set up the custom implementation of the implementation.

From your Okta Administration panel, navigate to Applications and then Add Application and Create New App.

Under Platform, select Web, and for Sign on Method, choose SAML 2.0.

Application Setup Okta

Under General Settings, you can enter the following values:

App name: Lytics

App Logo: You can save and upload the following image of the Lytics Logo:

Lytics Logo

Under SAML Settings, enter the following values:

Single Sign-on URL: <https://api.lytics.io/api/user/sso>

Make sure Use this for Recipient URL and Destination URL is not selected). Also select Allow this app to request other

Requestable SSO URLs: Add the following two URLs:

<https://api.lytics.io/api/user/verifyauth> (index 0)

<https://api.lytics.io/api/user/sso> (index 1)

Recipient URL: <https://api.lytics.io/api/user/verifyauth>

Destination URL: <https://api.lytics.io/api/user/verifyauth>

Audience URI (SP Entity ID): app.lytics.com

Name ID format: EmailAddress

NOTE: The <https://api.lytics.io/api/user/sso> URL should only be used for Okta. Other SSO providers use the default URL.

Okta Configuration

Okta Configuration

Click through the next step, and select Finish.

You can add users to view this application in their portal using the Assignments tab.

Navigate to the Sign On tab. And under settings, click on View Setup Instructions.

SAML Setup Instructions Okta

You will need to gather the information on this page and send it to Lytics for configuration of the Service Provider:

Identity Provider Single Sign-On URL

Identity Provider Issuer

X.509 Certificate

Once Lytics has completed the service provider implementation, you may begin to test the SSO implementation through

Implementing SSO with OneLogin

This document will walk you through how to implement Single Sign-On to the Lytics application with OneLogin as an custom application in OneLogin, which covers the IDP configuration portion of the implementation.

From the Administration menu, select Applications and then click Add App. Search for SAML Test Connector (Advanced)

Under Configuration > Portal enter the following:

Display Name: Lytics

Make sure Visible in Portal is selected.

For the rectangular icon, you can save and upload the following image:

Lytics Rectangle

For the square icon, you can save and upload the following image:

Lytics Square

Click Save to continue to the configuration process. Then click on the Configuration tab to set up the SAML details.

Enter the following into the Application details:

Audience (EntityID): app.lytics.com

Recipient: <https://api.lytics.io/api/user/verifyauth>

ACS (Consumer) URL Validator: <https://api.lytics.io/api/user/verifyauth>

ACS (Consumer) URL: <https://api.lytics.io/api/user/verifyauth>

Login URL: <https://app.lytics.com/login/sso>

SAML initiator: Service Provider

SAML nameID format: Email

OneLogin Config

Click on the Save to save your configuration changes.

You may configure any additional access details, such as users accessing this app in their portal for testing the integrations.

Click on the SSO tab, and you will need to gather the information on this page and send it to Lytics for configuration.

X.509 Certificate (click View Details to see the full cert).

Issuer URL

SLO Endpoint (HTTP)

Once Lytics has completed the service provider implementation, you may begin to test the SSO implementation through the Lytics portal.

Updated 3 months ago

Account Settings

Suggest Edits

Several account settings can be managed within the Vault interface. To access account settings, select "Vault" from the top navigation bar.

Account Settings" from the secondary account menu at the bottom left.

Configuration Options

Account Details

Configuration options related to your primary account. Here, you can adjust your account name and primary contact information.

JavaScript Tag

Configuration options related to the core Lytics SDK, such as client-side integration configuration, Pathfora Personalization, and more.

Lytics API

Configuration options related to the Lytics Personalization API. Here, you'll be able to manage detailed access settings for the API.

Content

Configuration options related to Lytics' content classification services. Here, you'll be able to configure settings related to content classification, content tagging, and more.

Security

Configuration options related to account security, such as password complexity, session duration, and two-factor authentication.

AI & Modeling Controls

Configuration options related to Lytics' suite of AI and modeling capabilities such as private fields, audience segmentation, and more.

Schema Controls

Configuration options related to your core profile schema.

Accessing Vault

Product Switcher Account Menu

Updated about 1 year ago

Account Details

Configuration of account details such as name and primary contact.

Suggest Edits

The following configuration options are available within the account settings Account Details section.

Account Name

Label for your account, which shows up in the account switcher, should you have access to more than one Lytics account.

Domain

Primary domain associated with your account.

Contact

Email address of the primary account contact.

Allow Access via API

To share profile data, such as audience membership, with integrations such as Google Analytics, profiles must be accessible via API.

Account ID (Non-configurable)

Your unique Lytics account identifier. Your primary identifier used in all API calls.

Account Number (Non-configurable)

Your unique Lytics account number. Used primarily for quick reference to which account you are currently within.
Updated about 1 year ago
JavaScript Tag Config
Configuration options for the core JavaScript SDK.

Suggest Edits

The following configuration options are available within the account settings JavaScript Tag section.

Client Side Integrations

A set of client-side integrations is facilitated by the core Lytics SDK. Each of those integrations is either deployed or disabled in the "Client Side Integrations" section. Each of those options follows the pattern outlined below.

Checking the box enables the integration and will result in either audience membership or attributes being passed to the integration. If no configurations are also in place on the page.

Client-side integrations supported out of the box:

Amazon DSP: Allow Lytics to pass the current visitor's _uid to Amazon DSP for improved identity resolution.

Amazon DSP Confirmation Events: If false, confirmation events will not be sent to the amazon_dsp stream upon successful events.

Criteo: Allow Lytics to pass the current visitor's _uid to Criteo and receive a Criteo GUM ID for improved identity resolution.

Google Ads Partner API: Allow Lytics to use the Google Partner API to make Google Ads calls.

Google Analytics 4 (GA4): Allow Lytics to pass the current visitor's _uid and audience membership to GA4.

Google DV360: Allow Lytics to pass the current visitor's _uid to Google DV360 and receive a unique DV360 ID for improved identity resolution.

Krux: Allow Lytics to pass the current visitor's _uid to Krux for improved identity resolution.

Lotame: Allow Lytics to pass the current visitor's _uid to Lotame for improved identity resolution.

Taboola: Allow Lytics to pass the current visitor's _uid to Taboola and receive a unique Taboola ID for improved identity resolution.

The Trade Desk: Allow Lytics to pass the current visitor's _uid to The Trade Desk and receive a unique Trade Desk ID for improved identity resolution.

Yahoo Ads: Allow Lytics to pass the current visitor's _uid to Yahoo and receive a unique Yahoo ID for improved identity resolution.

In addition to those outlined above, many automatic client-side integrations are included within the Lytics core SDK.

Lytics will sync audience membership with other tools if those tools have been configured on your site, too. For instance, Lytics will sync audience membership with tools like Facebook, Google Analytics, and others present on the page. To prevent these syncs, each integration can be blocked by adding the proper key to the "Integrations" section.

In this example, we have added the "google_analytics_dimensions" slug to the blocklist to prevent sending profile data to Google Analytics. Below is a list of available client-side integrations and their associated slugs:

Name	Slug	Description
------	------	-------------

Adroll	adroll	Adds a record named adroll_segments to the user, which houses audience membership.
--------	--------	------------------------------------------------------------------------------------

AddThis	add_this	Adds the current audience membership to the __attag cookie.
---------	----------	-------------------------------------------------------------

Amazon DSP	amazon	Adds the Amazon DSP pixel to the site.
------------	--------	----------------------------------------

Facebook Ads	facebook_ads	Pushes Lytics audience membership as an array to the custom Lytics Audiences value.
--------------	--------------	-------------------------------------------------------------------------------------

Google Analytics	google_analytics_dimensions	Pushes audience membership as a comma-separated string to a pre-defined custom dimension. The sync occurs for a user.
------------------	-----------------------------	-----------------------------------------------------------------------------------------------------------------------

Google DFP	google_dfp	Sets audience membership to a targeting param with the name LyticsSegments.
------------	------------	-----------------------------------------------------------------------------

Google Tag Manager	gtm	Collects the dataLayer object when the Lytics tag loads. It also passes audience membership to the dataLayer.
--------------------	-----	---------------------------------------------------------------------------------------------------------------

Krux	krux	Adds the Krux pixel to the site.
------	------	----------------------------------

Lotame	lotame	Adds the Lotame pixel to the site.
--------	--------	------------------------------------

Optimizely	optimizely	Pushes a custom tag with the audience membership.
------------	------------	---------------------------------------------------

Pathfora	pathfora	Allows for the installation of Pathfora (Lytics Personalization SDK).
----------	----------	-----------------------------------------------------------------------

Qubit	qubit	Collects the universal_variable object when the Lytics tag loads.
-------	-------	-------------------------------------------------------------------

Lytics Audience Cookie	segments_cookie	Stores current audience membership for the user as a cookie.
------------------------	-----------------	--------------------------------------------------------------

Taboola	taboola	Adds the Taboola pixel to the site.
---------	---------	-------------------------------------

Tealium	tealium	Collects the utag_data object when the Lytics tag loads.
---------	---------	----------------------------------------------------------

JavaScript Tag TTL (3.0)		
--------------------------	--	--

The amount of time to cache the core JavaScript SDK in seconds. This should be used only during testing to ensure smooth performance for high-traffic production use.

Enable pageview event on first send (3.0)

Enables the automatic pageview event (e: pv) in the first send event of every page load in JStag v3.

Allow Pathfora

Check this box to allow the Lytics JavaScript tag to automatically add the Pathfora personalization library to your site personalization.

Allow Event Block

Check this box to allow the Lytics JavaScript tag to automatically add the Pathfora personalization library to your site personalization.

0: Turns event block off.

1: Turns event block on if there are one or more active Lytics Experiences.

2: Turns event block on permanently at all times.

Global Stylesheet URL (CSS)

This custom stylesheet will be loaded in support of Lytics-managed Experiences.

URL Allowlist for Orchestration (Experiences)

List of strings that, if matched, will result in Experiences being evaluated in place of legacy campaign variations.

Personalization (Pathfora SDK) Generalized Config

The hashed JSON configuration is to be loaded globally for all Lytics-managed experiences.

Custom Cookie Keys

A list of cookie names that should be automatically collected by the core Lytics SDK.

Custom Data Layer

A list of JavaScript variables that should be automatically collected by the core Lytics SDK.

Disable Automatic Initial Pageview (2.0 - Deprecated)

Disable the automatic pageview event in the JStag v2 if pageview events are being sent manually.

Jstag2 Segment Cookie Duration (2.0 - Deprecated)

The number of minutes the ly_segs cookie that contains an individual's current audience memberships should persist.

Updated about 1 year ago

Personalization API

Configuration options for the Lytics personalization API and related features.

Suggest Edits

The following configuration options are available within the account settings Lytics API section.

Allow Public Personalization API?

Check the box to allow the user profile of anonymous users to be exposed to users for personalization. It will only in

Allow Event Collection from Bots

Enable event collection for visitors who have been identified as bots. Bot identification is based on a set of match att

Collect entire user agent string?

Should the full user-agent be collected as part of all web events?

Allowlist domains public API

A list of domains to allow the Personalization API to be called from. If blank, all domains will be allowed.

Allowlist fields for public API

List of fields to allow for surface via the Personalization API.

Entity Timeout (ms)

Max time entity-public API waits before cutting off and returning what it has.

Updated about 1 year ago

Content Services

Configuration options for Lytics' suite of content services.

Suggest Edits

The following configuration options are available within the account settings Content section.

Extract allowed topics

Enable extraction of allowed content topics from the body of the document.

Content topic blocklist

A list of topics to be hidden from the overall content taxonomy. For example, a dinner restaurant may not care about other affinities, as pictured above.

Content topic allowlist

A list of topics that must be included in the topic graph and candidates for content affinity. For example, building on ensure that topics they care about are absolutely present.

Content Allowed Query Parameters

A list of query parameters that should be retained during URL normalization such as page id or product sku.

Content domains blocklist

Domains that should not be classified even though there may be events collected from them. Note: to properly filter "m" or "sub.example.com"

Content blocked pages

Block any URL with an exact match to an item in this list (including the domain, not including the protocol. i.e. www.e

Content paths blocklist

Prevent classification of any page with a substring match of the path. For instance, /contact would prevent classification

Content boosted attributes

Content list of IDs/classes to boost during body extraction.

Content created since date

Only include content in the index if the created date is after the specified date.

Custom content properties delimiter

The delimiter to use when parsing custom content topics on HTML meta tags.

Content custom properties

List of meta tags to include as custom topics.

Observe robots.txt in content enrichment

Observe robots.txt and meta directives:

- robotstxt - Observe only robots.txt directives.
- meta - Observe only directives in meta tags.
- none - Do not observe any directives

Content since date

Only include content in the index if the enriched date is after the specified date.

Content domains allowlist

Perform content filtering based on exact matches of domains in a URL. Any entries should include relevant subdomains

Content paths allowlist

Perform content filtering based on partial matches of any URL component.

Supported Content Languages

List of languages to permit during the content enrichment process. If empty, then only English content will be processed

Updated about 1 year ago

Security

Configuration options for account security and authentication.

Suggest Edits

The following configuration options are available within the account settings Security section.

Logon session days

The number of days a user session should be valid before forcing a re-authentication.

Logon session timeout minutes

The number of minutes between activities that should elapse before forcing a re-authentication.

Enforce password complexity

Ensure all users leverage passwords with a high level of complexity.

Enforce password history

Prevent the re-use of a previous password for each user.

Enforce password bad count

Lock the user's account if there are too many concurrent failed password/login attempts.

Password max age

The maximum age in days that a password can be before it must be changed. A value of 0 disables the max-age restriction.

Enforce two-factor auth.

Ensure all account users leverage two-factor authentication.

Two-factor authentication is a technique that helps to make your account more secure. It does this by adding a second factor to your login process.

Single-factor authentication uses your email address and password to authenticate your Lytics session.

The second factor comes from the Authy app using an Authy SoftToken, a secret token that changes every 20 seconds.

What is an Authy SoftToken

An Authy SoftToken is a secret token that is broadcast to the Authy app every 20 seconds. This unique token serves as the second factor for authentication.

Authy is available to download for free as a desktop and mobile app: [Download Authy](#).

Using Two-Factor Authentication

The only difference between two-factor authentication and single-factor authentication is an extra step during login.

Authy SoftToken.

Using Two-Factor Authentication also requires every user to provide their phone number. Logging in with two-factor authentication requires a unique token from the Authy app.

Updated about 1 year ago

AI & Modeling Controls

Configuration options for the Data Science and AI related features.

Suggest Edits

The following configuration options are available within the account settings Data Science section.

EntQL Fields

List of fields to evaluate via EntQL in the merger, before indexing the entity

Enable Decisioning Triggers

When enabled, any Experiences configured to leverage decisioning will be automatically triggered.
Streams to Allow for Scoring

Add any custom streams that contain behavioral signals that should contribute to the overall set of scores.
Enable Decisioning Scores

Each profile will be enriched with a score representing their overall need for a message (needs_message) and the time to next message (needs_time). The needs message is a numeric score representing the distance to or past the predicted event. A higher score signals a higher need for a message.
Turn On Priority Only for Decisioning

Automated Experience decisions will only leverage manual priority rather than full AI capabilities when enabled.
Updated about 1 year ago

Schema Controls
Configuration options for the profile and/or schema related features.

Suggest Edits
The following configuration options are available within the account settings Schema section.

Private fields

All fields containing PII/Private data should not be exposed to users unless their assigned role(s) allow viewing such information.
Promoted Fields

List of fields outside the defaults that should be leveraged in data science modeling and reporting.
Hide Private Fields

Hide PII/Private fields from audience exports and scans initiated by users who do not have access to private fields.
Updated about 1 year ago

Data Policies
Welcome to the Data Policies section, where we will explore the important policies and controls that govern data management. We will cover the following topics:

Suggest Edits
Welcome to the Data Policies section, where we will explore the important policies and controls that govern data management. We will cover the following topics:

Platform Limits

Privacy and Data Protection

Impact of Browser Tracking Changes

Updated over 1 year ago

Platform Limits

Suggest Edits

Platform Limits

The following limits exist to ensure the optimum performance of the Lytics platform and your connected sites, data sources, and integrations. All accounts unless stated otherwise. If you have questions, please get in touch with your Lytics Account Manager.

Events

An event is any action or activity a user performs, any update to a profile, or any export out of the subscription service.

Resource Limit Description

Max input size 4 KB Byte size of a single input record.

Max size in batch 16 KB Byte size of a single input record within a batch.

Max size total batch 1 GB Byte size of all records within a batch.

Max event ingress rate Quota based on pricing Events collected per hour.

Max output size 1 MB Byte size of a single output record, also known as a trigger event.

Max trigger event egress rate Quota based on pricing Events triggered per hour.

User Profiles

A User Profile is the living record of an individual that interacts with your brand. Lytics user profiles provide a view of

Known profiles have at least one known identifier such as an email or CRM ID.

Anonymous profiles have only anonymous identifiers such as a cookie ID.

Resource Limit Description

Max size of a user profile 1 MB Byte size of a single user profile.

Max size of a profile identifier 256 bytes Unique identifiers are used to merge data fragments together into a single

Max number of user attribute values on a profile 1,000 Values include individual items in complex field types such as
_visited, only 900 values remain for the total count.

Max aggregate size of all user profiles Quota based on pricing Total size of all user profiles.

Max number of custom mapped fields Quota based on pricing Custom mapping of fields through Queries & LQL.

Max stitched Identifiers 50 „Stitched... identifiers can be used to look up a user such as email or user ID.

Audiences

Resource Limit Description

Max number of audiences 500 Total audiences per account.

Max number of conditions (custom rules) per audience 1,000 This limit includes parent and nested conditions.

Max audience re-evaluation rate Quota based on pricing Max number of profiles Lytics will reevaluate per hour for

Max length of audience name displayed 64 characters This limit is for displaying audience names in the Lytics UI.

Browsers and Cookies

The following limits apply to client-side integrations, which are implemented in browser-facing code through the Lyti

Resource Limit Description

Max number of audiences stored as a cookie 5 MB Browsers limit the max size of the sum of all cookies. This limit c
local storage instead of a cookie.

Max cookie size per domain 4,096 bytes This limit includes both Lytics-generated cookie names and the values of a

Maximum size of collect payload sent via URL 2,000 bytes Try to keep payload sizes small enough to fit in a URL usi
e sent via iframe transport, which can result in poor performance.

Integrations

The following limits apply to server-side integrations, which allow Lytics to communicate directly with third-party ser

Resource Limit Description

Max number of active import integrations 100 Import integrations ingest data such as user profiles, activity data, and

Max number of active export integrations („subscriptions...") 200 Export integrations send user profile data from Lyti
data or metrics can also be exported for monitoring.

Max integrations creations per hour 10 This limit applies to all server-side integrations, including imports, exports,

Max lifetime of completed jobs 90 days This limit applies to all jobs in a completed status, and defines how long Lyti

Max lifetime of paused / failed jobs 90 days This limit applies to all jobs in a paused or failed status, and defines ho
ng them purgeable.

Content

The Lytics Content Affinity Engine uses data science and Natural Language Processing to analyze content, extract top

Resource Limit Description

Max number of URLs enriched 20,000 URLs per month --

Max number of topics displayed 500 This limit is for how many topics are shown on the user profile.

Max number of topics per user 50 Lytics retains affinities for up to 50 topics per user, representing the content the
Custom Modeling

Lytics enables you to build custom Lookalike Models as a proprietary service that uses your first-party data to evalua

Resource Limit Description

Lookalike Model count 20 This limit applies to activated Lookalike Models.

Max model audience size 20M users per audience This limit applies to the source or target segment used in a Look
Updated over 1 year ago

Privacy and Data Protection

Suggest Edits

Privacy and Data Protection

As a service provider and data processor, Lytics assists its customers in enhancing security and meeting privacy and
ral Data Protection Regulation (GDPR) and California Consumer Privacy Act of 2018 (CCPA).

Listed below are the compliance-enabling functionalities Lytics provides. As used below, personally identifiable information include „personal data... as defined in the GDPR. Please consult your company's legal counsel or privacy professional to ensure you must comply with.

Lytics recognizes that its customers are the data controllers of the PII, which Lytics processes on their behalf. Each Lytics customer is responsible for determining the destinations to use with Lytics and the types and content of PII shared between its sources and destinations. Lytics does not collect or process any third-party data except via customer-directed integrations.

Access Control

Restrict access to personal information by role.

Role Based Access Controls (RBAC):

Account Admins can easily add and remove account users. Lytics has various defined user roles with respective permissions.

Single Sign On (SSO):

Add SSO to your user login process to enhance security.

Multi-factor authentication (MFA):

Add MFA to your user login process to enhance security.

Restrict access to PII:

You can indicate any user fields that contain PII via the private fields account setting. These fields will be hidden for all roles. You should verify with Lytics Support that the field hiding in the segment scan is also enabled for your account.

Data Mapping

Map personal information processed by Lytics, including sources and destinations.

Audit your data schema:

Use the Schema Audit feature to see what user fields are being populated, the data contained, the source(s), and if the data is shared.

Determine third-party data sources:

You can see the third-party data sources from which you send data to Lytics using the Lytics UI by navigating to Data > Sources. You will see your other integrations in the list of stream names using the drop-down menu to filter by the type of data source. You will see the Lytics received data.

Determine third-party data destinations:

You can view the activity history for a data destination using the Lytics UI by navigating to Data > Integrations. Click on the integration you are interested in. If the integration is currently running, you will automatically be taken to the overview page that shows a list of running imports and exports.

Notice and Consent

Manage user consent and preference data.

Obtaining Customer Consent

Lytics customers are responsible for obtaining consent for collecting and transferring PII to Lytics for processing. You can use various methods to obtain consent from your customers' online behavior. One consent mechanism is to implement a custom tag and trigger in your Google Tag Manager. You can also use the Lytics consent with Google Tag Manager.

For sites not using Google Tag Manager, customer consent on the web can be managed in several ways. Consent triggers can be used to track user behavior on your site. Another alternative is to use a cookie-consent solution, many of which exist. Github has documented a free and open source library to consume triggers from any of these solutions to manage consent for your customers.

Recording Proof of Consent:

Schema fields may be established for the purpose of storing customer consent.

Privacy Policy Notice:

When you use a Lytics modal to collect user information, you should include a link to your organization's privacy policy. This link can be added to any modal created using the Experience Editor.

Age Gating:

If you have collected accurate age data, you can build audiences that target or exclude certain ages.

Responding to Consumer Requests

Respond to the data subject (consumer) requests in compliance with regional and state privacy and data protection

Personal Data Access:

Using the find a user feature, enter the identifying details provided by the consumer to locate their profile. The profile will show any data on this user.

Personal Data Correction:

If user profile data requires correction, you must send the corrected data to Lytics, which will be remapped to convert

Determining Categories of Personal Data Collected:

You can use the Lytics UI to obtain information about the categories and specific pieces of PII collected on a consumer. You can view the fields of populated data and determine the appropriate consumer PII categories to disclose to a requestor.

Personal Data Portability:

We support the export of profile information via the Lytics UI or APIs. An individual's profile data from Lytics will be exported in a common, machine-readable file format.

Personal Data Deactivation/Suppression:

You can establish audiences to enforce consumer suppression and „do not market... choices and prioritize those choices. Audiences can be exported from Lytics to your downstream tools or "data destinations."

Personal Data Deletion:

We provide a method for deleting users via the Lytics UI. Our API may also be used for this purpose. This will send a request for the customer identifier provided.

SOC 2 Audits

An independent auditor has examined Lytics platform controls and confirmed they are in accordance with the Service Organization Control (SOC) 2 Security, Availability, and Confidentiality. You can learn more about our SOC 2 Type II examination in this blog post.

Lytics will continue to engage independent auditors to conduct SOC 2 Type II audits regularly and make our audit reports available. In addition, we retain independent security firms to conduct regular penetration tests and vulnerability scans on our systems. Google, also submits to regular, multiple independent audits, including SOC 2 Type II audits.

Safeguards and Transfers

Lytics Data Protection Safeguards:

Lytics and its data hosting partner, Google, have implemented numerous safeguards to protect the PII that Lytics processes. These safeguards are reviewed on an annual basis. For more information regarding these safeguards, please ask your Lytics account manager.

Transfers of Personal Data from EU:

Lytics participates in the EU-US and Swiss-US Privacy Shield Frameworks regarding collecting, using, and retaining personal data. We have certified with the U.S. Department of Commerce that we adhere to the Privacy Shield Principles. Please let us know if you wish to enter into the EU Standard Contractual Clauses regarding data transfer from the EU to the U.S.

Updated over 1 year ago

Impact of Browser Tracking Changes

Suggest Edits

Impact of Browser Tracking Changes

Announcements about browsers changing their cookie tracking policies have created tremendous concern and confusion. These changes represent dramatic shifts in how users and their activity are tracked across the web.

Browsers are removing third-party cookies and altering the default behavior of first-party cookies, which makes first-party cookies more important. The following explains how current and upcoming browser changes will impact your web tracking involving Lytics.

Overall impact

The blocking of third-party cookies, whether by browser default or user choice, will significantly impact the effectiveness of targeting, and measurement efforts dependent on the use of third-party cookies. In addition, Apple's ITP 2.2 impacts certain

For Safari users, cookies get cleared after seven days if there is no activity. So if a user visits a site once and returns every day for eight days, their cookie will remain intact.

This can skew analytics, impact personalization efforts for anonymous users, and affect conversion attribution. But despite the fact that mobile web traffic in the US for non-mobile and worldwide browser usage, Safari's market share is around 20 percent as of 2018.

More importantly, these browser changes continue the trend of giving individuals more control over what brands/enterprises can track. This points to the need for brands to maximize relevant content or offerings for a site visitor in real-time, which can lead to a more direct first-party relationship. These browser changes reflect the change from a model that exploits third-party data outside a brand's direct relationship with the consumer for mutual benefit.

Impact on Lytics

The good news is Lytics was designed to help companies establish first-party relationships with their customers and gives you the tools to collect and leverage first-party data in various ways, such as: acquiring new customers, serving ads on-site, and personalizing content to users based on their interests.

Onto the technical details. To understand how these browser changes will impact the Lytics JavaScript tag (aka Tracking Code), let's look at how Lytics builds profiles for your anonymous visitors.

It all begins with Lytics' unique client-side identifier, which is referred to as the UID or `_uid`. When a user visits your website, Lytics generates a unique value for this UID and stores it as a browser cookie. This UID is then appended to any behavioral data collected during the visit. On subsequent visits, this UID persists since it has been stored as a cookie.

The impact of cookie changes on this process will vary by browser. In the case of Safari, for example, it means that if the cookie containing the identifier is deleted, the identifier will be lost. As such, it will become imperative to double down on your identity resolution strategies and combine anonymous cookies with other known identifiers that can persist, such as user IDs upon login.

In addition, some customers leverage the `loadid` parameter as part of their tag configuration. Historically, this has been used to resolve identities. This `loadid` method relies on third-party cookies, which are impacted heavily by these cookie changes. Like the other strategies mentioned above, these strategies become more important.

To date, efforts have been made to rely less on browser cookies, given their uncertain nature. Version 3 of the Lytics JavaScript tag is designed to allow Lytics to respond to these types of changes more efficiently. At the end of the day, however, restricting or removing cookies is a trade-off for marketers who want to track, use and monitor the web. Focusing on safe, secure, and transparent resolution strategies is at the core of the Lytics approach. We will continue to evolve these strategies in the future.

Q. What cookies are issued in connection with the Lytics tag?

By default, Lytics sets a cookie `seerid` that is used to identify a user. This is then surfaced in a user's profile as `_uid`. Other cookies are used for web-based identity resolution.

Q. Are Lytics' cookies first-party cookies or third-party cookies?

When issued by the client website domain on which they run, Lytics' cookies are treated by browsers as first-party cookies. For companies that have established separate top-level domains for two or more of their brands, many have leveraged the `loadid` parameter as a third-party cookie. This allows for a hands-off approach to resolve identities across domains as described above.

Stale cookie removal feature

Lytics has enabled a „cookie culling... feature that prevents „stale... (old) cookies from being used as identifiers to stitch together user profiles. As mentioned above, this ensures only valid identifiers will be used to build your Lytics user profiles, which helps keep your data accurate.

For example, if a Safari user visits your site once and doesn't return within two months, Lytics can remove this cookie. The reason to keep this cookie as an identifier because it can't be used for personalization. Lytics recommends opting for a more direct first-party relationship customized according to your needs. Don't hesitate to contact your Lytics Account Manager if you are interested in exploring this feature.

Once Lytics removes cookies after your specified time frame, not only the „stale... cookies are removed, but any other data associated with the stitching process. As a result, any associated data with the old cookies will not be included on user profiles unless you have a direct first-party relationship (e.g. email or user ID). However, the data lost from turning on cookie culling can be found again if the feature is turned off.

iated with the missing data is no longer skipped.

Cookie changes per browser

The following information has been updated as of January 31, 2020.

Cookies are used for persistent login, preference storage, and tracking across websites. Google, Apple, and other browsers have implemented tracking of users across websites. This functionality is promoted as privacy-enhancing by giving a user greater control over third-party cookies, but Apple's technology also focuses on certain first-party cookies.

Apple

Apple introduced its Intelligent Tracking Protection (ITP) functionality in 2017 to block the use of cookies deployed on websites. To track Safari ITP changes on Apple's WebKit website. Generally, Safari has set a 7-day expiration period on first-party cookies with link decoration. This means for users that don't revisit a site within seven days to extend their expiration, the cookies expire the next time they visit the site.

Firefox

Like Safari, Firefox blocks third-party cookies by default through Enhanced Tracking Protection.

Google

In 2019 Google announced it was changing its approach to third-party cookies by not blocking them by default and providing users options to block the cookies they want to block and to delete cookies after a session (fingerprinting out as well).

In mid-January of 2020, Google announced its intent to phase out support for third-party cookies in Chrome by 2022.

Updated over 1 year ago

Compliance

Suggest Edits

Downloading a Customer Profile

GDPR requires the right of access and data portability. You can use the Lytics platform to download a JSON file of a customer's data as a customer profile.

Log into your Lytics account.

Click Audience > Find a user.

Select the field you would like to search on from the drop-down menu. By default, Email is selected.

In the Search box, please enter the email address (or selected search term) of the customer who has requested their data.

Select the desired customer from the results list.

Click Download profile.

You will receive a success message and profile.json will begin to download.

Deleting a Customer Profile

The GDPR and CCPA requirements grant the user in scope to request the removal of their personal data without undue burden from Lytics via the Lytics UI. In addition, you can use the Lytics API to delete a customer profile. A successful deletion removes the status of a customer in the Lytics UI until the profile has been removed from Lytics. The removal is handled in batches.

Log into your Lytics account.

Click Audience > Find a user.

Select the field you would like to search from the drop-down menu. By default, Email is selected.

In the Search box, enter the customer requesting removal's email address (or selected search term).

Select the desired customer from the results list.

Click Delete user.

A confirmation message will pop up; click Delete this user to confirm.

There will be a Request ID in this message to track or audit the success of the GDPR deletion request. The customer will be marked for deletion and removed from Lytics within the GDPR-required timeframe.

Updated 11 months ago

Authorizations

Suggest Edits

What are Authorizations?

Channel tool authorizations refer to the permissions granted to various integrations that allow our platform to access data. These integrations include email service providers, social media platforms, and advertising networks.

Once permission has been granted, Lytics surfaces "Authorizations," which allow you to manage and maintain the connections. You can use them to create and manage auths to ensure the long-term health of your data pipeline.

Creating Authorizations

Authorizations are currently accessed from your account navigation menu by selecting Account > Security > Authorizations.

Create new

Within the Authorizations section of the interface, you will see a list of existing Authorizations and a button to "+ Create Authorization". Clicking the button will take you to the authorization wizard, where you will:

Choose the provider.

Select the desired authorization method.

Add a name (label), description, and complete the configuration options.

You will need authorization for most jobs when going through the job creation wizard in Lytics. At the "Choose Authorization" step, you will be directed to the authorizations wizard in a new tab. You will be prompted to choose the method and configure the authorization.

Authorization Methods

Note that some integration providers only have one authorization method, but others offer multiple methods such as real-time vs. bulk audience exports. If you are unsure which method to use, you can find more information in the documentation.

Google auth methods

For example, if you are looking to import event data from Google Cloud Pub/Sub into Lytics, there is only one method available: the Import Data (Cloud Pub/Sub) job in Lytics.

Managing Authorizations

The Authorization Summary provides essential details about each authorization to help ensure an integration is appropriately configured. At the top of the summary page, you will see the following information:

Provider: The third-party tool that this authorization connects to.

Method: Indicates how the authorization was made, such as API keys, OAuth, personal or business users, etc.

Health: Indicates whether the Authorization is valid and active.

Owner: Lytics user who created the authorization.

Created At: The date the authorization was initially created.

Last Updated: The date the authorization was most recently edited.

What determines authorization health?

Lytics checks the status of authorizations daily to help you proactively resolve authorization issues. Our API will return a health status for your provider:

Healthy: The authorization is valid and active.

Unhealthy: The authorization is invalid and inactive.

Unknown: The authorization status is unknown.

If your authorization is marked "Unhealthy," Lytics cannot verify that it has the requisite access and permissions. Visit the external tool to verify the authorization credentials you entered have the necessary permissions in the external tool.

If adjusting permissions in the external tool does not apply or is not an option, you may need to create a new authorization to replace the unhealthy authorization.

Edit or Delete Authorizations

You can edit or delete authorizations directly in the Lytics UI using the buttons at the top right of each authorization card. You can edit the description and revise the configuration options, but the configuration options cannot be changed. If you need a different configuration, you must create a new authorization and delete the previous one.

f

You can only delete an authorization if no active jobs use it. If there are associated jobs with a running, sleeping, or pending status, you cannot delete the authorization. Likewise, you will need to delete any imported Experiences that are actively using an authorization.

Activity Metrics

Once metrics are available for each authorization, this chart will become populated with the number of requests made by the authorization. You can click on the chart to view the metrics hourly, daily, weekly, or monthly, depending on the available data.

Authorization metrics are currently only available for select providers (including Facebook, Iterable, Salesforce, SendGrid, and others). For a full list of providers, see the [Authorization Metrics](#) page.

Associated Jobs

This table provides a helpful organization of all active jobs using the current authorization. Jobs can be sorted by name, status, or creation date. For more details on the job summary page.

Authorizations Dashboard

Authorizing the connection between Lytics and other providers allows you to create jobs to import, export, and enrich data.

Navigate to Account > Security > Authorizations to find your Authorizations Dashboard, which gives an overview of all your authorizations.

All authorizations are displayed in a table and are sortable by the following fields:

Label: The name of the authorization as defined by your input for the "label" field.

Description: Optional text field to differentiate authorizations. This is particularly useful when you have multiple Authorizations with the same label.

Provider: The third-party tool that the authorization is connecting to Lytics.

Status: Indicates whether the authorization is valid and active.

Created By: Lytics user who created the authorization.

Last Modified: Date the authorization was last edited.

Click on a specific authorization to view its summary page, which will provide more details and allow you to edit or delete the authorization.

Updated 1 day ago

Access Tokens

Suggest Edits

Access Tokens

Creating and managing your access tokens gives you fine-grained control over how your Lytics account and data is accessed by third-party tools. You can manage access tokens in the menu in Vault. Note that just like all other account settings, you must have administrator privileges to view and manage access tokens.

Viewing existing Access tokens

If this is your first time creating an access token, you will see a message prompting you to create a new token.

Once you have created at least one access token, this page will display your tokens, their expiration dates, and the user who created them.

You can also click on a specific token to see the list of roles the token has been granted, the lifetime of the token, the user who created it, and the date it was last used.

Creating a new API Token

To create your first token, click the Create New button and fill out the following fields on the modal that appears:

Field Description

Name Token name.

Description Though optional, it's helpful to add a description so each token's purpose is clear.

Expiration Token expiration length. Current options include 90 days, 30 days, seven days, and No expiration, which

Roles Each token can be tailored to have the precise access needed and no more.

Click Generate token to create your token. You will be prompted with a one-time dialog window that contains the access token again, so please copy and paste it somewhere secure.

Deleting an existing API token

When an API token is no longer needed, or you otherwise wish to remove one or more tokens, click on the token from the list and click the delete icon. Confirm the deletion.

f

This operation is not reversible and you will be prompted twice before each token is deleted. Anyone or anything de

Updated over 1 year ago

What is Cloud Connect?

Suggest Edits

Introduction

Cloud Connect allows you to run complex SQL queries directly against your data warehouse and translate the results into

Cloud Connect in Action

,
Cloud Connect in Action

Check out the solution architecture developed with Google which leverages BigQuery and Analytics Hub in addition to Cloud Connect for activation and activation.

Getting Started

Accessing Cloud Connect

Cloud Connect is available to all Lytics customers. If you don't currently have access via the product switcher at the top of the page, reach out to your account management team.

Common Use-cases

Cloud Connect unlocks many use cases enabling marketers to segment their customers based on any attributes stored in your data warehouse.

Time Window: All users who did not log in last month.

Joins (B2B): All users associated with accounts without feature "x."

Lifetime Value (LTV) or Rollup: All users with a premium subscription have purchased at least two products.

,
Need a warehouse?

Don't currently have access to a data warehouse? Lytics has you covered with Lytics Warehouse. We provide a simple way to connect your data to the Lytics CDP, a key part of Conductor. Simply reach out to account manager or support for access and information.

How it Works

Cloud Connect augments the rest of the Lytics CDP suite by providing a secure and flexible entry point into your data warehouse. It allows you to perform a number of complex data manipulations or aggregations and stream the results directly to materialized user profiles, activate across hundreds of marketing channels in just a few clicks.

Below you'll find an overview of the general architecture, which can be summarized into these key value drivers:

Direct access to existing customer data in your warehouse.

No need to rip and replace or replicate consumer data in order to activate.

Leverage standard SQL to extract only what is relevant to your business goals.

Maintain your warehouse as the "source of truth."
Requires very little to implement and begin generating ROI.
Cloud Connect diagram

Updated about 1 year ago

Connecting Warehouses

Suggest Edits

Access

Cloud Connect tabs are found under Data Pipeline > Cloud Connect. Connections configure the access to your data warehouse, connect audience membership and profile attributes.

Creating a Connection

Click + Create New Connection from the Connections Dashboard at the top right and complete the following steps.

Choose the provider.

Choose the Connection type.

Select an existing Authorization or create a new one by following the Authorization instructions.

Add a name (label), description, and complete the configuration options. These will vary slightly between providers.

Authorization & Security

The authorization selected for your Connection will control your Lytics account users' access to your data warehouse, by dataset or individual tables, maintaining your security and governance practices within your data warehouse.

Supported Data Warehouses

Cloud Connect currently supports a number of popular data warehouses:

Amazon Redshift

Databricks

Google BigQuery

Microsoft Azure SQL Database

Snowflake

Managing Connections

Once you have created a Connection, you can access a summary page showing how data from your data warehouse is being used. The following information is displayed:

Provider: Data warehouse that you are connecting with Lytics.

Authorization: Name of the authorization, such as „Cloud Connect JWT.... Note: Lytics users can access any data tables within the authorization.

Type: Indicates the type based on your provider.

Created By: Lytics user who created the Connection.

Created On: Date the Connection was initially created.

Last Updated: Date the Connection was most recently edited.

The rest of the Summary tab shows how many active and inactive data models are built using this Connection as a data source. The Activity chart displays how many rows are being queried, which can have cost implications based on your data warehouse.

Explore

The Explore tab provides a simple Schema Explorer to validate that the data shown is as you would expect to see in your data warehouse. For an individual table, but here you will see as many tables as the authorization has read access to.

Details

The Details section displays all the information about setting up your Connection, including the authorization and configuration options.

Logs

The Logs section records the history of events for this Connection, which are helpful to ensure your connection is working as expected. See the Logs section for more details.

Connection Events Description

Created First event indicating the Connection is active.

Updated Connection was updated by a Lytics user.

Deleted Connection was removed and any data models built on this Connection will no longer be updated.

Updated 3 months ago

Data Models & Queries

Suggest Edits

Cloud Connect Data Models

A Data Model is used by Cloud Connect to link or "connect" a user's external data warehouse to Lytics profiles. Each Data Model will also configure a join key which defines how the Data Model is joined to Lytics Profiles.

Once a Connection to your data warehouse has been created, Data Models can be created via the Data Models tab.

Creating a Data Model

Click + Create New from the Data Models dashboard and complete the following steps.

Select the Connection.

Write the SQL query.

Query Editor: Write and test standard SQL queries directly in Lytics. Alternatively, copy and paste queries you've tested elsewhere. When you click Test Query, Lytics will return 10 sample records.

Test the query and validate the results.

Configure the Data Model.

Name: Data Model name

Description: The Description of the Data Model.

Slug: The Data Model name that will be used in the membership and activation fields in the audience builder. It is important to be unique across all audiences in the tool. If no slug is selected, it will auto-populate using the model name.

Primary Key: Select the data warehouse ID that will be used to match against profiles in Lytics.

External Lytics Key: Select a Lytics user profile field that is a unique identifier to map to an identifier in your incoming data.

Activated Fields: Select the fields you want to bring into Lytics profiles for activation. Note that each data model can only activate a limited number of fields.

Sync Frequency: Select a time interval to run the query on - this is also the frequency that your Lytics profiles will be updated. Options range from daily to every 3 weeks.

Create new Lytics profile: Check this box if you want to create new profiles in Lytics if a profile with a matching ID does not exist. If unchecked, only users with matching IDs to the Data Model IDs will be updated.

Activate the Model. Until the Data Model is activated, its SQL query will not be run against your data warehouse and no data will be imported into Lytics.

When building multiple Cloud Connect Data Models with the same primary key, you must select the same External Lytics Key. Otherwise, you will receive an error when you try to save the new Data Model.

Example Use Case

Consider this scenario to demonstrate why you'd want to create a Cloud Connect Data Model driven audience instead of a standard Lytics audience. You want to run a holiday campaign that sends a promotion to any customer who purchased an e-bike in November or December and who are interested in particular bike brands. All of this purchase and product data already exists in your Google BigQuery database.

BQ-SQL-holiday-sample

Instead of directly importing all that purchase history data into Lytics, you can write a SQL query to find which customers purchased a product and then import that data into the Lytics Model Builder (as shown in the generate a query screenshot above), you'll create a new audience in the Lytics Model Builder.

Once the Data Model is activated, profiles will be updated with a membership field that can be used to create an audience. This field can then be used for more dynamic audiences.

SQL Translator

If you would like to skip having to write a SQL query, simply describe the Data Model of what you wish to fetch and upload to Lytics. Lytics makes queries to your database to ensure that the data is up to date and accurate when creating the Data Model.

Updated 20 days ago

Activating Data Models

Suggest Edits

Activating Data Models

Because the queries that power your Cloud Connect Data Models can have costs associated with your database usage, you should be aware of the costs associated with your database usage.

After you've created a new Cloud Connect Data Model, you'll notice that the Status is listed as Inactive:

In order to activate, click the Activate button on the Data Model page:

Once you click the Activate button, your query will complete its first run within an hour of the Next Sync On timestamp. During each sync, the SQL query will be run against your database, and your Lytics profiles will be updated with the new data. The Data Model will be active.

Once synced, the Total Records metric on the Data Model summary tab may take up to 2 hours to display. This number will update as more data is synced.

Data Model Field Creation

Upon activation of a Data Model, Lytics will automatically create one customer profile field which represents the membership field. If the Data Model was configured to include Activated Fields, below is a guide for how to locate these generated fields (on the Custom Fields tab)

Profile Field Structure of Short Description Structure of Slug

Data Model Membership Data Model Slug (membership) cc_Data Model Slug_membership

Data Model Activated Fields Data Model Slug - Query Field cc_Data Model Slug_Query Field

Deactivate Data Model

If you need to disable a Cloud Connect Data Model from syncing at any point, you can do so by navigating to the Data Model page.

Deactivating the Data Model will not remove its corresponding fields from schema or from Lytics profiles; however, it will stop syncing data. In order to completely remove the fields from schema/profiles, the Data Model must be deleted.

Total Records vs. Count of Users

Throughout the app, you may see different metrics for Cloud Connect Data Models.

Total Records within the Data Model Summary

On the summary page for your Cloud Connect Data Model, under the Total Records statistic (shown in the graph below), the number of records is based on the unique number of primary keys in the origin database. This may be equal to or less than the number of records you see in the audience builder when utilizing this Data Model.

Count of Users within the Audience Builder

When adding Cloud Connect Data Model membership or activated fields to audience logic in the Audience Builder, the count of users within the Audience Builder is the number of users in your Lytics account who met the qualifications for the audience.

on the Data Model. This number reflects the number of records that would be included if you were to run an export

Most common reasons for why this number may be smaller than the Total Records metric are the following:

If the "Create New Profiles" option isn't selected on the Data Model, the count of users in the Audience Builder will only include profiles that have been created.

Audience Builder will exclude any profiles which are marked as being Unhealthy. Learn more about unhealthy profiles in the [Audience Builder - profiles](#) article.

Exporting Data Models

Once activating a Data Model, its associated fields will be stored as user fields on the customer profile (to see the names of the Data Model, you will first need to create an audience in the Audience Builder. You can create an audience using the data from the Data Model, or you can create an audience using the data from the Data Model and the most out of your customer data.

For example, you could build a Cloud Connect audience using account-level data for a B2B use case and add one of the user engagement level.

To learn how to export audiences, visit our [Activating Audiences](#) documentation here.

Updated 20 days ago

Cloud Connect Troubleshooting and FAQs

Suggest Edits

Syncs

What is a „sync... in the context of Cloud Connect?

A sync refers to the process of running SQL queries configured on Data Models against data warehouses and then updating the data in the previous sync.

Will my SQL query only be run once per sync?

Most of the time, yes. However, if an interruption occurs during the sync (either due to a processing or network error), the sync will be interrupted and the query will not be run. If that occurs, we'll need to rerun the query.

Why are the sync times slightly different than what I'd expect?

On the Data Model Details page, we display two timestamps related to the sync for your Data Model: „Last Sync on... and „Next Sync On... The „Last Sync on... timestamp shows the time when your Data Model was last synced. However, the timestamp shown for „Next Sync On... will vary slightly from the „Last Sync on... timestamp.

In an effort to reduce the number of syncs needed, we set the „Next Sync On... timestamp in a way where more models are ready to sync - if „Next Sync on... is in the past, we sync the model.

Ultimately, you should think of your selected sync frequency as the number of times per day your model will sync (e.g., once per day). The „Next Sync on... timestamp gives an estimate as to when the next sync will occur rather than the exact time.

Data Models

Why can't I create a Data Model that returns no rows?

We use the columns returned by the SQL query to set up our internal schema. If no data is returned by the query, we cannot create a schema. Note that once the schema is set up, your Data Model will continue to sync regardless of how many rows are returned.

Why can't I create/update a Data Model when my schema has unpublished changes?

Creating a Data Model modifies your Conductor schema - at a minimum, a true/false „membership... field will be added to the schema. When updating a Data Model, one field is added for each activated field that's selected. In order for these fields to be added, your Conductor schema must be published.

When updating a Data Model, we'll attempt to update your Conductor schema if any activated fields are being changed. If your schema has unpublished changes, we'll reject the update.

Can I select any field as the primary key?

The UI allows for selecting any column that's returned by your SQL query as your primary key. However, it's important to note that if there are duplicate values, we cannot guarantee that the values for your activated fields will be correct.

You should choose a field whose values will match to the values of an existing „by... field in your Lytics schema.

Can I select any field as the Lytics key?

No; only „by... fields can be selected. Further, once a Data Model has been created for a certain primary key, all subsequent reports will use the same Lytics key.

Sync Failures Related To Collation

Cloud Connect requires that the column you use as the primary key of your data model queries be UTF-8 collated. If your data is not UTF-8 collated, the sync will fail. Most database providers allow columns to be converted to a particular collation within a query. If you are using a database provider, consult the documentation of your database provider to ensure a compatible collation:

Snowflake

BigQuery

Azure

RedShift

Updated 10 months ago

Managing Reports

Suggest Edits

Reports come fully equipped to manage your Reports across your team.

Editing and Deleting Reports

Users can quickly Edit and Delete reports at the top of the Report page. The Edit option allows you to rename the Report.

Sharing Your Reports

Creating Audience Reports in Lytics can be a collaborative process. By default, Reports within Lytics are shared across all users with the Reporting role. The Reporting role can be disabled for specific users.

In addition to the global role, a report creator can further restrict access to their reports by setting the report as private. The screenshot below outlines the simple setup process to fine-tune collaboration and access to individual reports.

Change Logs

Lytics records a log of all changes made to a Report in the „Logs... tab. This record lets users keep track of who updated a Report.

Downloading Reports

Each component in Lytics provides the option to download the associated data in a CSV format. The Download icon is located in the top right corner of each component.

Updated over 1 year ago

Dashboard Report

How to interpret and configure your Dashboard Report

Suggest Edits

Once data is flowing into your account, Lytics will automatically generate a Dashboard Report on the homepage of the account. The Dashboard Report is a customizable, the out-of-the-box Report consists of 4 Report Components that incorporate some of Lytics' Behavioral Audiences.

Size Component of the All, Lytics New, and Lytics Highly Engaged audiences

Dataflow Component of the 5 most "used" audiences (ie the audiences that appear in the most Jobs).

Composition Component of Lytics' Content Affinity

Overlap Component of the All, Lytics New and Lytics Highly Engaged audiences.

To modify or edit the Dashboard Report, simply click on the Edit Dashboard button located on the top-right corner of the Dashboard Report page, where you can add or delete Components, and edit existing Components.

Updated over 1 year ago

Evaluating Lookalike Models

Suggest Edits

Lookalike Model Summary

The Lookalike Model summary view consists of three tabs each containing valuable information about your model. The Accuracy tab displays the accuracy, reach, and its usage. The Configuration tab displays the options selected to build the model, and the Diagnostics tab displays diagnostic or debugging.

Model Summary

The model summary dashboard surfaces the key metrics of a model. Here you can see a model's accuracy and reach, which contribute to the model, what audiences are using the model, and any diagnostic and troubleshooting messages.

lookalike-model-summary

What determines model health?

At the top of the summary page, you will see a tag indicating whether your model is healthy or unhealthy. Lytics determines model health based on the model's predictions -- predictions from unhealthy models should not be trusted. Models are considered unhealthy if the model is overfit (with an R2 value greater than 0.975).

Accuracy

Accuracy describes the precision of a Lookalike Model's predictions -- that is, how closely a model's predictions match the actual outcomes. Accuracy can be used to narrowly target the strongest lookalike candidates, which are better applied to later stages of your targeting funnel.

lookalike-model-summary-accuracy

Reach

Reach describes the relative size of a Lookalike Model's addressable audience. Higher reach models can broadly target a large audience, while lower reach models can more narrowly target a specific audience.

lookalike-model-summary-reach

Model predictions

The model prediction section displays the size of your source and target audiences and charts the predictions for the target audience. The Y axis represents relative density. The shape of the graph is most important, which indicates the amount of overlap between the source and target audiences. For examples of how different trade-offs between accuracy and reach will look in the model predictions graph.

Model feature importance

Model feature importance indicates features among users likely to convert from the source to the target audience. The features are ranked according to the model calculation ordered from most to least important. Identifying these features may give you a better understanding of the target audience. You may even opt to include some of these features when building an audience with this model.

lookalike-model-summary-feature-importance

Model usage

Model usage displays a list of audiences built that are utilizing this model for targeting as well as a button to create a new audience. Audiences built from a healthy Lookalike Model perform very well, providing higher conversion rates. Clicking the "Create Audience" button will take you to the Audience Builder, pre-populated with a rule utilizing the prediction score user field for your model. More details on building audiences are available in the Audience Builder documentation.

Diagnostics & Troubleshooting

The Diagnostics and Troubleshooting section provides messages pertaining to the performance and status of your Lookalike Model. Warnings and error messages may indicate why your model is unhealthy or provide suggestions on how to resolve the issue.

lookalike-model-summary-diagnostics-troubleshooting

To resolve these messages, try changing the source or target audiences, or any of the other configurations as suggested.

Model Configuration

The Configuration tab displays the settings used when creating the model.

lookalike-model-summary-configuration

Model Diagnostics

The Diagnostics tab provides an advanced view into a Lookalike Model's performance and can help provide assurance that the model is healthy. This information is helpful for technical users looking to gain extra insight into the data science behind this Lookalike Model.

lookalike-model-summary-diagnostics

R2: A measure of how successfully a model is able to predict its desired outcome. It can be interpreted as the percentage of variance in the target variable that is explained by the model's predictions.

perfect model would have an R2 value of 1, and an ineffective model would have a value of zero.

Mean Square Error: A measure of how closely a model's predictions matches actual outcomes. The scale of MSE has no units. Lower MSE would indicate a better fit.

False Positive Rate: The percentage of „false positives... in a model -- that is, percentage of users in the source audience who are incorrectly classified as being in the target audience. While the false positive rate shouldn't be too high, a large false positive rate tends to yield a model with higher reach.

False Negative Rate: The percentage of „false negatives... in a model -- that is, percentage of users in the target audience who are incorrectly classified as being in the source audience. A lower false positive rate means that a model succeeds in correctly identifying the latent characteristics of the target audience.

Overall Error Rate: The total percentage of misclassifications that occurred during model training.

Accuracy: A measure derived from a model's R2 value, rounded to the nearest decile.

Reach: A measure derived from a model's specificity value, typically ranging from 0 to 10. A larger value of reach indicates a model that is more similar to the target audience.

AUC: A measure of the area under a receiver operating characteristic curve (or „Area Under the Curve...). ROC curves are used to evaluate a model.

Decision Threshold: The „optimal... model prediction value to use as a cutoff when constructing an audience definition. A model with a threshold of 0.5 would have the same „penalty... as a false negative.

Updated over 1 year ago

Creating Predictive Audiences

Suggest Edits

Once a Lookalike Model is built and users are scored (make sure the Model Training Only option is unchecked or on), you can create Predictive Audiences with different prediction decision thresholds for the model or percentiles based on the predicted score.

From the Lookalike Models list view, click the model you'd like to use to build a Predictive Audience. Then find the Model Predictions field.

create-predictive-audience

This opens the Audience Builder. All Lookalike Models are keys under the user field Lookalike Model Predictions. The predicted score is on a scale of 0-1. Users closer to 0 represent a low likelihood to look like users in the target audience and vice versa for users closer to 1.

lookalike-model-predictions

By default, this rule is populated with the model's Decision Threshold, computed as the equilibrium prediction score (the score at which the number of false positives equals the number of false negatives). However you can adjust this threshold as you like or add additional rules before saving the audience. See Improving Audience Quality. Any audiences built using the audience prediction score for your model will display in the model usage module.

Using Lookalike Model Percentiles

Another option to build a Predictive Audience is by using the Lookalike Model Percentiles field. Similar to the Lookalike Model Predictions field, the Lookalike Model Percentiles field.

lookalike-model-percentiles

The percentile for a model represents the value at which a percentage of the predictions fall below. For example, the 20th percentile means that all other scores fall below, or more simply put; the top 20% of users. Percentiles help account for the shape of a model's distribution. To determine who the best users are based solely on the prediction scores, if the distribution is skewed in any direction.

Updated over 1 year ago

Leveraging User Profiles

Suggest Edits

Welcome to the Leveraging User Profiles section, where we will explore how to use Lytics user profiles to gain a deeper understanding of user experiences. This section will cover how to define and segment audiences, personalize experiences, and track user behavior. As a data analyst, this section will provide you with the essential knowledge needed to leverage Lytics user profiles and gain insights into user behavior.

Updated over 1 year ago

Accessing Profiles Client Side

Suggest Edits

Accessing Data from JavaScript Tag

Once user fields have been surfaced, they can be used to personalize a user's experience. For example, a theoretical user's vehicle search results in the following data available in the browser.

JSON

```
{
  "last_make": "audi",
```

```

"last_model": "rx8",
"last_color": "onyx",
"segments": [
  "all",
  "potential_buyers",
  "high_momentum"
]
}

```

Populate a Form

Create a JavaScript function that populates form fields.

JavaScript

```

function(data){
  if(data.last_make !== ""){
    $('input[name="make"]').val(data.last_make);
  }

  if(data.last_model !== ""){
    $('input[name="model"]').val(data.last_model);
  }

  if(data.last_color !== ""){
    $('input[name="color"]').val(data.last_color);
  }
}

```

Initialize the callback handler.

HTML

```
<script type="text/javascript">
```

```

</script>!function(l,a){a.liosetup=a.liosetup||{},a.liosetup.callback=a.liosetup.callback||[],a.liosetup.addEntityLoadedCallback=function(i){var i=[];i.push(a.liosetup.callback),a.liosetup.callback=i}a.lio&&a.lio.loaded?l(a.lio.data):o?a.liosetup.callback.unshift(a.lio.data):a.liosetup.callback.push(a.lio.data)}(window,document)

```

Create a callback: This will execute an action via JavaScript once the visitor has been identified and the profile has been loaded.

HTML

```

<script type="text/javascript">
  window.liosetup.addEntityLoadedCallback(function(data){
    if(data.last_make !== ""){
      $('input[name="make"]').val(data.last_make);
    }

    if(data.last_model !== ""){
      $('input[name="model"]').val(data.last_model);
    }

    if(data.last_color !== ""){
      $('input[name="color"]').val(data.last_color);
    }
  });
</script>

```

Once the profile has loaded, Lytics will call the JavaScript function and populate the form on the site. Once all the above code is in place, the form will be populated with the visitor's profile data.

HTML

```

<!doctype html>
<html lang="en">
<head>

```

```

<meta charset="utf-8">
<title>form value demo</title>
<script src="https://code.jquery.com/jquery-1.10.2.js"></script>
<script type="text/javascript">
  // Lytics JavaScript Tag
  window.jstag=function(){function t(t){return function(){return t.apply(this,arguments),this}}function n(){var n=["rea
call(arguments)),this._q.push(n))}}var i={_q:[],_c:{},ts:(new Date).getTime(),ver:"2.0.0"},c=Array.prototype.slice;return i.
r(t),this},i.loadtagmgr=function(t){var n=document.createElement("script");n.type="text/javascript",n.async=!0,n.src=t
gName("script")[0];i.parentNode.insertBefore(n,i)},i.ready=n(),i.send=n("send"),i.mock=n("mock"),i.identify=n("identify
c.call(arguments,1))}},i.block=t(function(){i._c.blockload=!0}),i.unblock=t(function(){i._c.blockload=!1}),i},window.jstag
oadid:false});
</script>
<script type="text/javascript">
  // Lytics Callback Handler
  !function(l,a){a.liosetup=a.liosetup|[]|{,a.liosetup.callback=a.liosetup.callback|[]|[],a.liosetup.addEntityLoadedCallba
var i=[];i.push(a.liosetup.callback),a.liosetup.callback=i}a.lio&&a.lio.loaded?(a.lio.data):o?a.liosetup.callback.unshift(l)

  // Custom Callback Function
  // This particular callback simply checks that the three values are not empty strings and then uses jQuery to update
  window.liosetup.addEntityLoadedCallback(function(data){
    if(data.last_make !== ""){
      $('input[name="make"]').val(data.last_make);
    }

    if(data.last_model !== ""){
      $('input[name="model"]').val(data.last_model);
    }

    if(data.last_color !== ""){
      $('input[name="color"]').val(data.last_color);
    }
  });
</script>
</head>
<body>
<h2>A Sample Form</h2>
<form action="" method='post'>
  <label>Make: </label><input type='text' name='make' value="" /><br/>
  <label>Model: </label><input type='text' name='model' value="" /><br/>
  <label>Color: </label><input type='text' name='color' value="" />
</form>
</body>
</html>

```

This is one basic example of leveraging the current visitor's profile information. The personalization powered by Lytics will show new listings on the site's index page that match a user's last search or present other vehicles that user may be interested in.

Updated over 1 year ago

Accessing Profiles Server Side

Suggest Edits

Accessing Data from Server Side

By default, Lytics returns a user's profile and any surfaced fields to the browser. This, however, is not always sufficient. One approach is to do the lookup based on the user cookie, utilizing the server side language of your choice. First, read the browser cookie, then make a request to the Personalization to get the visitor's current profile, and then surface the profile as a JavaScript object.

Identifying the Visitor

Identifying the user can be done in a variety of ways. By default, Lytics sets a cookie `seerid` that is used to identify a user. The `_uid` field is what the Lytics JavaScript tag uses for web based identity resolution. That said, if users are logged in or identified by another means, you can use a different `seerid` in the following examples.

Since we already have a cookie, all we need to do is read those to get the user profile from the Lytics API.

PHP

```
$_COOKIE["seerid"];
```

Get Visitor's Profile

Once you have an identifier, make a GET request to our personalization API. In this example, the cookie value is used as the identifier. The API returns a JSON object with the following fields: `st_make`, `last_model`, `last_color`. If you are using a custom key value pair for identification, replace the `$fieldname` parameter with the value of that key.

PHP

```
<?php
```

```
$fieldname = "_uid";
```

```
$value = $_COOKIE["seerid"];
```

```
$apitoken = "{yourapitoken}";
```

```
$url = "https://api.lytics.io/api/entity/user/" . $fieldname . "/" . $value . "?fields=last_make,last_model,last_color&key=" . $api_key;
```

```
$data = json_decode(file_get_contents($url), true);
```

```
$profile = $data["data"];
```

?>

Surface User Data in the Browser

Next, parse the response and either inject the variables directly into your template or surface the profile as a JavaScript

PHP

<?php

```
$fieldname = "_uid";
```

```
$value = $_COOKIE["seerid"];
```

```
$apikey = "{yourapitoken}";
```

```
$url = "https://api.lytics.io/api/entity/user/" . $fieldname . "/" . $value . "?fields=last_make,last_model,last_color&key=" . $api_key;
```

```
$data = json_decode(file_get_contents($url), true);
```

```
$profile = $data["data"];
```

?>

<html>

<head>

```
<script type="text/javascript">
```

```

window.jstag=function(){function t(t){return function(){return t.apply(this,arguments),this}}function n(){var n=["rea
call(arguments)),this._q.push(n))]var i={_q:[],_c:{},ts:(new Date).getTime(),ver:"2.0.0"},c=Array.prototype.slice;return i
r(t),this),i.loadtagmgr=function(t){var n=document.createElement("script");n.type="text/javascript",n.async=!0,n.src=t
gName("script")[0];i.parentNode.insertBefore(n,i),i.ready=n(),i.send=n("send"),i.mock=n("mock"),i.identify=n("identify
c.call(arguments,1))},i.block=t(function(){i._c.blockload=!0}),i.unblock=t(function(){i._c.blockload=!1}),i},window.jstag
oadid:false});

```

```
var lyticsProfile = <?php echo json_encode($profile); ?>;
```

</script>

</head>

<body>

Make: <?php echo \$profile["last_make"]; ?>

Model: <?php echo \$profile["last_model"]; ?>

Color: <?php echo \$profile["last_color"]; ?>

</body>

</html>

Updated over 1 year ago

Working with Anonymous Profiles

Suggest Edits

Anonymous Visitors

Lytics considers anonymous visitors to be users identified only by a `_uid`. A `_uid` is a Lytics tracking value assigned to

browsers or computers, and aren't strictly associated with an individual, they are unreliable. For these reasons, Lytics

The User Profile of an Anonymous Visitor

Anonymous visitors can still have rich behavioral data. Every page they visited, the number of times they visited, the time they visited, and behavior patterns for data science models. A User Profile is nothing more than the set of known User Field values associated with the _uid of the Anonymous Visitor.

user profile anonymous empty

Identifying Anonymous Visitors

Since all information about anonymous visitors relies on the continued use of a cookie, it is in a marketer's best interest to convert these unknown users to known users. Cookies are temporary, but email addresses are not.

Note that when a user converts from anonymous to known, all information we knew about them carries with them. When their _uid becomes associated with an email address (or any other By Field), both Profiles (the _uid profile and the email address profile) merge through a process called Identity Resolution.

Updated over 1 year ago

Get Started with Lytics Segments

Suggest Edits

Get Started with Lytics Segments

Lets take an in-depth look at Lytics segments. We'll cover how to define a segment with SegmentQL or SegmentAST, scan, and list the segments in your account.

Note that throughout this guide we use the term segment and audience interchangeably. Specifically, the term audience is also a separate table and schema for content. A segment on this table is often called a content collection.

Difficulty Beginner

TLDR: Checkout the Segment API documentation.

Other resources:

Lytics Go Client - Lytics API library for the go programming language.

QLBridge - a go SQL Runtime Engine.

Lytics Segment Scan with SQL.

Prerequisites

httpie - (optional) a more user friendly version of cURL that we'll be using in this guide for readability.

Lytics Command Line Tool - (optional) we'll show you how to execute commands with our CLI for each endpoint that

A valid API token for your Lytics account (learn about managing API tokens). For ease of use we suggest adding as an alias in Bash

```
export LIOKEY={API Token}
```

1. Defining A Segment

If you haven't yet, take a minute to check out the audience builder in your account. This can be accessed from the Audience Builder audience.

The audience builder is a visual interface for what we will cover in the first section of this guide - defining a segment. A segment is a logical statement. If the data we have collected on a user meets the conditions of the segment then they are considered a member of that segment definitions: SegmentQL, and SegmentAST.

SegmentQL

SegmentQL is simple Query Language using filter statements. Before we dive too deep into the syntax, lets look at a simple SegmentQL definition.

Example Segment

```
FILTER OR (
```

```
  AND (
```

```
    cm_status = "active"
```



```

    INCLUDE last_open_7_days
  )
  last_active_ts > "now-7d"
)

```

FROM user

ALIAS active_web_or_email

In this example cm_status refers to the subscriber status in Campaign Monitor, an email provider. So a marketer might cross their email and web channels in the last 7 days.

For us developers, this simple query language is probably quite intuitive! Our UI exists for marketers to build audiences and create a segment through the API. Just to cover all our bases lets take an in depth look at the anatomy of a filter statement.

```

FILTER [phrase] [from] [alias]
[phrase] Notes
AND ([phrase], [phrase], ...)
OR ([phrase], [phrase], ...)
NOT [phrase]
EXISTS [identifier] [identifier] is a field name.
[identifier] IN ([literal], [literal], ...)
[identifier] CONTAINS [literal]
[identifier] INTERSECTS ([literal], [literal], ...) INTERSECTS = "contains one of"
[identifier] LIKE [literal] [literal] must be a string. Can use * a wildcard
INCLUDE [identifier] [identifier] is a segment id/slug. (to include existing segment)
[identifier] > [literal]
[identifier] >= [literal]
[identifier] < [literal]
[identifier] <= [literal]
[identifier] = [literal]
[identifier] != [literal]
[identifier] = alphanumeric string (often the name of a data field, table, or segment).
[literal] = a string, int, float, boolean, or timestamp literal value.
[from] Notes
FROM [identifier] [identifier] is the name of the table ie. user, content, etc.
[from] is optional if creating a filter on the user table.
[alias] Notes
ALIAS [identifier] [identifier] is the saved name of the segment.
[alias] is optional if you do not wish to save the segment.
Segment AST
SegmentAST expressions look very similar to SegmentQL but instead of having its own filter syntax it is formatted in JSON.
When using a program which creates or updates segment definitions it might be easier to format the bodies of your requests as SegmentAST.

```

Example Segment

JSON

```

{
  "op": "or",
  "args": [{
    "op": "and",
    "args": [{
      "op": "=",
      "args": [{
        "ident": "cm_status"
      }],
      {
        "val": "active"
      }
    ]
  }],
}

```

```
{
  "op": "include",
  "args": [{
    "ident": "last_open_7_days"
  }]
},
{
  "op": ">",
  "args": [{
    "ident": "last_active_ts"
  }],
  {
    "val": "now-7d"
  }
}]
}
```

Definitely not as easy to look at as SegmentQL, but if you spend a second scanning the JSON that you'll see it's general enough to handle a wide variety of operations:

key type Notes

op string operator used to evaluate the args

args array a list of objects containing the identifiers and literals operate on

ident string identifier name

val string literal value

2. Segment Validate API

The segment validate endpoint accepts a plain text SegmentQL statement and ensures that the definition is valid. Though if you're making a request to build the segment via the API you might want to validate it first.

Shell

```
echo 'FILTER AND (
  utm_campaigns CONTAINS "summer_froyo_promo",
  EXISTS email,
  scores.momentum > 40
)' | http POST https://api.lytics.io/api/segment/validate access_token==$LIOKEY
```

Note that this endpoint does not validate field names in your account. It simply checks the syntax of the SegmentQL.

3. Segment CRUD API

Segment Create

The segment create endpoint accepts plain text (containing SegmentQL) or a JSON body (SegmentQL or AST).

Bash

```
# example 1: plain text
echo 'FILTER (
  lytics_content.Yogurt >= 0.75
)
FROM user
ALIAS affinity_for_yogurt
' | http POST https://api.lytics.io/api/segment Content-type:text/plain access_token==$LIOKEY
Bash
```

example 2: json segmentQL

```
echo '{
  "name": "High Affinity for Yogurt",
  "segment_ql": "FILTER (lytics_content.Yogurt >= 0.75)",
  "slug_name": "affinity_for_yogurt",
  "is_public": true,
  "description": "Segment containing all users highly interested in the topic Yogurt."
}
```

```
"table": "user"
```

```
} | http POST https://api.lytics.io/api/segment Content-type:application/json access_token==$LIOKEY  
Bash
```

```
# example 3: json segmentAST
```

```
echo '{  
  "name": "High Affinity for Yogurt",  
  "slug_name": "affinity_for_yogurt",  
  "is_public": true,  
  "description": "Segment containing all users highly interested in the topic Yogurt.",  
  "table": "user",  
  "ast": {  
    "op": ">=",  
    "args": [{  
      "ident": "lytics_content.Yogurt"  
    }],  
    {  
      "val": "0.75"  
    }  
  }  
}'
```

```
} | http POST https://api.lytics.io/api/segment Content-type:application/json access_token==$LIOKEY  
Segment Read
```

You must know the segment ID or slug name to fetch the segment via the read endpoint.

Bash

```
# example 1:
```

```
http GET https://api.lytics.io/api/segment/affinity_for_yogurt access_token==$LIOKEY  
Bash
```

```
# example 2: lytics cli
```

```
lytics --segments=affinity_for_yogurt segment
```

Segment Update

You must know the ID or the slug of the segment to update.

Bash

```
echo '{  
  "slug_name": "affinity_for_yogurt",  
  "segment_q1": "FILTER (lytics_content.Yogurt >= 0.9)"  
}' | http PUT https://api.lytics.io/api/segment/affinity_for_yogurt access_token==$LIOKEY  
Segment Delete
```

You must know the ID or the slug of the segment to delete.

Bash

```
http DELETE https://api.lytics.io/api/segment/affinity_for_yogurt access_token==$LIOKEY
```

4. Segment List API

For many of the CRUD operations we just talked about you need to know the ID or slug name of the segment. It may not be convenient to use as a reference. The list endpoint will return a list of all segments for your account including id, slug, and the

Bash

```
# example 1: list all audiences
```

```
http GET https://api.lytics.io/api/segment access_token==$LIOKEY  
Bash
```

```
# example 2: lytics cli
```

lytics segment
Bash

```
# example 3: list all content collections
http GET https://api.lytics.io/api/segment access_token==$LIOKEY table==content
5. Segment Scan API
```

The scan endpoint provides a list of all entities in a segment. For audiences (segments on the user table) the entities number of different query parameters, we won't cover all of them here, but as always you can check out our docs for the

You can use this endpoint to scan an existing, saved segment in which case you'll need the segment ID or slug. You can also scan a new segment.

Bash

```
# example 1: existing audience
http GET https://api.lytics.io/api/segment/affinity_for_yogurt/scan access_token==$LIOKEY sortfield==last_purchase_timestamp
Bash
```

```
# example 2: lytics cli
lytics --id=affinity_for_yogurt segmentscan
Bash
```

```
# example 3: ad-hoc content collection
echo 'FILTER OR (
  global.Yogurt > 0.5,
  global.`Ice Cream` > 0.5
)
FROM content
' | http GET https://api.lytics.io/api/segment/scan access_token==$LIOKEY limit==10
Bash
```

```
# example 4: ad-hoc lytics cli
lytics segmentscan '
FILTER (
  aspects INTERSECTS ("article", "video")
)
FROM content'
```

If the total number of entities in the segment exceeds the limit provided in the request (the default value is 20) then the API will return a token value with the key next. To get the next page of results in the scan, make a subsequent request with the query token. The API will continue to return next tokens in each response until you've fully scanned the segment.

What's next?

The endpoints we covered here will get you started with Lytics segments. We do have a couple segment APIs that we haven't covered for this is quite simple, check out our docs if you're interested. It also may help to look at the catalog schema and/or the Lytics account. Keep in mind that everything you see in the Lytics web UI is built off of our APIs. It is never "necessary" to use the time and resources you can build many different custom applications to interface with your Lytics account through the API, creating new use cases. If our docs aren't enough you can always reach out to us.

Updated over 1 year ago
Inline Content Recommendations
Suggest Edits
Inline Content Recommendations

Learn how to add seamless content recommendations to your website fueled by Lytics' Content Affinity Engine; all it takes is a few lines of code.

Difficulty: Beginner

TLDR: Here's some docs instead.

Other resources:

A blog post - covering basically the same thing - with a live example!

Lytics Segment API Docs (for segment creation API and SegmentQL)

Prerequisites

Lytics Javascript Tag installed on your website.

Pathfora SDK installed on your website. If you're using Lytics for personalized modals, this is already done for you!

Content classified in your Lytics account (go to the content section of your Lytics account to check if we've classified y

A valid API token for your Lytics account (learn about managing API tokens). For ease of use we suggest adding as an

Shell

```
export LIOKEY={API Token}
```

1. Build a Content Collection

Before we go crazy adding recommendations to your website we should think about what kind of content we want to

ill be the target pool of documents for this content recommendation block.

To do this, we can use the Lytics Segment API create endpoint.

Bash

```
curl -s -XPOST "https://api.lytics.io/api/segment" \
```

```
-H "Content-type: text/plain" \
```

```
-H "Authorization: $LIOKEY" \
```

```
-d '
```

```
FILTER AND (
```

```
  path CONTAINS "froyo/flavors"
```

```
)
```

```
FROM content
```

```
ALIAS froyo_content
```

```
' | jq ''
```

In the command above, we made a POST request to create a new segment on the content table, or as we call it - a co

ntent in your account. In the example above, we're only accepting urls that contain the path froyo/flavors. Simple en

that the logic for building a segment is essentially the same format.

Here's a more complex example:

```
FILTER AND (           // AND operator on the following statements
```

```
  global CONTAINS "strawberry" // require the topic strawberry
```

```
  global CONTAINS "vanilla"    // require the topic vanilla
```

```
  path CONTAINS "posts"        // url must contain the path "/posts/"
```

```
  published > now-30d          // require published in the last 30 days
```

```
)
```

```
FROM content           // entities from the content table
```

```
ALIAS strawberry_vanilla // saved name of the collection
```

You're probably wondering where these field names are coming from! These are the standard fields used in our con

g a content collection:

fieldname type description

global map[string]number a map of topics for the document and their relevance scores

aspects []string type of document: eg. article, profile, video, etc

primary_video string url of the primary video featured in the document

primary_image string url of the primary (usually meta image) - can be displayed in recommendation

language string language of the documents

wordct int word count for the document

author string name of the author of the document

product_description string description of the product (if the document is of type product)

price string price of the product (if the document is of type product)

description string title of the document

long_description string meta description of the document

url string url of the document

path []string list of paths within the url of the document

published date date when the document was published
fetched date date when the document was fetched and enriched by Lytics
type string source channel of the document (eg. web, email, etc)
Also if you have access to your queries in the Data tab of the Lytics UI you can see how the raw data from the lytics_

2. Set up an inline recommendation

To add inline content recommendations to your website all we have to do is draft some simple HTML. Pathfora will handle the logic of these elements with recommendations from our APIs.

First lets look at a simple, quick example, then we'll break down what these attributes mean:

HTML

```
<div data-pfblock="froyo_1" data-pfrecommend="froyo_content">  
  <img data-pftype="image" alt="frozen yogurt recommendation">  
  <h2 data-pftype="title"></h2>  
  <p data-pftype="description"></p>  
  <p><a data-pftype="url">Read More...</a>  
</div>
```

Here's a quick rundown of the relevant data attributes:

attribute name value description

data-pfblock a unique name for a single recommended document (there may be multiple documents per page)

data-pfrecommend id or slug (saved name ALIAS) of the source content collection

data-pftype field of the document to set as the contents of that element (ex. url, title, description, image, author, published)

image data-pftype attributes will set the src value if applied to an img element (as with the example above) or if used with the data-pftype="round-image" attribute to be the primary image from the document.

Similarly, the url data-pftype attribute will set the href value to the url of the document if applied to an a tag, but otherwise it will be ignored.

Multiple Recommendations

You may have seen a section on blogs "If you liked this, you may also like..." with a list of three or more related articles.

We repeat the same HTML pattern to create multiple recommendations in a set, just remember to change the data-pfblock attribute to a different content collection then we will ensure that the same piece of content is not shown twice on one page.

Here's a more in depth example, with a little CSS this can be easily styled to match your website:

HTML

```
<h1>Recommended Content</h1>
```

```
<!-- Recommendation 1 of 2 -->
```

```
<div class="rec-block" data-pfblock="froyo_1" data-pfrecommend="froyo_content">  
  <div class="rec-img" data-pftype="image"></div>  
  <h2 data-pftype="title"></h2>  
  <p class="rec-info">by <span data-pftype="author"></span> on <span data-pftype="published"></span></p>  
  <p data-pftype="description"></p>  
  <p><a data-pftype="url">Read More...</a>  
</div>
```

```
<!-- Recommendation 2 of 2 -->
```

```
<div class="rec-block" data-pfblock="froyo_2" data-pfrecommend="froyo_content">  
  <div class="rec-img" data-pftype="image"></div>  
  <h2 data-pftype="title"></h2>  
  <p class="rec-info">by <span data-pftype="author"></span> on <span data-pftype="published"></span></p>  
  <p data-pftype="description"></p>  
  <p><a data-pftype="url">Read More...</a>  
</div>
```

Additional Notes

Our SDK may take a second to set the proper hide/show settings for inline elements. If you experience a "flicker" when the recommendations first load, this is normal.

you can add the following CSS line to your website to prevent this:
CSS

```
[data-pftrigger], [data-pfrecommend] { display: none; }
```

This exact CSS gets loaded by Pathfora, but adding it to your CSS ensures that elements are hidden as the document

Since this is primarily a developer use case we don't provide any native reporting of inline recommendations at the m
ction data to Lytics or Google Analytics. Check out our Lytics Javascript Tag documentation for more.

If you'd like a "pre-baked" version of inline recommendations, we also have a version that's built from our personaliz

Updated over 1 year ago

Data Collection & Onboarding

Suggest Edits

Welcome to the Data Collection and Onboarding section, where we will explore the process of collecting and onboard
e data quality and accuracy, including how to integrate with various data sources and best practices for data manage
analyst, this section will provide you with the essential knowledge needed to gain deeper insights into customer beh

Updated over 1 year ago

Working with Custom Data

Suggest Edits

Custom Data Sources

Lytics can onboard and process data from custom data sources, helping to build out further User Profiles with data s
or your marketing efforts as it
allows you to personalize web experiences and build meaningful audiences using data unique to your users.

Onboarding process

Unlike data collected from web activity and integrated marketing tools which automatically map data to Lytics user fi
mplementation process. The Lytics team will assist in identifying use cases and the data fields needed to achieve the
s the structure, format, location and relationship of your data. The data dictionary is used to write queries which map
inally your custom data is uploaded to the Lytics and made available for use to build audiences, personalize web exp

Uploading custom data

Data can be uploaded to Lytics via CSV SFTP upload or our collection APIs. Lytics can ingest both large, bulk uploads
re uploading data to Lytics, ensure all necessary LQL queries are in place or it will not be mapped to Lytics user fields
oads to the Lytics platform, be aware that this may cause an event backlog on your account. This occurs as our platfo
e upload. Processing begins when the upload completes and can last considerably longer than the initial upload. Thi
n your account as well as delays in any outbound, trigger-based workflows.

Custom Data Ingestion

Getting your custom data ready to be properly ingested is crucial during your onboarding process with Lytics. This d
om data sources sent to Lytics via batch CSV or JSON, utilizing S3 or SFTP import workflows, as well as batch or real-t

File naming

File compression

Field formatting

Headers

Timestamps

JSON formatting

File naming

When you have a recurring bulk import, from S3 or SFTP for example, you must consistently follow naming conventi

Keep file naming consistent by determining casing and spacing.

E.g. all lower-case, use underscores for spaces: file_source_1_date.

Name each successive file with an identical 'root' along with a time-based suffix such as YYYYMMDD in filetype-20191

Lytics will import files that match the 'root' filename and will use the modified timestamp of the file to determine the
port for continuous imports.

File compression

If needed, files may be compressed using the zip format prior to ingestion. The zip file will be decompressed and del

Field formatting

Phone numbers should be standardized. Lytics suggests normalizing phone numbers in a format such as 1222333344.

Omit double quotes or escape quotes.

Omit newlines.

Keep all free form text in quotes if possible.

Avoid page breaks or special characters.

For more, reference the basic rules for CSV files.

Headers

Keep headers consistent across your organization and your vendors by determining casing and spacing.

Column headers have to match the sample file exactly.

When adding a new source, review current mappings and headers to determine if any headers need to be mapped or renamed.

E.g. if the field mobile comes in from source A, and the field cell comes in from source B, it's likely these should be mapped to a single field, e.g. mobile.

Timestamps

Lytics is able to ingest data in a different order than the events described in that data transpired. For this to work, imports must be timestamped with them. For workflow-based imports, select the key/column which contains the event timestamp when importing. For REST API field URL parameter.

All imports require the following format: YYYY-MM-DDTHH:MM:S. If an explicit timestamp is not specified, the data will be timestamped at the time of import.

When Lytics looks for new files, it will choose first based on the file's last modified date. If multiple files have the same last modified date, it will use the file's name to select the next file to import.

JSON formatting

JSON file formatting varies slightly depending on the method and nature of the data to import. See examples for bulk and real-time imports.

Bulk

Files imported via S3, SFTP, or bulk collection should be newline delimited, meaning each object represents a single record.

```
{"event":"register","date":"2014/04/05"}
```

```
{"event":"login","date":"2014/04/05"}
```

Real-time

Files sent to collection should be formatted as regular JSON, where each record/event is an object in an array of objects.

```
[  
  {"event":"register","date":"2014/04/05"},  
  {"event":"login","date":"2014/04/05"}  
]
```

Lytics has a limited ability to parse nested data. Objects may contain other objects and arrays, but objects may not contain arrays.

Updated about 1 year ago

Working with Web Data

Suggest Edits

Onboarding Web Data

Lytics provides two ways to onboard data from your website, email marketing, and online advertising. The Lytics JavaScript tag sends data to Lytics. The Lytics pixel can be embedded in email or ads and configured to send user data back to Lytics on load.

Lytics JavaScript Tag

The Lytics JavaScript tag sends page views and any custom events you have configured to the default stream (unless otherwise specified). It also sends all fields in the Lytics schema with predefined or custom mappings. For example, the following request would send a user's name and email address to the default stream.

JavaScript

```
jstag.send({
```



```
"email": "gump@hanks-roles.com",  
"first_name": "Forrest",  
"last_name": "Gump",  
})
```

Using a tag manager

Our tag can be installed manually or with a tag manager such as Google Tag Manager. If the Lytics JavaScript tag is in the correct function to call.

Lytics Image Pixel

When the Lytics JavaScript tag cannot be installed, the Lytics Image Pixel can be used to onboard user data from any new ads.

Building the pixel

There are three key variables that need to be defined when sending information to Lytics using the Lytics Image Pixel: event data in the form of query parameters. For example, this pixel will send the user's email address and Google Universal Analytics stream of account 24546133b65465413w9.

HTML

```

```

This image tag can be inserted into any HTML document including advertisements and emails and will send the event data to Lytics.

HTML

```
<html>  
  <head />  
  <body>  
    <h1>Welcome to Hanks Roles!</h1>  
    <p>Thanks for signing up for all the Tom Hanks news you handle.</p>  
  
    <!-- lytics pixel at bottom of email -->  
      
  </body>  
</html>
```

Predefined user fields

Lytics does not have a fixed schema, any key-value pair can be passed using the JavaScript tag or pixel. However, only the following fields are used in audiences. Out of the box, the Lytics JavaScript Tag and Image Pixel have the ability to onboard the following fields:

General Visit Data

Name Identifier
Time of Last Visit lastvist_ts
Time of First Visit firstvisit_ts
Total Number of Visits visitct
All Channels Used channels
Web Events By Hour hourly
Web Events By Week weekly
Domains Visited domains
Domains Referred From refdomain
Devices Used devices
Has Accessed Mobile Web is_mobile
Last Visit Country visit_country
Last Visit City visit_city
Last Visit State/Province visit_region
Timezone timezone
User is a bot is_bot
User Agent user_agent
Hashed URLs Visited and Total View Count hashurls
User Identity
Name Identifier
Web Cookie ID _uid

User ID user_id
Full Name name
First Name first_name
Last Name last_name
Email email
Title title
Company company
Phone Number phone
Cell Number cell
Age age
Gender gender
City city
State state
Zip zip
Country country
Origin origin
User Status status
Event Data

Events regarding specific user actions such as a CTA click can also be sent to Lytics. In addition to any of the parameters you send request resulting in a few more fields on the profile:

Name Identifier
Total Pageview Count pageviewct
Event First Time Seen event_first_seen
Event Last Time Seen event_last_seen
Campaign Data (Google UTM)

Google UTM campaign parameters are automatically pulled in and added to profiles as well. For more details on Google UTM parameters, see [Google UTM Parameters](#). The following fields will be added to the profile when UTM parameters are received:

Name Identifier
UTM Campaign Referred By utm_campaign
UTM Source Referred By utm_source
UTM Medium Referred By utm_medium
UTM Content Referred By utm_content
Last UTM Campaign Referred By utm_campaign_last
Last UTM Source Referred By utm_source_last
Last UTM Medium Referred By utm_medium_last
Last UTM Contents Referred By utm_content_last
Form Data

In order to collect data entered on a form, you will need to use `jstag.send()` when a user submits that form. The field name for the form is passed as `form_name`, and any fields in the form need to start with `formdata_`, for example `formdata_country`.

Name Identifier
Last Time Web Form Submitted last_form_submitted_by_date
Web Forms Submitted form_submitted
Web Form Data form_data
Conversion Data

As a convenience, a standard conversion event has been defined in order to surface some useful conversion related data. When passing conversion events, the event name must be "conversion" for the following mapping to occur:

Name Identifier
Most Recent Conversion Time cvt_last_time
Oldest Conversion Time cvt_first_time
Most Recent Campaign Attributed to Conversion cvt_last_campaign
All Campaigns Converted From cvt_campaigns
Most Recent Variation Attributed to Conversion cvt_last_variation
All Variations Converted From cvt_variations
Most Recent Currency Used cvt_currency
Most Recent Conversion Value cvt_value
Conversion Values By Campaign cvt_history

Updated about 2 months ago

Collect Mobile Data with Firebase + GTM

Suggest Edits

The following tutorial provides step-by-step guidance for setting up, configuring, and testing data collection from a mobile app using Google Tag Manager (GTM). This is an excellent alternative for minimizing the number of SDKs that must be deployed from your app, while still collecting data from your app in Firebase Analytics using GA4.

In this tutorial, we'll cover the following steps:

Setting up Firebase and installing the SDK.

Setting up Google Tag Manager and installing the SDK.

Configuring the Lytics Image Pixel collection method.

Testing the implementation end-to-end.

Benefits

Events are delivered to Lytics in real-time.

Reduces the number of SDKs installed.

Leverages Google-managed SDKs to streamline app approval.

Easy access to advanced testing and debugging tools.

Automatically passes events to Google Analytics or analysis or further delivery to BigQuery.

Install Firebase SDK

This solution leverages Firebase Analytics which Google Analytics powers under the hood. This means that in addition to using Firebase Analytics, you also get the benefits of Google Analytics and its direct integrations to other Google products such as BigQuery as an automatic value add.

Google provides thorough documentation for getting a Firebase project set up and installed in your platform of choice. [firebase-1](#)

Once you have completed the setup and made it to the Start logging events section of the documentation, you are ready to start logging events. If you need further help, refer the documentation, either way, we'll come back here further down in this tutorial.

Install Google Tag Manager SDK

Next, we'll leverage Google Tag Manager, which offers seamless integration with Firebase Analytics to forward events to Lytics. For more details, refer the documentation for getting a Google Tag Manager account set up and the iOS or Android SDKs configured. [gtm-2](#)

Sending User Properties & Events

Now that we have configured Firebase and Google Tag Manager, we are ready to start configuring and sending our events. This makes sending user properties and interaction-based events easy. For more details on the power of Firebase, you can revisit the Start Logging Events section.

```
Analytics.logEvent("did_something", parameters: [
  "_uid": UUID().uuidString,
  "lytics_test_key": "some value",
])
```

The above represents an example of a custom event named `did_something`. For that event, we also include two parameters. These parameters are user identifiers and should be in support of your use case. Keep a record of the parameters used, as you will need to explicitly define them in your Lytics account.

That said, including an identifier of some for every event is essential to your identity resolution strategy. In the above example, we use a UUID to generate a unique user identifier. Consult your account manager or solutions team for the optimal approach here. Regardless of the approach, it is essential to generate and store a value on the device. For this step, it is best to consult your app development team.

One example would be to use Swift's UUID to generate a unique user identifier and store it using UserDefaults. The next step is to log the results. It is only essential to include that value in all events and have it persist for as long as possible. On the other hand, you can include multiple identifiers such as email, or IDFA, to maximize the match rate.

Configuring Google Tag Manager for Event Forwarding

The final step in getting user details from your app to Lytics is configuring your Google Tag Manager Container. We'll configure the container to forward events from your app. Based on our example, we'll define two variables: `_uid` and `lytics_test_key`.

Step One

From the Google Tag Manager dashboard, navigate to Variables in the left-hand menu.
step1-3

Step Two

Create a new variable, one for each parameter, using the New button. In the new window that loads, click Choose a variable type. You can use any variable type based on your needs. Event Name may also prove helpful in adding more context to the variable.
variable-4

Step Three

Finish your variable configuration by selecting Custom Parameter, followed by inputting one of the parameter names as the value of an empty string. Save your settings and move to the next parameter until you have created a variable for all parameters.
variable2-5

Step Four

With your variables defined, we'll move on to creating a new Tag within our container. Select Tags from the left-hand menu. In the new window, click Choose a tag type and select Custom Image.
tag-6

Step Five

With your tag type selected, we'll configure the Image URL as well as the Triggering rules. The Image URL will leverage the Lytics API. I need to know your Account ID.

For simplicity, we'll break the full URL into two parts that will be joined together in actual execution.

Focusing first on the path, everything from https up to the ? you will need to replace the YOUR-AID portion of the path with your desired ingest stream.

`https://c.lytics.io/c/YOUR-AID/STREAM`

The query parameters will contain the field name you'd like to pass to Lytics, as well as a value that is populated by your variables.

`?_uid={{Lytics UID}}&lytics_test_key={{Lytics Test Key}}`

Putting those two parts together results in the final value of the Image URL in our config.

`https://c.lytics.io/c/myaccountid123/mobileevents?_uid={{Lytics UID}}&lytics_test_key={{Lytics Test Key}}`

Last we'll configure the event that triggers our forward by selecting All Events.

tagconfig-7

Step Six

With our tag defined, the final step is publishing the Google Tag Manager container and storing the config that is generated. This provides instructions for doing so.

Test & Debug Implementation

With our implementation all configured, the final step is to test and debug. Each of the steps defined above supports testing and debugging as shown below.

Firestore Event Debugger

Google Tag Manager Debugger

Lytics Data Streams: Once successfully tested, you can do the final validation by reviewing your incoming data as part of the Lytics Data Streams.

Updated over 1 year ago

Data Management

Suggest Edits

Welcome to the Data Management section, where we will explore the various tools and techniques available within the Lytics platform. This section covers how to ensure data accuracy and completeness, including data validation, deduplication, and transformation. As a Lytics administrator, this section will provide you with the essential knowledge needed to manage your customer data and optimize your data pipeline.

Updated over 1 year ago

Profile Stitching Best Practices

Suggest Edits

Stitching User Profiles

This guide gives context on Lytics user profiles and helps illustrate the most common ways to combine or stitch profiles. If you are not a developer, you will likely need a developer on your team to implement some of these tactics.

The problem

„How do I know that an anonymous user browsing my website is the same user that just opened my email?...

„How can I make sure I am not over-marketing to the same user because I can't tell who they are across all my data sources?...

„How do I achieve 1:1 personalization without a single source of truth for my users?...

How Lytics helps

Lytics aims to solve these problems by creating a holistic customer view through data unification. Lytics is based on unified data. A user profile is generated and defined by an identifier:

For anonymous users, this identifier can be the Lytics cookie placed after you install the Lytics JavaScript tag.

For known users, this is most commonly an email, but can also be various forms of user IDs across your data sources.

Richer profiles that include attributes from various data sources allow marketers to run their campaigns more efficiently, resulting in lower cost per action (CPA), among other results. Therefore, profile stitching early on in your journey with Lytics will benefit your marketing efforts.

But how does Lytics actually stitch your Customer Relationship Management (CRM) users, to your web browsers, to your email clients, and so on? The answer lies in the identifiers that connect these different tools.

Common identifiers

A common identifier is used by Lytics to create a user profile, but it's one that also exists in another source. Some examples include:

User 1: Imported from email tool

Identifier: email address (abc@123.com)

User 2: Imported from CRM

Identifier: email address (abc@123.com), user ID (0101010)

User 3: Imported from data warehouse

Identifier: user ID (0101010)

All of these users will merge into one profile in Lytics because they share identifiers. This will happen automatically at the next merge event.

Merging events

But how would you merge an anonymous profile that ONLY has a cookie ID with a profile that has a known identifier, like an email address, and one that will set you up for success.

To do this, you'll need to identify or create potential merge events. Again, more details can be found in the Profiles and Segments documentation, which includes examples of common scenarios that you may consider when planning your strategy.

Lytics Modal Experiences

You can build Email Capture Modal Experiences directly in Lytics that prompt users to sign up for an e-newsletter, or to provide their contact information. If a user provides an email, Lytics will tie the anonymous cookie profile to the email they submitted.

If their email was already in our system under a different profile, Lytics will merge the two profiles into one. For information on how to build these experiences, see the Experience Editor documentation.

On-site events: forms

Any of your website forms where a user submits an email are great places to create a merge event of a Lytics cookie. Tag Manager Trigger, Lytics can attach an email to an anonymous profile. Some types of commonly used forms include:

User log-in

Any sign-ups (events, newsletters)

Registrations / warranties (products)

Gated content (webinars, whitepapers)

Note that you can submit other data using this method. For example, if users fill in their name, profession, address, and end it to their profile.

Tip: You can also send Lytics a form name and timestamp using this method, which could be helpful if you'd like to record a specific period.

Keep in mind this will only work for forms living on your website. For forms populated in an iframe or hosted in a different domain, you'll need to capture this information.

See the document [Sending Data to Lytics](#) for more information on using the JavaScript tag and other methods.

Email click-through identifier

Many email service providers (ESPs) have a unique user ID within their system, separate from email. It's encouraged to use this identifier as a query string parameter within links you include in any email communication. If you are familiar with UTM parameters, this is similar.

Once you set this up in your ESP, Lytics can associate this additional identifier as another merge rule. So moving forward, when you return to your website with the Lytics tag, Lytics will be able to stitch the email to a cookie.

Examples:

`domain.com/news/article_name?<your_esp_identifier_or_hashed_email_name>=<value>`

`domain.com/news/article_name?esp_id=2ncihj894y2jnknfjsd`

Tracking growth

Next, you can create an audience in Lytics to track the growth of profiles that have stitched a cookie to an email. An example identifier is shown below:

`stitching_guide_img_2`

On the Audience Summary page, you can view how your audience of unknown to known users has grown over time.

`unknown-known-audience-summary`

Updated about 1 year ago

Migrating from Queries to Conductor Schema

Suggest Edits

Deep Dive: Migrating from Queries to Conductor Schema

TL;DR: Many things you liked about LQL are still here, and many things you didn't like are out. Migrating your schemas to Conductor is a big change, but your workflows.

Intro

Lytics' profile pipeline processes your customer data in real-time to perform data cleansing, data transformation, data enrichment, and more.

Historically, these functions were performed in an expressive SQL-like scripting language called LQL. After listening to feedback, we developed to ensure reliability while enabling process scalability, we introduced a new Schema API. Conductor provides a more robust and scalable way to manage your data.

Longtime Lytics users who are comfortable with LQL have appreciated:

Brevity: LQL combined both data definition (DDL) and data manipulation (DML).

Readability: LQL reads similarly to SQL, which makes sense to a lot of database practitioners.

Longtime Lytics users who were uncomfortable with LQL didn't appreciate:

DRY violations: A common development mantra states Don't Repeat Yourself (DRY). Repeating your code in multiple places to map data into the same field could yield inconsistent or unexpected results.

Lack of versioning: When you post LQL, you're doing it live! This can be problematic when there are large changesets or when you need to rollback.

at changed at a given time.

Learning curve: Not every update needs to be complicated, and requiring users to learn LQL to make even small changes. Lytics' new Schema API allows you to use LQL expressions within a JSON framework with native version control.

Analogs

An LQL statement combines features of data definition and data manipulation. In the Schema API, data definition is split into two parts: field mappings and mapping transformations.

Field Analog Properties

LQL Schema API

AS ID

SHORTDESC ShortDesc

LONGDESC LongDesc

KIND Type

MERGEOP MergeOp

CAP Capacity, KeepDays

BY IsIdentifier

Mapping Analog Properties

LQL Schema API

‡ ID

AS Field

FROM Stream

KIND Expr

IF Guard

Example

Imagine we had the following simple LQL Select statement processing data from a "purchases" data stream. This statement would select all purchases that have been refunded over their lifetime.

SQL

SELECT

SUM(order_total) AS `refunded` IF eq(type, "refund") SHORTDESC "Refund Amount" LONGDESC "Total USD refunded"

email(email) AS `email` SHORTDESC "Email Address"

FROM orders

INTO user

BY email

Instead of providing these definitions and transformations in a single statement, we would first create the fields into the schema. The email and refunded fields would look like the following.

JSON

```
[
  {
    "id": "email",
    "shortdesc": "Email Address",
    "type": "string",
    "mergeop": "latest",
    "is_identifier": true
  },
  {
    "id": "refunded",
    "shortdesc": "Refund Amount",
    "longdesc": "Total USD refunded",
    "type": "number",
    "mergeop": "latest"
  }
]
```

Once we've defined the fields, we would define the mappings for both, which would look like the following.

JSON

```
[
  {
    "id": "abc123",
    "field": "refunded",
    "stream": "orders",
    "expr": "SUM(order_total)",
    "guard": "eq(type, \"refund\")"
  },
  {
    "id": "def456",
    "field": "email",
    "stream": "orders",
    "expr": "email(email)"
  }
]
```

After creating fields and mappings for the desired schema changes, simply publish your changes to see their effect in

How to Get Started

Getting started is simple. All of your queries have been automatically translated into appropriate fields and mapping

It is recommended to immediately publish your default schema before making any changes to it. After it has been published, you can see the effect of your changes on what changes go out.

While you're not required to immediately start using Lytics' new Schema API, the classic Query API will be deprecated

Updated 11 months ago

LQL & Data Import Basics

Suggest Edits

LQL & Data Import Basics

Let's take an in-depth look at Lytics Query Language (LQL) and custom data imports. We'll cover a basic overview of LQL, how you can map custom data and finally a few of the different ways to import custom data into your account.

Difficulty: Intermediate

TLDR: Checkout the data upload documentation and the query documentation.

Prerequisites

httpie - (optional) a more user friendly version of cURL that we'll be using in this guide for readability.

Lytics Command Line Tool - (optional) we'll show you how to execute commands with our CLI for each endpoint that

A valid API token for your Lytics account (learn about managing API tokens). For ease of use we suggest adding as an alias in your Bash

```
export LIOKEY={API Token}
```

1. Lytics Data Processing Review

Due to the intermediate nature of this course it is important that you already possess a strong understanding of segmenting and the Lytics data pipeline we recommend reviewing the following documentation before proceeding with the rest of the steps

Introduction: Lytics Data Collection

Introduction: Lytics User Profiles & User Fields

Introduction: Lytics Audiences

2. Sending Data from a Website (JavaScript)

When the Lytics JavaScript tag is installed we will automatically begin collecting standard data. This includes page-view events, etc. In addition to the standard page-view event the Lytics tag empowers marketers to push more sophisticated data to

An example of such data to a custom stream called custom_data might look something like:

JavaScript


```
jstag.send('custom_data', {
  userid: "1234",
  email: "jon@castleblack.com",
  first_name: "Jon",
  last_name: "Snow",
  title: "King in the North",
  company: "House Stark",
  phone: "555-555-5555",
  age: 21,
  gender: "m",
  city: "Winterfell",
  state: "Westeros",
  status: "planning"
});
```

While visiting your site that has the Lytics JavaScript tag installed you can leverage Chrome's developer tools, specifically the Network tab, to pass the payload above to Lytics. Keep in mind, all data sent from your browser will automatically be associated with your user profile. Be sure to always use a fresh incognito browser. Chrome tools has great documentation if they are new to you as well.

Check out our JavaScript tag data collection docs for more details on the power of the Lytics tag.

3. What is LQL?

Lytics Query Language(LQL) is the transformation layer that makes passing arbitrary, non-schema specific data to Lytics. It takes a combination of raw event data to user field data, the Lytics schema. In more advanced implementations complex calculations and aggregations can be performed. An example might be rolling all purchase_total values for individual conversions into a purchase_total_over_time. This is a simple example, but it illustrates the power of LQL.

LQL is very powerful and we recommend working directly with our services team to get the most out of it. We also have a team of LQL experts who can help you get started.

As mentioned in our documentation, LQL can be broken down into 6 key components:

SELECT

Select data to be added to user profiles. Including Maps, Counts, and other complex data types. As an example from our documentation, we can select the custom field current_status in order to unify data across streams. In that case your select statement would be something like:

```
status AS current_status SHORTDESC "Current Status of a User"
```

where status is the key name from the event stream and current_status is the field name that is used across multiple streams. This is a simple example, but it illustrates the power of LQL. It is important to note that you must use the same field name for the same field in order to simplify working with the data inside the Lytics admin.

FROM

The stream to select from. In our example case this would simply be:

```
FROM custom_data
```

INTO

This is USER for all user profiles. (technically you could create other types, such as „account...“).

INTO user

WHERE

Filters out entire records to not be included/analyzed. Bots, Employees, Test data. In our example we might want to filter out test data.

WHERE

```
city = "Winterfell"
```

BY

What field are we going to identify this entity by. This is among the most important aspects of LQL. This defines which data is associated with a user profile. For instance:

BY email OR userid

Means that anytime an email address or userid matches, regardless of its origin, we can assume that is the same user

ALIAS

The alias is primarily an internal value of how to reference this particular LQL statement. When accessing LQL through

ALIAS custom_lql_example

4. Writing LQL

Now that we know the basics of LQL we can put it all together in such a way that our sample data sent can be mapped. Making the end product might look something like:

```
/*
  Custom User Data Example
*/
SELECT
  userid      AS user_id      SHORTDESC "Users ID"
, email      AS email        SHORTDESC "Email Address"
, first_name AS first_name    SHORTDESC "First Name"
, last_name  AS last_name    SHORTDESC "Last Name"
, set(title) AS title         SHORTDESC "Job Title"
, set(company) AS company     SHORTDESC "Company Name"
, phone      AS phone        SHORTDESC "Phone Number"
, age        AS age          SHORTDESC "Age"          KIND INT
, gender     AS gender        SHORTDESC "Gender"
, city       AS city          SHORTDESC "City"
, state      AS state         SHORTDESC "State"
, status     AS status        SHORTDESC "Current Status"
```

FROM custom_data

INTO user BY email OR user_id

ALIAS custom_data_example

All that is left once we have completed writing is to save our LQL file. This file name is completely up to you but should start with 'lql'. Save our LQL before uploading it to our account.

5. Testing LQL

Install Lytics CLI

Now that we have our LQL saved locally using the .lql extension we are ready to test. To test we will use the Lytics CLI tool found on Github.

Once we have installed the CLI tool let's verify it is working properly. In the terminal of your choice run the following commands:

Bash

```
$ lytics -h
```

If you are presented with a help menu you are good to go, if an error comes back or something was not found we reconfigured.

Create a Sample CSV

The Lytics CLI can test LQL in two ways. First, it will read some of the recent events from the target stream and show them. It can also test existing LQL but may present challenges when you are writing new LQL on data that has not been sent to the stream locally. In addition, though not demonstrated here, Lytics watch can also use a JSON file locally in addition to the CSV.

CSV's can be created using a variety of tools. When creating your own CSV from scratch just ensure that the first row is a header and each following row represents an individual user's data.

_created _modified user_id email first_name last_name title company phone age gender city state status
2017-07-27T03:04:04.960952144Z 2017-07-27T03:04:04.960952144Z 1234 jon@castleblack.com Jon Snow King in the North
When saving your CSV ensure the file name matches that of your lql with .csv in place of .lql. For instance if your LQL file is named sample_file.lql, the CSV file should be named sample_file.csv and be located in the same directory. The Lytics CLI will look for CSVs with the same file name as your LQL file.

Activate Lytics Watch

Lytics watch will listen for changes to your LQL file. When that happens it will take the CSV or recent data from your LQL file and the data is then output into your terminal in order to validate that all mappings are working properly.

JSON

```
{
  "_created": "2017-07-27T03:04:04.960952144Z",
  "_modified": "2017-07-27T03:04:04.960952144Z",
  "user_id": "1234",
  "email": "jon@castleblack.com",
  "first_name": "Jon",
  "last_name": "Snow",
  "title": "King in the North",
  "company": "House Stark",
  "phone": "555-555-5555",
  "age": 21,
  "gender": "m",
  "city": "Winterfell",
  "state": "Westeros",
  "status": "planning"
}
```

6. Uploading LQL to an Account

NOTE: Please be sure you are using the correct API token when performing the next operation. To check which API token is associated with your account, run the following command:

Shell

```
$ echo $LIOKEY
```

The value output should match the API token pulled from your account. If this is not the case please revisit the prerequisites section.

Query POST Request

To upload the query you will make a simple POST request to our API. Upon a successful upload you will get a response that looks like the following:

Bash

```
$ https POST https://api.lytics.io/api/query access_token==$LIOKEY < my_custom_lql_file_name.lql
```

7. Importing a CSV

Finally we can import our CSV using a standard cURL command. Be sure to update your file and stream name in the command below.

Bash

```
curl -s -H "Authorization: $LIOKEY" \
  -H 'Content-type: application/csv' \
  --data-binary @your_file_name.csv \
  "https://bulk.lytics.io/collect/bulk/your_stream_name"
```

What's next?

We've covered a basic example of getting custom data into Lytics. This is very much just the beginning of how LQL can be used to get data into Lytics. Check out our technical docs and reach out to our services team for more in-depth training on LQL and data management.

Updated 10 months ago

Acquire New Customers with Lytics

Suggest Edits

Customer acquisition is a core focus for all marketing departments. Traditionally, B2C marketers have relied upon sponsored ad placements. However, these teams often do not have access to key customer data, and thus have to rely upon third-party agencies typically do not have access to personalize the customer experience on site.

There is a better way. First-party data can be extremely valuable for ad targeting on certain networks, and marketers can leverage this data to create highly targeted audiences.

The following is the Lytics Playbook for acquiring new customers.

The framework

There are two important concepts within acquisition marketing as it pertains to Lytics:

Driving traffic: advertise to relevant audiences by using your first-party data.

Generating leads: deliver personalized experiences to new visitors to create leads.

How Lytics helps

Lytics helps marketers to drive quality traffic by unleashing streams of powerful first-party data to ad platforms such as Facebook and Google Ads for building lookalike audiences.

How effective is this? Most companies see their return on ad spend (ROAS) of campaigns on these platforms improve significantly.

Using your first-party data, Lytics helps to improve the retargeting of your past visitors in two main ways: by identifying visitors who are most likely to return.

You can then use Lytics to target personalized experiences on site to encourage the right visitors to sign up for your product or service and visitors who could/should be targeted with offers and content to drive a first purchase.

The playbook

Step 1: Send all customers to ad platforms for suppression

Why would you want to throw valuable acquisition dollars on a wasteful audience of existing customers? The answer is simple (and the answer everyone is) because they don't have a good source for this audience. With Lytics, you can send all customers to Facebook and Google Ads from lead generating advertising.

Effort	Impact	Developer Required
Low	Medium	No

Building the audience

The key to building the right audience is to include all the customers and/or free subscribers you don't want to spend on. This is the first step in the documentation to get started. The following examples skip the first few steps and jump right into building custom audiences.

For most companies, an audience to suppress would resemble something like this:

All customers sample audience

Be sure to suppress active users on all your lists and products. This will give the most comprehensive set of customer data.

Export a continuously updated file

Export these audiences to the key ad platforms: Google Ads, Facebook and, Yahoo Ads.

Build suppression in your ad campaigns

And then create suppression lists in your ad campaigns. See the respective help docs for more information on Facebook and Google Ads.

Step 2: Build lookalikes of your best customers

If you're using lookalikes of your customer file as a targeting audience, it's likely already the best performing audience you can build. Is it the 80/20 Rule? 80% of your revenue comes from 20% of your customers. For most companies, it's more like 90/10.

If you can build lookalikes off your absolute best customers, you'll help Google, Facebook, and Verizon Media to identify your best customers.

Effort	Impact	Developer Required
Low	High	No

Building the audience

In order to help train the big ad platforms as to who your best users are, send the audience over that most resembles people with a high lifetime value (LTV) or multiple purchases. For example, your audiences might look like this:

audience-high-ltv-example

Note that there are several ways to determine high LTV, so feel free to customize to your specific requirements. The goal is to identify your best customers, not lookalikes of all customers.

Export a continuously updated file

Export these audiences to the key ad platforms: Google Ads, Facebook and, Yahoo Ads.

Build lookalikes in your campaigns

Build a lookalike in Facebook or a „similar audience... (lookalike) in Google Ads.

Step 3: Setup data science based remarketing audiences

The next step is to route audiences to the most impactful remarketing campaigns. For example, you may have certain content affinity over another. Additionally, you probably want to consider engagement-based suppressions for remarketing.

Effort	Impact	Developer Required
Low	Medium	No

Building the audience

Think through which content affinities best align to your creatives. For example, break out your core audiences based on the following:

audience-marketer-affinity

...or the user might belong to this group:

audience-technical-affinity

You can and should use „Any Affinity... to start. You'll note that users with any affinity for a topic show above-average engagement, limiting to „High Affinity....

Also, be thinking about the likelihood of a user to re-engage. Typically, your time window of attention is very short. Content Predictive Modeling of interactions may also play a role. Your audience might look something like the following:

audience-score-example

Exporting the audience

Anonymous audiences are continuously exported from the Lytics JavaScript Tag to Facebook and Google Analytics to track user behavior.

Continue with step-by-step instructions to promote relevant content to users based on their interests.

Step 4: Collect emails

As we mentioned at the start of this playbook, acquisition marketers seldom have the ability to personalize web landing pages. One of the best ways to win your acquisition marketing game is to extend the lead generating efforts to your site with targeted email collection.

Effort	Impact	Developer Required
Low-medium	Medium-high	No, but useful for customizing the look and feel

Building the audience

Don't taint the user experience with untargeted email collection. For known users, suppress your email collection efforts. For new users, use a targeted approach.

audience-build-email

Building email collection campaigns

See the Experience editor doc for how to set up this type of campaign in Lytics. The Pathfora SDK allows easy custom personalize-example

Continue with step-by-step instructions on how to grow your email marketing list.

Step 5: Targeted offers

Many prospective customers will need an additional nudge to make a first purchase. The big advantage of using Lytics is that you can combine email, in-app, and ads. This consistent approach between channels frequently provides up to a 10% lift in conversion rates.

Effort Impact Developer Required

Low-medium Medium-high No, but useful for customizing the look and feel

Building the audience

Your audience can be limited to non-customers for special incentives, and may include engagement filters or price sensitivity. This is a common starting audience:

audience-known-non-customers-example

As a final note, keep in mind there are many other areas to cover including copy choices, optimizations, and common mistakes. All of these should be taken into consideration while developing your marketing strategy to acquire new customers.

Updated over 1 year ago

Best Practices for Personalizing Your Ad or Search Landing Page

Suggest Edits

Today's most popular ad platforms, such as Google and Facebook, care a great deal about the overall user experience. They will penalize you if the landing page or associated content. It is imperative that both the content and overall experience be considered when creating ads.

Personalized modals are no exception to this rule. In fact, when not approached strategically, modals may negatively impact the user experience. In the simplest of terms, an auction occurs to determine which ad(s) are served to any given user. Ads with bad experience scores may be eliminated entirely from the candidate pool to make room for ads with more potential.

Google Ad Rank Overview

Additional Information About Google Ad Position and Ad Rank

Facebook Ads Targeting & Positioning Policy

Facebook Low Quality or Disruptive Content Policy

There is, however, a silver lining. Gone are the days of blanket marketing initiatives which force an experience that may not be relevant to the many. With Lytics, you gain an understanding of each individual's past behavior or lack thereof in real-time. This allows you to create ads that are relevant, personalized and positive. Which in turn is the best way to get the highest return from your ad and drive conversions.

The topic of ad and landing page optimization is broad. We simply could not cover all important aspects here. Rather, we provide a high-level overview to keep in mind as you develop your marketing and personalization strategies. In addition, some may find the following examples helpful in identifying sessions or other anomalies in performance.

1. Targeting

Though it may be obvious, considering who you're targeting with modals is extremely important and often overlooked. When a user is visiting a site for the very first time, it is unlikely your experience will be positive if the vast majority of the site that you are targeting is not relevant to them as soon as the page loads.

Example of full page takeover on the second click of a user's first visit before they had a chance to explore the brand's content. neilpatel-purple-popup

Example of half page takeover on user's first visit. neilpatel-popup

Added for comedic value. First page view popup on article talking about discussion one of Google's many announcements. random-popup

Because Lytics allows you to target users based on profile details, audience membership and behavior in real-time, putting a bit more bold with visitors showing strong signs of conversion or history with your brand.

It is all too common on the web today to see companies take a "spray and pray" marketing approach where all visitors are targeted at the beginning of their experience. This is seen as a negative for nearly all search and ad providers, many of which address this.

Google Blog Post Addressing Mobile Popups

2. Modal Size & Position

To build on Targeting, there may be cases where delivering a message at the beginning of a visit is important and acceptable where that message is delivered. Does it cover prime real estate requiring the user to alter their patterns in order to proceed and offer a more efficient path to value while not restricting consumption? There is no silver bullet here but testing has yielded great results.

3. Device

The device being used is often not considered when attempting to locate the root cause of a negative change. Keep in mind that user behavior is typically different from device to device. A modal that is not intrusive on desktop may cover the entire canvas on mobile. If your devices are suffering from a drop off. Follow up by altering the targeting and presentation based on device or consider a more mobile.

Example of cookie acceptance modal on desktop - small and not disruptive.
nextweb-desktop

Example of cookie acceptance modal on mobile - much larger and more disruptive.
nextweb-mobile

4. Time on Site & Session Depth

As we mentioned in our first example where the visitor received an offer immediately upon their first page view, one can offer a variety of display options which allow you to only show the modal after a set amount of seconds, after some amount of time delivered after a visitor has made a connection and consumed enough content in order to make the modal experience more relevant.

5. Inline Personalization

The Lytics web personalization SDK is one method for delivering personalized communications on site. It was designed to address the needs of every use case. As such, the Lytics JavaScript tag and various Personalization APIs can be leveraged in many different content in virtually any situation.

At the end of the day, the user experience is everything and this should be a core focus of all marketing applications and strategies.

Updated 10 months ago

Capture More Information from Qualified Leads

Suggest Edits

Converting a user from unknown to known is a key part of many marketing journeys. You can use Lytics web Experience to capture more information. If you want, you may want to collect a bit more information from your user when they subscribe to your brand.

In this guide, you will learn how to create a widget with a custom form component. The example will offer users to select a topic of interest, a field as well as list of checkboxes that can be used to select which topics to subscribe to.

Custom Form

You can download the complete JavaScript configuration, CSS, and API override command for this example from the Lytics GitHub repository.

Currently in the Lytics UI, you may only choose to include the following fields in your capture leads form:

Email (Email input)

Name (Text input)

Job Title (Text input)

Company (Text input)

Phone Number (Text input)

Message (Text area)

Country (Select box)

However, with the Pathfora SDK, or using API overrides, you can include any custom field of the following types:

Text input

Email input

Text area

Select box

Checkboxes

Radio buttons

JavaScript Configuration

Begin with a configuration for a form widget. Without defining any field customizations, Pathfora will default to using the following elements in this guide.

JavaScript

```
var customFormWidget = window.pathfora.Form({
  id: "custom-form-widget",
  layout: "slideout",
  position: "bottom-left",
  className: "custom-form-widget",
  headline: "Sign up for our Newsletter",
  okMessage: "Subscribe"
});
```

```
window.pathfora.initializeWidgets([customFormWidget]);
```

This config will generate a simple slideout with a form containing some default fields.

Legacy Form

Once you have an idea of what form elements you want to include in your widget, you may use the form builder drag and drop interface to build the form you wish to display. Each field can be marked as required and may have a label and name (used for as the primary key for the field). Text fields may also include placeholder text, and for fields with multiple options you may define the display text and value for each option.

Form Builder

Once you click Save, the builder will output the formElements field that you can simply copy and paste into your widget configuration. This is useful for creating configurations with custom forms.

Form Builder Config

JavaScript

```
var customFormWidget = window.pathfora.Form({
  // widget configuration
  formElements: [
    {
      "type": "text",
      "required": true,
      "label": "Email Address",
      "name": "email"
    },
    {
      "type": "checkbox-group",
      "required": true,
      "label": "Which feeds would you like to subscribe to?",
      "name": "subscription_feeds",
      "values": [
        {
          "label": "Beauty & Perfumes",
          "value": "beauty"
        }
      ]
    }
  ]
});
```



```

    },
    {
      "label": "Electronics",
      "value": "electronics"
    },
    {
      "label": "Fashion",
      "value": "fashion"
    }
  ]
}
});

```

Styling Tweaks

You may style the form using custom CSS. The example in this guide applies some small CSS changes to adjust the s

Remember you can download the code for this example to get the complete JavaScript, CSS, and API override. This c

Custom Form

Once you've tested and are happy with the look and feel of your form slideout, it's time for the most important step: e.

Defining the Lytics Audience

The example widget in this guide will be targeted at an audience of anonymous users with a high intensity. Rememb e widget to.

Alternatively, you may build your widget as a web personalize Experience with the Capture Leads tactic in the Lytics I

Updated 10 months ago

Content Modularization in Email with Lytics Audiences

Suggest Edits

Running a complex nurture program can be difficult to manage, especially if you're using countless static email temp ting language to allow for dynamic user data to be embedded in an email. Dynamic templates are not only useful for s user data such as audience membership and other user profile fields to simplify your email marketing programs al

This guide will walk through how to modularize major components of an email based on Lytics audience membership s applicable to most ESPs that Lytics integrates with, just be sure to refer to the your ESP's documentation for dynam

Before you begin

static email template

You should have a generic HTML email template set up in your ESP. This guide uses an email template promoting ne of shoes the user is interested in. Think about what parts of the email you would like to change based on audience m

Build your audiences

In Lytics you'll need a target audience - the group of users to send the email to - and one or more additional audienc e Promotional List as the target audience, and two audiences built with content affinity, Interested in Dress Shoes an content pieces for.

content-modularization-audience-0621

Sync the audience to your email tool

Once you have your audiences ready, you will sync your target audience to an ESP. This can be done with a Lytics exp . If you're using another ESP browse our list of integrations and the read the instructions for exporting an audience.

Select Promotional List as the target audience to sync. For Campaign Monitor the other two audiences will be record will use in the template to determine what to show and hide for the user.

campaign monitor export

Regardless of the provider you choose you should see the Lytics users in your email tool not long after you click the

Modify your email template

In this example the header of the email will change based on audience membership. For every element we want to p

Users in the Interested in Dress Shoes audience.

Users in the Interested in Athletic Shoes audience.

Users in the Promotional List who have not shown interest in either shoe type (generic promotion).

If a user is interested in both dress shoes and athletic shoes we can prioritize which to show based on the order of th
promote dress shoes in this example for that case.

email modularization

Follow these steps to personalize your email template in this manor:

Open your template in an HTML editor, locate the text, image or HTML element you want to change based on audier

Use the custom LyticsAudience field and conditional logic to bind your HTML content in if/else statements. Take a lo
change the text on the banner of the example email:

HTML

```
<!-- custom headline -->
```

```
[if:LyticsAudiences=Interested in Dress Shoes]Dress shoes for any occasion
```

```
[elseif:LyticsAudiences=Interested in Athletic Shoes]Kick start your summer with new shoes
```

```
[else]Shoes of all shapes and sizes you'll love[endif]
```

Continue to edit your template to modularize the content of your email. In our example, we also switch out the imag
membership:

HTML

```
<!-- custom image -->
```

```
[if:LyticsAudiences=Interested in Dress Shoes]
```

```
[elseif:LyticsAudiences=Interested in Athletic Shoes]
```

```
[else][endif]
```

...

```
<!-- custom CTA -->
```

```
[if:LyticsAudiences=Interested in Dress Shoes]<a href="https://shoepalace.com/products/dress-shoes">Shop Dress S
```

```
[elseif:LyticsAudiences=Interested in Athletic Shoes]<a href="https://shoepalace.com/products/athletic-shoes">Shop
```

```
[else]<a href="https://shoepalace.com/products/shoes">Shop New Shoes</a>[endif]
```

In the next step you will be able to see the rendering of the templates in each of the three contexts.

Send your campaign to the target audience

Create a new campaign in Campaign Monitor. On the Content step upload the HTML template you've created that in
review the email and toggle between the LyticsAudience dynamic content options to view each of the expected outp

campaign monitor preview

On the Recipients step be sure to select the audience you exported to Campaign Monitor from Lytics.

lytics audience in campaign monitor

Continue on to the Delivery step. You can send a test email to yourself and/or schedule your campaign for delivery.

This guide has demonstrated how you can send three distinct, personalized emails to users based on their interests
ur customers, you can send an email that speaks to a user's interests.

This is just the tip of iceberg, think of how you can personalize and modularize your email templates for more advanced use cases, and you can show and hide entire content sections in an email based on audience membership. Your Lytics representative can help you with these use cases. If you have any questions or want to talk through use cases reach out to the Lytics team.

Updated over 1 year ago

Customize your Web Experiences with Branded Images

Suggest Edits

A well designed Experience can really capture a user's attention, and often images are a key component to design. Lytics provides a way to customize images, but out of the box there is not much flexibility - that's where custom CSS can help.

In this guide, you will learn how to create a widget with a stylized image component. Images are a natively supported feature in Lytics, but the default styles for images are rigid. This guide will explore a few different methods on how to customize images for your web experiences.

Custom Image Slideout

You can download the complete JavaScript configuration and CSS, for this example from the GitHub examples repository.

Custom Image Approach

There are three ways you might approach adding an image to a web Experience or Pathfora widget:

Setting a background image with custom CSS.

Using the natively supported image feature with some custom CSS to adjust the size and placement of the image relative to the widget.

Adding an image via inline HTML to the message or headline field.

This guide will discuss the implementation of options 2 and 3. Note that option 3 is only available for widgets built directly in the Lytics Experience editor.

JavaScript Configuration

Using the Default Image Feature

To use the native image capability in Pathfora, simply add the variant and image settings to a basic widget configuration.

JavaScript

```
var imageWidget = window.pathfora.Message({
  id: "image-widget",
  layout: "slideout",
  position: "bottom-right",
  className: "image-widget",
  headline: "Save 10%",
  msg: "24 hour sale on all Apple mobile phone products!",
  image: "assets/iphone-modal.png",
  variant: "2",
  okShow: true,
  okMessage: "Save Now",
  cancelShow: false
});
```

```
window.pathfora.initializeWidgets([imageWidget]);
```

This config will generate a simple slideout, with a headline, message and CTA, and a small, circular image above the text.

Custom Image No Styles

Adding an image in this way is natively supported for web Experiences in the design step of the editor.

Using an Inline Image

You could also present an image within the text content of the widget by adding an img HTML element in the headline or message field.

JavaScript

```
var imageWidget = window.pathfora.Message({
```

```
// widget configuration
msg: "<img src=\"assets/iphone-mini.png\" class=\"pf-widget-img-inline\" alt=\"iPhone\">24 hour sale on all Apple m
});
```

You may want to add a custom class name to this element to help style it with CSS later.

Note: As mentioned above using HTML inline images in this manor is not compatible with web Experiences created t works if you are implementing widgets with the SDK directly.

The complete code example in this guide will contain both a large feature image using the native Pathfora setting, and

Images with no styles

Styling the Image Slideout

With some additional CSS work, you can style images of both types to fit the slideout.

Styling the Default Image Setting

The image generated by the image setting in the config has the class name pf-widget-img. In the CSS, to select this el ensure that your styles will override the default Pathfora styles for images. The example below specifies the pf-widge s name of your widget.

CSS

```
.pf-widget.pf-widget-variant-2.image-widget .pf-widget-img {
  border-radius: 0;
  width: 100%;
  height: auto;
  top: 0px;
  left: 0px;
  margin: 0px;
}
```

These styles make the image span the full width of the widget, and position it at the top.

Next you can adjust the text to appear below the image, by applying a margin to the headline. Remember that CSS s

CSS

```
.pf-widget.pf-widget-variant-2.image-widget .pf-widget-content .pf-widget-headline {
  margin-top: 190px;
}
```

Lastly, we'll need to adjust the "x" button in the corner of the modal to display prominently above the image.

CSS

```
.pf-widget.pf-widget-variant-2.image-widget .pf-widget-img {
  /* existing styles */
  z-index: 0;
}
```

```
.pf-widget.image-widget .pf-widget-close {
  color: white;
  z-index: 1;
}
```

Custom Image Styled

Styling an Inline Image

Now to style the inline image, select the custom class name you defined. The example in this guide floats the image

CSS

```
.pf-widget.image-widget .pf-widget-img-inline {
```

```
float: left;
width: 25px;
margin-right: 20px;
}
```

```
.pf-widget.image-widget .pf-widget-message {
  text-align: left;
}
```

Custom Image Slideout

Remember you can download the code for this example to get the completed JavaScript and CSS. This can act as a starting point for your own implementation.

Defining the Lytics Audience

Once you've tested and are happy with the look and feel of your image slideout, it's time for the most important step: defining the audience.

The example widget in this guide is targeted at an audience of users with a high momentum and some affinity for IP. This is configured with the audience ID you wish to serve the widget to.

Alternatively, as mentioned above, the image feature is natively supported by the Lytics web Experiences. You would use the image feature configuration of your widget.

Updated about 1 year ago

Deliver Targeted Content

Suggest Edits

Lytics makes it possible to recommend content to a user based on their affinity for certain types of content, linking the content to the user's interests.

Lytics content affinity engine automatically scans and categorizes the content that is associated with each page on your site. This content is then mapped against that semantic understanding of your site's content producing a user affinity score for each user.

You can create a Recommend Content Experience by following these steps but the keys to creating an effective recommendation are the audience and the content.

Build a Content Affinity Audience

You can use the Content Affinity tab of the audience builder to build an audience of users who show interest in a particular topic. This audience can be used to target users with the desired Affinity in a personalized Experience.

From your Lytics dashboard select Audience.

Click Create New Audience.

Select the Content Affinity tab.

Select the Affinity you would like to target.

Any Affinity is selected by default. You can adjust the affinity score range to refine the users who will be targeted or scored.

In the Name your audience... field, give your audience a descriptive name.

Select Enable API Access.

In the ID field, enter an audience ID.

Click Save.

content-affinity-audience-builder-data-science

Audience with any affinity for articles about "data science"

This audience can now be used in a Recommend Content (or any other) campaign to target users who have the desired affinity. You can also use the rules and rulesets to create highly refined audiences. You are also not limited to audiences based on content affinity.

Content Collections

Content collections give you the ability to group content together that you can then recommend in your campaigns.

Content collections can be created based on a variety of criteria, such as, recently published, author, hand picked by you, or a combination of any of these criteria. In addition, these collections can be automatically added to or removed automatically when they match or fail to match your criteria. For more information, see Content Collections.

targeted-content-topic-select

Content collection grouping article related to "data science" on learn.lytics.com.

Build Your Campaign

Next, you'll need to navigate to the Experiences tab and pick the "Recommend Content" tactic for Lytics. The first step is to select the collection you want to recommend content from.

Next, select whether you want this to be based on a user's interests ("Highest Affinity"), content or product freshness ("Last Interaction"):

Screen Shot 2021-06-07 at 11.23.57 AM

Don't forget to give your Experience a name. See our [Experience editor documentation](#) for step-by-step instructions.

Updated over 1 year ago

Drive Email Capture & Engagement with Content

Suggest Edits

Content recommendations are a powerful, direct method for driving engagement of anonymous users with the goal to present personalized ad sets through marketing platforms like Google Ads and Facebook. You can take the personalerve content based on each user's personal audience membership and content affinity.

Leveraging user behavior

Lytics uses data science scores to move users into preconfigured audiences based on their behavior. This use case re-identify those who are highly engaged with your content generally and, through further refining, individual content items.

Building your target audience

Users can be members of preconfigured audiences determined by a combination of**characteristics .

Anonymous Deeply Engaged

For this use case we want to combine the characteristic Email Capture Status: Unknown email with the behavioral audience: Deeply Engaged. We will target users without email addresses, i.e. anonymous users, who are deeply engaged with your content. What we have here is a combination of two different ads or emails based off the content they engage with the most.

image

Using fields to refine your audience

While all the users in your Anonymous Deeply Engaged audience are engaged with your content different behaviors over a long time period while others may have consumed content over a relatively short period of time. Using the frequency score, you can identify longtime anonymous, engaged users in effort to convert them to know users. Similarly, the recency score can be used to identify users who have recently engaged with your content in effort to convert them.

Activating Your Audience

Your Anonymous Deeply Engaged audience is a great target audience for site personalizations, content recommendations,

Delivering targeted content

To deliver the content most relevant to a user's interest you can build additional audiences based on Anonymous Data sources: Content Affinity to learn how to add content affinity rules to new and existing audiences.

Driving email capture

Your Anonymous Deeply Engaged audience is an excellent target to capture email address and convert users to a known user. See [this](#) documentation to learn how.

Individualized content recommendations

Lytics offers a Content API that provides recommendations based on user content affinity. Our Developer Academy and our Content API documentation provide guidance on delivering content recommendations programmatically. These documents are targeted for developers and may provide some guidance.

Updated 10 months ago

Drive Mobile App Downloads With Lytics

Suggest Edits

A common marketing use case to improve mobile engagement is to create journeys that drive downloads of your brand's mobile app. You can build audiences of anyone who has previously browsed your sites using a mobile device since the Lytics JavaScript SDK tracks mobile devices. You can then use Lytics, Google AdWords, and Facebook, as well as other ad platforms, for driving mobile app installations.

Building the audience

From the Lytics dashboard, select Audience and then Create New Audience. Using a Custom Rule, build an audience

Screen Shot 2019-03-05 at 12.39.34 PM

This audience will include anyone who has reached your site from a mobile device.

Note: You do not need any identifying information for this audience. Be sure to select the Enable API Access checkbox on a mobile device.

Targeting the audience

Now you can build campaigns in your chosen marketing platforms to effectively target this audience of mobile web users. If you are using Facebook, follow these steps to target with Facebook ads. Upon completion, your advertising team or agency can launch the campaign.

Updated over 1 year ago

Engage Qualified Users with Targeted Ads

Suggest Edits

Using Lytics audiences you can identify anonymous traffic with higher potential to engage and/or convert. A powerful use case is Google Ads, driving them to further interact with your brand.

Leveraging user data

This use case will provide a few suggested audience definitions to support your retargeting effort. These audiences will be built using

Lytics preconfigured audiences such as characteristics and behavioral audiences.

Commerce user fields such as Lifetime Value (LTV) or Lead Status.

Lytics data science scores such as frequency, recency, and intensity.

Note: The Lytics StartSmart data schema supports a standard set of commerce fields including the two used in this use case. You can also map your own fields to the standard schema. If you've imported data from Salesforce you should have access to the standard schema.

Building your target audiences

Using the building blocks mentioned above you can build one or more of these suggested audience to use in ad campaigns.

Anonymous Deeply Engaged

This audience combines the characteristic Email Capture Status: Unknown email with the behavioral audience Lytics uses the intensity score, and is a great way to qualify anonymous traffic for targeting.

anonymous-deeply-engaged

Known High LTV

Combine the Email Capture Status: Known email with the User's Lifetime Value field using the "AND" rule to create a high quality audience. This can be a great source audience to generate a lookalike audience.

known-high-ltv

Known Leads

For B2B marketing you can substitute LTV with a field like Lead Status to build an audience of open leads for retargeting.

known-leads

Refining your audiences with user fields

Want to further narrow your audience to only the most qualified users? Try experimenting with additional rules using Lytics user fields. For example, you can use the recency score to target users who recently interacted with your brand. These qualifiers can help you create a further refined audience.

Activating your audience

Once you've built your audience(s) you can export them to your ads platform for use using Lytics' built in integration with Facebook and Google Ads.

Retargeting anonymous users

Anonymous users can be advertised to directly in Facebook. You can access the Anonymous Deeply Engaged audience you built in Lytics and use it in your Facebook ads. You can also target anonymous users in Google platforms such as Adwords through our JavaScript integration with Google Analytics.

Reach new, qualified users with Lookalike audiences

Known users can be targeted across both Facebook and Google, as well as drive lookalike audiences. You can export audiences and Google Adwords remarketing lists.

To power lookalike audiences which will find users similar to your best customers, read more on creating a lookalike

Updated over 1 year ago

Enhance Personalized Messaging with User Profile Fields

Suggest Edits

Targeting Lytics audiences with web Experiences allows you to achieve a level of personalized messaging to a group. You can further personalize your Experiences at a one-to-one level by including specific user profile fields in the message.

In this guide, you will create a bar widget containing a coupon code unique to the user using entity field templates. Then, you will add it in the widget.

Bar with Coupon Code

This technique can be used to include any user field inline within a widget in the headline, message, or CTA link of an Experience. You can use their first and/or last name if the data is available in Lytics.

You can download the complete JavaScript configuration and API override command for this example from the GitHub repository. It is created by web personalize Experiences in the Lytics UI.

JavaScript Configuration

Begin with a generic widget configuration containing a static coupon code in the message field.

JavaScript

```
var entityTemplateWidget = window.pathfora.Message({
  id: "entity-template-widget",
  layout: "bar",
  position: "bottom-fixed",
  className: "entity-template-widget",
  msg: "Use code <strong>FREE2019</strong> for free shipping on your next purchase!",
  okMessage: "Get Free Shipping!",
  cancelShow: false
});
```

```
window.pathfora.initializeWidgets([entityTemplateWidget]);
```

This config will generate a bar widget with a CTA. You can use the `confirmAction` callback to redirect the user to a checkout page. You can use the `confirmAction` callback to redirect the user to a checkout page.

JavaScript

```
var entityTemplateWidget = window.pathfora.Message({
  // widget configuration
  confirmAction: {
    callback: function () {
      window.location.href = "https://yourwebsite.com/checkout?code=FREE2019";
    }
  }
});
```

Coupon Applied

To customize this coupon code per user, you can access a field by the field name (in this example, `offer_code`) from the user profile.

Fields that you wish to use in the template of your widget must be surfaced first.

JavaScript


```

var entityTemplateWidget = window.pathfora.Message({
  id: "entity-template-widget",
  layout: "bar",
  position: "bottom-fixed",
  className: "entity-template-widget",
  msg: "Use code <strong>{{offer_code}}</strong> for free shipping on your next purchase!",
  okMessage: "Get Free Shipping!",
  cancelShow: false,
  confirmAction: {
    callback: function () {
      window.location.href = "https://yourwebsite.com/checkout?code={{offer_code}}";
    }
  }
});

```

window.pathfora.initializeWidgets([entityTemplateWidget]);
 Bar with Coupon Code

You may use this bracket notation in the Choose URL step of the Experience editor. So there is no need to make API

You may be wondering, what happens when the current user does not have the field offer_code on their profile? The

Simply ensure that the targeting audience requires that the field in question exist.

You can define a fallback value. This is a default static value to be shown if offer_code does not exist on the user profile.

Alternately, you may set the display condition showOnMissingFields via the JavaScript configuration if you are using the

Testing the Personalized Widget

Testing widgets that contain user profile fields can be tricky. If you are using the Pathfora SDK, it may help to use the Pathfora SDK during the development process.

JavaScript

```

window.pathfora.customData = {
  "offer_code": "FREE2984712312"
};

```

Alternatively, add the offer_code field to your current profile by sending an API request to the data stream which contains the user ID (usually _uid).

You can use the jstag.getid() function in the developer console of your website which has the Lytics JStag loaded to get the user ID.

Here is an example API call to add the offer_code field via the default stream.

Shell

```
curl -XGET http://c.lytics.io/c/{lytics_account_id}/default?offer_code=FREE324234234&_uid={{uid_here}}
```

This example makes a request to the Lytics collector API for your account. If offer_code is mapped in the LQL for the user profile, it should add the value FREE324234234 to the user profile. Once you see it show up in your user profile in the Lytics user interface, you are done with the new field value.

Entity Field

Define the Lytics Audience

The example widget in this guide will be targeted at an audience of high LTV customers who have the user profile field LTV. You wish to serve the widget to if you are using the Pathfora SDK.

Updated over 1 year ago

Grow Your Email Marketing List

Suggest Edits

Email newsletters are a valuable tool to share content, product features, or announcements with your customers. Reach out to interested and valuable customers your brand has. A Lytics personalized collect lead campaign can assist in growing your email list and engage.

Create an Audience of High Activity Users

Effective campaigns begin with an effective audience. The goal here is to collect email addresses of visitors to your site who are a high activity audience combines multiple rules to target users with a quantity score of at least one whose email is not known. These users are on your site and are not great candidates for engagement with an email mailing list. You also don't want to present them with offers that are not limited to this criteria. You can target any audience who you think would be valuable to engage with via email.

grow-email-audience

Create Collect Lead Campaign

Once you have an audience of users you would like to target use it when building your collect lead campaign. The key is to make sure you are reaching out to your users. You may only want to solicit email addresses from a particular section of your site and only display the campaign on your site.

grow-email-display-options

Export Leads to Email Service Provider

Once your new campaign collects email addresses you can use your campaign to build a new audience. A user who completes a campaign can be exported to your ESP and added to your email list. The specific export instructions for your ESP will vary, see our documentation for more details.

grow-email-audience-converted

Updated over 1 year ago

Import an Audience from your Data Warehouse

Suggest Edits

In general, audiences should be created in Lytics using the audiences builder, then pushed to actioning platforms via integrations. This process can be automated to keep downstream tools in sync.

However, in some cases, there are more narrow and targeted audiences of users who you need to target for one-off campaigns. This can be achieved by creating the audience in your data warehouse then pushing that group of users to Lytics.

Examples of audiences to build in Lytics:

Users who have visited the site in the last week

Users who purchased a product within the last three days

Users who opened an email and have not logged in yet

Examples of one-off audiences to push from an external querying tool:

Users who attended event A and purchased a product over \$100 two quarters ago

Users who visited a booth during conference A and, on average, ordered at least three products per month over the last year

Users who should receive a promotional offer based on a list provided by an outside agency

Pushing a one-off audience to Lytics

In this example, we'll use Google BigQuery as our data warehouse and querying tool.

Once notifying your account manager, your Lytics instance will be configured to pull data from a table called lytics_custom_audience_push.

customer_id (string)

custom_audience_name (string)

custom_audience_value (string)

Instructions

Create a table called lytics_custom_audience_push with the above schema.

Write your query. The query output must have three columns: the audience name, the audience value, and the customer ID.

Under query settings, set lytics_custom_audience_push as the destination table for your query results.

Lytics pulls data from lytics_custom_audience_push on a rolling basis, so you should see the custom_audiences field in the Lytics interface.

Example

Let's say you need to push an audience of VIP users selected for targeting based on their purchase history and attention.

SQL

```
SELECT user_id,  
       "vip_users" AS custom_audience_name,  
       "true" AS custom_audience_value  
FROM user_db u  
     JOIN event_db e  
       ON e.id = u.event_id  
WHERE u.purchase_avg > 50  
      AND u.purchase_num > 5  
      AND e.id IN ( 748, 573, 934 )
```

Your lytics_custom_audience_push table should look something like this:

Screen Shot 2019-05-06 at 4.19.00 PM

After this query is pushed to the lytics_custom_audience_push table, you can use it in the Lytics audience builder under Audience Builder > Custom Audiences.

Screen Shot 2019-05-06 at 3.57.38 PM

Notes

The lytics_custom_audience_push table is configured to accept strings as values. If you need to push a different data type, you'll need to create mappings for integers, floats, and timestamps.

One-off audience pushes are typically used for one downstream campaign. If you find yourself repeatedly targeting the same audience, you can create a custom audience and reconfigure it as a user attribute.

If you are using a different data warehouse, you can send the table as a CSV to an SFTP server or S3 bucket for Lytics to ingest.

Updated over 1 year ago

Improve Ad Campaign Metrics with Predictive Targeting

Suggest Edits

Lytics predictive audiences can be leveraged in advertising tools to make intelligent targeting and bid strategy decisions. By using predictive audiences in your ad tools as seed audiences for lookalike modeling, this use case specifically uses Lytics data science to identify users most likely to complete an action instead of anyone who may be eligible for a remarketing campaign.

Build a Predictive Audience in Lytics

Leverage Content Affinity in Audiences

Custom Predictive Audiences

Build a Predictive Audience in Lytics

There are several ways to build your audience in Lytics that you can utilize in your ad platform for targeting. To begin with, you can create a deterministic rule for who you'd like to retarget in your ad strategy. This most commonly looks like:

„Has done event X but not Y....

„Has web activity but is not a paying subscriber....

It's likely that you have rules like this already set up in your ad tools. To improve the performance metrics of these campaigns, you can create a predictive audience that is a subset of the audience size available for targeting but increase the likelihood of conversion against your business goal. This will impact your reach, but it will also increase your conversions, impressions but increasing conversion rate and/or decreasing cost per acquisition.

To do this, layer in one of the Lytics out-of-the-box behavioral audiences that use data science, such as Engagement: High.

build predictive audience

*This rule set includes all deeply engaged users who haven't made a purchase.

By passing this audience directly into an ad channel, your remarketing efforts will be more refined to only include users who are most likely to convert, instead of your entire total user base.

Leverage Content Affinity in Audiences

Lytics can also create data-science based audiences specific to a topic of interest for an individual user. This logic is based on the assumption that what they have shown past interest in because it evaluates changes in topics in real-time and assesses a user's interest in a topic based on their intent and their actions. Read more about user-level topic affinities here.

You can utilize a content-affinity audience for remarketing as a single rule for an audience, or as part of a rule set as a layer in a more complex audience.

content affinity audience

This rule set includes users with a high affinity for 'Chicken' and who have their membership on pause.

By using content affinity-based audiences for remarketing in ad channels, your ad copy and creative can be strategic to users they have previously shown an interest in.

Custom Predictive Audiences

The last strategy for predictive audiences within Lytics is to rely on our proprietary SegmentML product to do lookalikes. If you are familiar with lookalike modeling within ad channels to extend your reach. You can also mine your own first-party data to create audiences within an ad channel.

Your Lytics Account Manager or Services team can help you create predictive audiences within your account, which can be used to create a new audience built within Lytics. See the Integrations documentation for provider-specific information.

segml audience

This summary report shows the growth of a custom predictive audience over a week.

While these audiences may be smaller than the lookalikes you'll get back from Facebook or Google Ads directly, because they are showing signals towards behaving like your seed audience, they will have a higher conversion rate that can be influential in your marketing.

Updated over 1 year ago

Increase Conversions with Lytics and Facebook Lookalike

Suggest Edits

With Lytics audiences and Facebook lookalike you can increase conversions by targeting new prospects that have similar characteristics to your source audience. This guide covers how you can create a source audience of your top lifetime purchasers in Lytics and use it in a Facebook lookalike campaign to reach people who are likely to become purchasers.

Build an Audience of High Value Purchasers in Lytics

For purchase data, this example uses an account that has been connected with Shopify data via Lytics' Shopify import. You can use any method such as the Javascript Tag, CSV, or one of our other commerce integrations. When building your audience to target high value purchasers.

Follow the steps outlined in this guide to build source audiences in Lytics. The steps that follow in this guide provide a step-by-step guide to building a source audience.

In the audience builder, select Custom Rule.

If you have imported data via Shopify, search for and select the field Lifetime Order Price. This field contains the sum of all orders a user has placed in their lifetime order price.

From the rule options Lifetime Order Price must be at least. In the These values textbox enter a price threshold to filter for high value purchasers.

Click Add Condition.

Verify the user count of your audience. Facebook expects the source audience to be anywhere from 1,000 to 50,000 users. Adjust your rules to generate an audience of the appropriate size and specificity.

NOTE: If you're unable to generate an audience of appropriate size with the Lifetime Order Price field, consider creating a list of specific products instead.

Click Create to save your audience.

Using your Lytics Audience in a Lookalike Ad Campaign

Follow through with the steps below to use your high value purchasers audience for the source of a Facebook lookalike campaign.

Export to Facebook

Create a Facebook Lookalike Audience

Assign your Lookalike Audience to an Ad Campaign

Launch your high value ad

Updated over 1 year ago

Keep Visitors Engaged with Content Recommendation Experiences

Suggest Edits

Lytics content recommendations are a powerful tool for personalization. They enable you to engage your visitors with personalized Experiences support content recommendations natively, but with API overrides or the Pathfora SDK, you can create a custom widget.

In this guide, you will build a recommendation modal to reel users back in when they are thinking about leaving your website. The modal will use CSS and additional display fields such as the name of the author and the publication date.

Content Recommendation Modal

You can download the complete JavaScript configuration, CSS, and API override command for this example from the [GitHub repository](#).

JavaScript Configuration

Start with a basic message modal configuration. The goal of this Experience is to entice users to stay on your website by showing a modal to only show the modal when the user is about to leave the page.

JavaScript

```
var contentRecWidget = window.pathfora.Message({
  id: "content-rec-widget",
  layout: "modal",
  className: "content-rec-widget",
  headline: "Wait! Before you go...",
  msg: "... We think you may like this. Maybe check it out before you leave.",
  okShow: false,
  cancelShow: false,
  displayConditions: {
    showOnExitIntent: true
  }
});
```

```
window.pathfora.initializeWidgets([contentRecWidget]);
```

Modal with no Recommendations

Before you proceed, you will need to create a content collection. See the [documentation on content collections](#) to learn how to create content collections for recommendations for instructions on how to get the ID of the collection, which you will need in the configuration for the modal that have been published within the last year.

Now to set up the recommendations in the config. Add the recommend and variant settings to your config. Then add the `visited` field to ensure that Lytics recommendations only return content that the user has not already visited.

JavaScript

```
var contentRecWidget = window.pathfora.Message({
  // widget configuration
  variant: 3,
  recommend: {
    visited: false,
    collection: "{your_collection_id}"
  }
});
```

Basic Recommendation Modal

Now you will have a basic modal with content recommendations displaying on exit intent. Since the example display was published and the author name. To do so, you can add the display object to your configuration.

JavaScript

```

var contentRecWidget = window.pathfora.Message({
  // widget configuration
  variant: 3,
  recommend: {
    visited: false,
    collection: "{your_collection_id}",
    display: {
      date: true,
      author: true
    }
  }
});

```

You can even add some additional settings to style the date, and extend the amount of text shown in the article description.

JavaScript

```

var contentRecWidget = window.pathfora.Message({
  // widget configuration
  variant: 3,
  recommend: {
    visited: false,
    collection: "{your_collection_id}",
    display: {
      date: true,
      author: true,
      dateOptions: {
        weekday: "long",
        year: "numeric",
        month: "long",
        day: "numeric"
      },
      descriptionLimit: 220
    }
  }
});

```

Unstyled Content Recommendation

Check out the Pathfora content recommendations documentation to see the full list of settings you can apply to customize your content recommendations.

Styling the Recommendation Modal

In this example, we will make some small adjustments to soften the look of the content recommendations. You can customize the look of the recommendation modal by adding custom styles to the modal.

Content Recommendation Modal

When writing your own custom styles, remember that different users will see a different pieces of content based on their location in relation to the text. Images from different articles may be of a different size, and text descriptions may vary in length.

Be sure to test your modal with multiple different content items. You can do this by adding the shuffle setting to your modal. You may also want to ensure that all content in your collection has similar image sizes if you do want to adjust the modal.

Content Recommendation Modal Alternate Content

Define the Lytics Audience

The example widget in this guide will be targeted at users in the Likely to Reengage characteristic. Remember to update the widget to target the correct audience.

Alternatively, you may build your widget as a web personalize Experience in the Lytics UI with the Recommend Content widget. See the Github Repository for the exact override command.

Updated over 1 year ago

Leverage Lookalike Models and Predictive Audiences

Suggest Edits

When you need to reach new users that look like your best customers, create Predictive Audiences powered by Lytics. Predictive Audiences help you drive higher customer engagement, drive higher conversion rates, and reduce ad spend.

lytics-lookalike-models

Using Predictive Audiences can benefit marketers in a number of ways:

Helps you define better targeting criteria based on behavioral data rather than demographics or third-party data.

Enables you to optimize each audience for reach or accuracy based on campaign goals.

Deliver use cases that drive conversions across the customer lifecycle.

Better targeting criteria

Predictive Audiences simplify your targeting strategies by eliminating the need to manually define parameters to solve the question, "how do I know what to target users on?" Each Lookalike Model in Lytics has a Feature Importance chart that shows the attributes that drive conversions for your source and target audiences. These attributes include Lytics behavioral scores, content affinities, and other user data.

model-feature-importance-learn-use-case

In the example above, Lytics created a Lookalike Model to understand which of our documentation site users are most likely to become repeat purchasers. As you can see in the chart, our model determined that "Total Pageview Count", "Quantity", and an affinity for "marketing content" were the most important factors in becoming deeply engaged. Your Lookalike Models may simply confirm assumptions, but in other cases, they can offer new insights that may influence desired actions.

Optimize based on goals

Once a Lookalike Model is built, you can easily create different Predictive Audiences to optimize the balance between reach and accuracy for your campaign. When reach is a priority, you can build Predictive Audiences with higher reach but lower accuracy to target a larger audience. You can also build Predictive Audiences with higher accuracy but lower reach to target only the users most likely to convert.

high-accuracy-low-reach-summary

For example, if you're targeting single purchasers to encourage them to become repeat purchasers, accuracy may be more important than reach. The trade-off between accuracy vs. reach.

Use cases across the customer lifecycle

Predictive Audiences help you engage and convert customers across their lifecycle, from anonymous users visiting your website to known users that are brand advocates. Below are common examples of how to build Lookalike Models that power Predictive Audiences.

Earlier stages:

Which of my anonymous users are most likely to convert to known users?

Which of my users with free trials are most likely to convert to subscribers?

Mid stages:

Which of my single purchasers are most likely to become multi-purchasers?

Which of my highly engaged users are most likely to become subscribers?

Later stages:

Which of my subscribers are most likely to churn?

Which of my engaged users are likely to make a purchase soon and do not require additional marketing?

Updated over 1 year ago

Lookalike Models: Conserve marketing spend on engaged users

Suggest Edits

Determining which users to exclude from a campaign is equally as important as finding the users you want to target. If you know which users are unlikely to purchase soon, you can reduce marketing spend by suppressing those users from your campaign audiences.

Define Source and Target Audiences

The first step to building a Lookalike Model is selecting your source and target audiences. Start by defining the desired audience with a pending order.

The source audience is the group that you want to find similar users from, focusing on those who've shown signals to your built audience lytics_deeply_engaged or build your own audience based on your criteria for engagement.

Build Lookalike Model

In the Lytics UI, navigate to the Laboratory section and click Create New Model at the top right. On the configuration page, select the source and target audiences previously defined.

lookalike-model-configuration-engaged-users-pending-order

By default, the Auto Tune option is turned off. If you're not getting enough performance out of your models, try Auto Tune. Select Model Training Only if you're in an exploratory phase and not ready to target users based on this model's predictions. For more information, see the Auto Tune documentation.

Depending on the sizes of your source and target audiences, your Lookalike Model may take a few minutes up to a few hours to reach a pending status until the model is complete.

Create Predictive Audience

Once your model is built, view the Model Summary to determine if it's ready for use in your targeting. You must Activate the model. The model will evaluate users and write the model's prediction scores to user profiles. This process can take up to three days.

lookalike-model-summary-engaged-users-pending-order

On the summary page, click Create Predictive Audience in the Model Usage section, which will open the audience builder. The default decision threshold of 0.5 is used, but you can adjust the decision threshold as needed to reach more users or be more precise in targeting a purchaser without additional marketing.

lookalike-deeply-engaged-pending-order

Next Steps

After creating an audience of users likely to make a purchase soon, you may want to add these users to a suppression list.

Updated over 1 year ago

Lookalike Models: Convert anonymous users to known users

Suggest Edits

Converting unknown users to known users is a key starting point for many marketing campaigns. Collecting data from a mobile app, website, or other source can help you intelligently target anonymous users who are most likely to convert.

Define Source and Target Audiences

The first step to building a Lookalike Model is selecting your source and target audiences. As a marketer, this starts by defining the source audience. In this case, "known users". You can create this target audience in Lytics for users with identifying field(s) such as email, phone number, or name.

The source audience is the group that you want to find similar users from, focusing on those who've shown signals to your built audience lytics_deeply_engaged or build your own audience based on your criteria for engagement.

Build Lookalike Model

In the Lytics UI, head to the Laboratory section and click Create New Model at the top right. On the configuration page, select the source and target audiences previously defined.

lookalike-model-configuration-anonymous-known

By default, the Auto Tune option is turned off. If you're not getting enough performance out of your models, try Auto Tune. Select Model Training Only if you're in an exploratory phase and not ready to target users based on this model's predictions. For more information, see the Auto Tune documentation.

Depending on the sizes of your source and target audiences, your Lookalike Model may take a few minutes up to a few hours to reach a pending status until the model is complete.

Create Predictive Audience

Once your model is built, view the Model Summary to determine if it's ready for use in your campaign. You must Activate the model. The model will evaluate users and write the model's prediction scores to user profiles. This process can take up to three days.

h will evaluate users and write the model's prediction scores to user profiles. This process can take up to three days

lookalike-model-summary-anonymous-known

On the summary page, click Create Predictive Audience in the Model Usage section, which will open the audience builder. A default decision threshold of 0.5 is used, but you can adjust the decision threshold as needed to reach more users or be more precise.

lookalike-anonymous-to-known-predictive-audience

Next Steps

After creating a Predictive Audience from anonymous users most likely to convert to known users, you may want to target them with ads, or emails to drive sales. Learn more about how to acquire new customers using Lytics.

Updated over 1 year ago

Lookalike Models: Convert single purchasers to multi-purchasers

Suggest Edits

Once you've acquired customers, a good way to increase sales and revenue without increasing advertising cost significantly is to reach out to a second or third purchase. Reaching out to these customers with the appropriate messaging can be the first step in increasing their lifetime value.

Define Source and Target Audiences

Begin building a Lookalike Model by selecting your source and target audiences. The desired outcome for the target audience will be "single purchasers" who have shown that they are likely to convert and become repeat purchasers.

Build Lookalike Model

In the Lytics UI, go to the Laboratory section and click Create New Model at the top right. On the configuration page, select Single Purchasers as the source audience and Multi-Purchasers as the target audience.

lookalike-model-configuration-single-multi-purchasers

By default, the Auto Tune option is turned off. If you're not getting enough performance out of your models, try Auto Tune. Model Training Only if you're in an exploratory phase and not ready to target users based on this model's predictions. For more information, see the documentation.

Depending on the sizes of your source and target audiences, your Lookalike Model may take a few minutes up to a few hours to complete. You will see a progress status until the model is complete.

Create Predictive Audience

Once your model is built, view the Model Summary to determine if it's ready for use in your campaign. You must Activate the model. h will evaluate users and write the model's prediction scores to user profiles. This process can take up to three days

lookalike-model-summary-single-multi-purchasers.png

On the summary page, click Create Predictive Audience in the Model Usage section, which will open the audience builder. A default decision threshold of 0.5 is used, but you can adjust the decision threshold as needed to reach more users or be more precise. tional purchases.

lookalike-single-to-multi-purchaser

Next Steps

After creating a Predictive Audience of single purchasers most likely to become multi-purchasers, you may want to target them with ads, or emails to drive sales.

Updated over 1 year ago

Lookalike Models: Determine which subscribers are likely to churn

Suggest Edits

Retaining customers is more important now than ever, and for businesses that have a subscription model, reaching out to at-risk customers for retention and customer engagement.

Define Source and Target Audiences

The first step to building a Lookalike Model is selecting your source and target audiences. The target audience in this example is defined based on the users' last engagement time or some other measure of engagement. The source audience is defined based on the subscription field(s) from your imported data.

Build Lookalike Model

In the Lytics UI, head to the Laboratory section and click Create New Model at the top right. On the configuration page, select the target audience you previously defined.

lookalike-model-configuration-subscribers-churn-risk

By default, the Auto Tune option is turned off. If you're not getting enough performance out of your models, try Auto Tune. Model Training Only if you're in an exploratory phase and not ready to target users based on this model's predictions. For more information, see our documentation.

Depending on the sizes of your source and target audiences, your Lookalike Model may take a few minutes up to a few hours to reach a status until the model is complete.

Create Predictive Audience

Once your model is built, view the Model Summary to determine if it's ready for use in your targeting. You must Activate the model. Lytics will evaluate users and write the model's prediction scores to user profiles. This process can take up to three days to complete. lookalike-model-summary-subscribers-churn-risk

On the summary page, click Create Predictive Audience in the Model Usage section, which will open the audience builder. The default decision threshold of 0.5 is used, but you can adjust the decision threshold as needed to reach more users or be more precise.

lookalike-subscriber-churn

Next Steps

After creating a Predictive Audience, you may want to target this audience with emails to help them re-engage with your brand or with other marketing campaigns.

Updated over 1 year ago

Mobile Messaging with Lytics Webhooks and Serverless Functions

Suggest Edits

Lytics webhooks paired with serverless functions are a great way to build custom workflows to power your marketing campaigns. This example uses a trigger for a variety of platforms to send emails, push notifications, and alerts. This example uses a serverless cloud function to send a text message when they become highly engaged via Twilio.

Setup the audience

In this example, you're going to send a personalized text message to a user when they become highly engaged determined by Lytics data-science scores. Therefore, you are going to need to create an audience of users who are highly engaged based on Lytics data-science scores. Second, their profile has a phone number. And third, they are not yet subscribed to your service.

In the audience builder, you're going to use an existing audience for highly engaged users and two custom rules for filtering audiences, check out the audience builder docs.

NOTE: Twilio's API requires phone numbers with a country code (e.g. +11234567890). If phone numbers are not stored with a country code, you will need to be used to correctly format the number.

Screen Shot 2019-01-16 at 1 26 28 PM

Create the cloud function

This example uses Google Cloud Functions to power this custom workflow, however this will work just as well with other serverless functions. For more information on how to get started on Google Cloud Functions see Quickstart: Using the gcloud Command-Line Tool.

The Lytics webhook sends the full profile of a user when they enter or exit the audience. The code below receives the profile and uses it to send a custom text message to the user.

Note: You'll also need a Twilio account. When using a Twilio free trial, you can only send text messages to the number you provided during sign-up.

```

// Twilio Client
const accountSid = process.env.ACCOUNT_SID;
const authToken = process.env.AUTH_TOKEN;
const client = require('twilio')(accountSid, authToken);

exports.sms = (req, res) => {
  const r = req.body;

  // Do not send a text message when the user exits the audience
  if (r.data.segment_events[0].event === "exit") {
    res.status(200).end();
    return;
  }

  const from = 'your twilio number';
  const to = r.data.phone_number;
  const body = `Hey ${r.data.first_name} | | 'there', we thought you might be interested in subscribing. As an added b
uid=${r.data._uid}`

  // Send the message
  client.messages
    .create({ from, to, body })
    .then(message => console.log(message.sid))
    .then(res.status(200).end('success'))
    .catch(err => res.status(500).end(` error: ${err}`));
}

```

Next, deploy the function. You can get your account_sid and auth_token from the Twilio dashboard.

```
gcloud functions deploy sms --set-env-vars ACCOUNT_SID=<your_account_sid>,AUTH_TOKEN=<your_auth_token> --r
```

After deploying the cloud function, you will be provided with an endpoint that triggers the function. You can also find

Screen Shot 2019-01-16 at 1 28 48 PM

Create the Webhook

Next, you need to export the audience previously created. Return to Lytics and navigate to Data > Integrations and s
bhook, select the audience created previously moving it to the right column, enter the url trigger for your cloud func
ould received a message that your webhook was successfully created. When a user enters or exits the selected audie
L, triggering the cloud function.

Screen Shot 2019-01-16 at 1 30 32 PM

That's it! You've successfully created a webhook trigger that will automatically send a text message to unsubscribed u

Updated over 1 year ago

Optimize Remarketing Spend

Suggest Edits

Remarketing is considered to be one of the most efficient channels available for return on ad spend. If done correctl
st advertisers remarket wastefully. If your remarketing partners were more efficient you'd have more dollars to purs

Lytics can quantify recency, depth of visit, likelihood to return, likelihood to transact, and other metrics based on pre
factors such as categories browsed, content affinities, and whether or not someone is a subscriber or customer. Usin
r investment in Lytics within days or weeks.

Build an Effective Remarketing Audience

Remarketing to users most likely to transact starts by building an audience that includes the users who show the gre
rtisers have had success measuring engagement with a semi-arbitrary metrics such as 3 or more pageviews but Lytic
cs data science scores, in this case quantity. Similarly, using Lytics recency score has been more effective than a fixed

Using the Lytics audience builder, you can build an audience based on two custom rules. The goal is to produce an audience that is highly engaged and active. To do this, you can add a custom rule based on score_quantity, setting the value that includes the top ~15% of your total audience. Second, you can add a rule based on recency, setting the value that includes the top ~10% of your total audience. You should play with these numbers to find the audience that makes the most sense for your business. An audience of users who are more deeply engaged while a higher recency score will target users who have been active recently.

remarketing-audience

Finally, if you're really looking to get the extra mile, you can avoid spending on remarketing to users that you can already reach through other channels, such as email. To filter users with known email addresses, add a new rule based on email_address and toggle Excluded.

remarketing-audience-filtered

Export Remarketing Audience to Ad Networks

Your remarketing audience can be exported to Facebook, Google Ads, or other ad networks for use in your advertising campaigns.

Updated over 1 year ago

Personalize the Messaging of your Website Based on Audience

Suggest Edits

Lytics web Experiences are a great way to quickly spin up a modal or slideout on your website with personalized messaging. You can take this a step further by tailoring inline elements on your web page towards different Lytics audiences using the Pathfora SDK.

In this guide, you will learn how to implement inline content modularization which will show and hide different pieces of content based on user attributes. This example will provide content for following audiences: high momentum users, first time visitors, and users who have visited the site before.

Inline Content Modularization diagram

You can download the complete HTML code for this example from the Github examples repository. This example code includes an example JavaScript configuration is included should you wish to try the alternate implementation.

However, this guide will focus on the HTML solution, as it is often the preferred method given that the development is simpler and there is no pre-defined structure of the content that may be displayed.

Defining your Audiences and Content

In this example you will use three audiences to determine what content should be shown in a section to the right of the main content area. As you can imagine this technique could be applied to multiple, major parts of your website to completely customize the user experience.

Before your team does any design or copy writing it may help to determine which key Lytics audiences you would like to target. You may want to think about how your website should communicate and look to each of these personas and then build out the content. Even if the scale of your project, the audiences should inform the messaging.

Audiences used for inline content modularization must be API Enabled and have an API ID. You will need to know the API ID for each audience you want to use.

Once you have finalized the audience definitions, designs, and copy for your inline personalized modules, simply start by adding the CSS on your website. Start with the content to be displayed to a single audience. This is the HTML example of content for high momentum users.

HTML

```
<div class="feature-block">
  <div class="block-head">
    Thanks <span>for visiting</span>
  </div>
  <div class="block-inner">
    
    <p>Check out our new products. We provide the highest quality in everything from the newest smart phones to your favorite accessories.
    <a class="btn" href="shop.html">Shop Now</a>
  </div>
</div>
```

First Visit Block

Go ahead and build out all the code for each of your audiences. It may help to comment out the options that will be what an end user in each audience will see instead of viewing all personalized content options at once.

Implementing Content Modularization

When all of your content is good to go it's time to apply the audiences. Consider the following definitions:

A block is a unit of content to be displayed to a single audience within a group.

A group may be made up of one or many content blocks, each with unique audiences. The end user will only be able to p might need to be formatted similarly as they will take up the same space on the page.

In the example code, there is a single group which will control the content displayed to the right of the hero, and four h block to be displayed to users who are not a member of any of the audiences associated with the other blocks in the ibutes:

data-pftrigger - The ID of the audience you wish to display the block to. Each unique value for this attribute defines the data-pfgroup - A string name unique to the group which differentiates it from other personalization groups defined by ty between its content blocks such as placement on the page.

Remember to check out the Pathfora documentation for more information on these attributes and additional code e ization example code looks like:

HTML

```
<!-- High Momentum block-->
<div class="feature-block" data-pfgroup="right_hero_feature" data-pftrigger="high_momentum">
  <div class="block-head">
    Get new coupons <span>every month</span>
  </div>
  <div class="block-inner">
    
    <p>Want to get monthly coupons for up to 30% off your order every month? Sign up below to subscribe to our co
    <a class="btn" href="signup.html">Sign Up</a>
  </div>
</div>

<!--First Time Visitors block-->
<div class="feature-block" data-pfgroup="right_hero_feature" data-pftrigger="first_time_visitor">
  <div class="block-head">
    Thanks <span>for visiting</span>
  </div>
  <div class="block-inner">
    
    <p>Check out our new products. We provide the highest quality in everything from the newest smart phones to yo
    <a class="btn" href="shop.html">Shop Now</a>
  </div>
</div>

<!--Cart Abandoners block-->
<div class="feature-block" data-pfgroup="right_hero_feature" data-pftrigger="cart_abandoners">
  <div class="block-head">
    Get your order with <span>free shipping</span>
  </div>
  <div class="block-inner">
    
    <p>Looks like you were interested in purchasing a product on our site recently. We'd like to offer you free shipping
    <a class="btn" href="checkout.html">Apply Discount</a>
  </div>
</div>

<!--Default block-->
```

```
<div data-pfgroup="right_hero_feature" data-pftrigger="default">
  <!-- Default content here-->
</div>
```

,

Ordering of blocks does matter. For example, you may be wondering what would a user who has a high momentum above? Pathfora will select the first item in the group that is applicable to the current user based on audience membership. If you wanted it to be possible for the user to see both content blocks at once you would need to define them as a group.

High Momentum block

Additional Notes

You may notice some flickering of elements when the page loads. This can happen if Pathfora library and is loaded after the following line of CSS to your website:

CSS

```
[data-pftrigger], [data-pfrecommend]{ display: none; }
```

This ensures that all blocks are hidden upon page load, and once Pathfora has loaded it will set the proper display style.

Updated over 1 year ago

Populate Your Website With One-to-One Content Recommendations

Suggest Edits

Content recommendations are a powerful tool to keep visitors engaged on your website. You can build an even deeper relationship with your users by recommending content directly onto your site.

In this guide, you will learn to implement inline content recommendations using the Pathfora SDK.

Inline Content Recommendation

You can download the complete HTML code for this example from the Github examples repository.

Build a Content Collection

Before you proceed, you will need to create a content collection. See the documentation on content collections to learn how to create content collections for recommendations for instructions on how to get the ID of the collection, which you will need in the code examples, so that the recommendation algorithm can recommend products to users based on their interests.

Constructing The Recommendations

Start by thinking about how you want your content recommendations to look on the page. Check the Pathfora documentation for examples of the presentation of your content. This example will include an image, title, description and a link to the product page.

Inline Recommendation Block

Next determine how many recommendations you want to show on the page. Similar to content modularization, Pathfora can recommend content in different contexts:

A block is a single unit of content (a single article, product, etc).

A group defines one or multiple blocks of content recommendations that pull recommendations from the same Lytics profile. A block can appear more than once in a group.

This example has four blocks in the group. These concepts map to the following data- HTML attributes:

data-pfblock - A unique string name for a single recommendation block.

data-pfrecommend - The ID of the content collection. Blocks with the same data-pfrecommend value are considered part of the same collection. There is an additional attribute data-pftype which is applied to multiple elements within the block. This identifies what type of content is being recommended. See the documentation for in-depth breakdown of what fields are available and how they should be named.

Now consider this HTML example for a single recommendation block:

HTML

```
<li class="product" data-pfblock="product_rec_1" data-pfrecommend="{collection_id}">
```

```

<img data-pftype="image" alt="product recommendation">
<h4 data-pftype="title"></h4>
<p data-pftype="description"></p>
<a class="button" data-pftype="url">Buy<span class="icon"></span></a>
</li>

```

These appear to be empty elements, however once Pathfora makes the call to the Lytics content recommendation API for the img, the href value for the a element, and the inner text of the title and description elements.

That's really all there is to it. You can repeat this code to include multiple blocks for multiple recommendations, and

HTML

```

<!-- Recommendations -->
<div class="recommendation-block">
  <div class="block-head">
    
    Recommended <span>just for you</span>
  </div>

  <ul class="block-inner">
    <!-- Product 1 -->
    <li class="product" data-pfblock="product_rec_1" data-pfrecommend="{collection_id}">
      <img data-pftype="image" alt="product recommendation">
      <h4 data-pftype="title"></h4>
      <p data-pftype="description"></p>
      <a class="button" data-pftype="url">Buy<span class="icon"></span></a>
    </li>

    <!-- Product 2 -->
    <li class="product" data-pfblock="product_rec_2" data-pfrecommend="{collection_id}">
      <img data-pftype="image" alt="product recommendation">
      <h4 data-pftype="title"></h4>
      <p data-pftype="description"></p>
      <a class="button" data-pftype="url">Buy<span class="icon"></span></a>
    </li>

    <!-- Product 3 -->
    <li class="product" data-pfblock="product_rec_3" data-pfrecommend="{collection_id}">
      <img data-pftype="image" alt="product recommendation">
      <h4 data-pftype="title"></h4>
      <p data-pftype="description"></p>
      <a class="button" data-pftype="url">Buy<span class="icon"></span></a>
    </li>

    <!-- Product 4 -->
    <li class="product" data-pfblock="product_rec_4" data-pfrecommend="{collection_id}">
      <img data-pftype="image" alt="product recommendation">
      <h4 data-pftype="title"></h4>
      <p data-pftype="description"></p>
      <a class="button" data-pftype="url">Buy<span class="icon"></span></a>
    </li>
  </ul>
</div>

```

Inline Content Recommendation

Remember that the recommendation API will return different content for different users based on their affinities in I th different pieces of content in your collection to ensure that it will work for content with or without an image, desc our content collection accordingly.

Additional Notes

As mentioned in the content modularization guide, you may notice some flickering of elements when the page loads as the Lytics recommendation API is loaded after the DOM itself has loaded. To get around this, you can add the following CSS

CSS

```
[data-pftrigger], [data-pfrecommend]{ display: none; }
```

This ensures that all recommendation blocks are hidden upon page load, and once Pathfora has loaded it will populate

Updated over 1 year ago

Promote Relevant Content to Users based on their Interests

Suggest Edits

Lytics audiences using Affinities in combination with Facebook custom audiences allows you to target known users with content to promote an upcoming event, product launch, or a new piece of content for your brand. You can also use the same audience to promote relevant content to new users.

Identify Key Topics

The Lytics content affinity engine builds a topic taxonomy from content on your website. Lytics will use this taxonomy to identify key topics. These content affinity scores are available in the audience builder, and will be core to building your target audience.

Select the piece of content you wish to promote. Then take some time to familiarize yourself with the topics in your Lytics content affinity engine relevant to your content. You can sort by the average level of affinity on the topic list view to ensure you are using a good mix of these topics to include in an Affinity, which you can then use to create audiences.

Clicking on a topic displays a summary page which shows a full distribution of user interest.

lytics topic summary

Build an Audience Using Content Affinity

For most marketing use cases, it's useful to target users based on their interests in a group of related topics. For this, you can use Lytics Content Affinities. You can also target users based on their interest in an individual topic. See our Audience Builder documentation for more details.

Once you have your Affinity rules included, verify that the user count of your audience looks correct. If you plan to use this audience anywhere from 1,000 to 50,000 users. For a simple retargeting audience, you'll want to reach a broad number of users.

Using your Lytics Audience in an Ad Campaign

To use your Lytics content affinity audience in Facebook lookalike ads campaign, Follow through with the steps below:

Export to Facebook

Create a Facebook Lookalike Audience

Assign your Lookalike Audience to an Ad

To run a retargeting ad to the users in this audience:

Export to Facebook as a custom audience.

Create your ad following your normal process. If you are building an ad for the first time, follow the Facebook Ad creation steps.

Select the custom audience created by the export during the audience set up process of ad creation.

content affinity ad

Updated over 1 year ago

Reach the Right People Using Lytics

Suggest Edits

Isolated, channel-centric events, like website clicks or email opens, don't provide a comprehensive view of your customer. Relying on these events to build the audience lists for your marketing ad campaigns can lead to your ad dollars being spent inefficiently by either targeting the wrong individuals who are unlikely to convert.

Lytics helps you improve your ROI by advertising to the right people based on the content they are interested in and consuming.

Let Lytics Build Your Audiences

Using the Lytics JavaScript Tag, Lytics will catalog your website content in real-time and group individuals based on their interests. Additionally, using behavioral algorithms, the platform will analyze and score your customers by their overall engagement with your content.

Content affinity audience

Audiences list

As your audiences change over time, use Lytics as a centralized hub for automatically syncing them with your connections to DoubleClick for Publishers, for an up-to-date view of your target customers.

Refine Your Target Audience

The Lytics Query Language (LQL) allows you to refine your target audience and remove customers with certain profile attributes or spend.

Targeted Ad Campaigns

Examples of how Lytics audiences can help with your most common targeted advertising.

Smarter 'lookalike' prospecting with Facebook „Lookalike... audiences based on your most valued customers' behavior.

Target your users with content they are interested in using Lytics topics and content affinity scores for building your content strategy.

Updated over 1 year ago

Retain Existing Customers With Lytics

Suggest Edits

Now more than ever, companies must focus on retaining and growing their existing customers. Acquiring customers is expensive, and retention is a key acquisition tactic. For this reason, lower to mid-funnel tactics are gaining attention from CMOs. Companies can no longer rely solely on top-funnel tactics.

The following is the Lytics Playbook for retaining existing subscribers.

The framework

There are two important concepts within retention marketing as it pertains to Lytics. While these concepts apply across all marketing, they are the core of this playbook. For our purposes, we will define these concepts as follows:

Renewals: customers who have an upcoming renewal date.

Winbacks: recovery efforts for subscribers whose renewal date has lapsed.

How Lytics helps

Lytics helps you renew and winback customers in multiple ways:

Identifying individual users who have upcoming renewals.

Scoring the likelihood of renewal or churn.

Triggering cross-channel messaging to encourage renewal.

Executing targeted promotions and discounts.

Recognizing a user whose renewal date has past or subscription has expired.

Driving personalized winback communications across channels.

The big breakthrough for marketers is that retention teams now have the opportunity to automate consistent messaging across all channels.

found „Companies with extremely strong omnichannel customer engagement see a 9.5% year-over-year increase in revenue per user (RPU).

ly, strong omnichannel companies see a 7.5% year-over-year decrease in cost per contact, compared to a 0.2% year-over-year increase.

The playbook

Step 1: Create a renewals audience

Consider the fields available for renewal in your audiences. Some common ones might include the time of last purchase or the time of last engagement. You can also add additional criteria such as filters excluding complementary subscriptions.

Subscription-Screenshot

Effort Impact Developer Required

Low Low (building block for cross-channel use cases) No

Consider that you will likely have several renewals audiences - one per subscription.

Step 2: Export audience to all key integrations

The focus of renewals and winbacks will be cross-channel which can include:

Google, Facebook, Verizon Media (Yahoo Gemini), your email service provider (ESP) available as Integrations, and the Lytics mobile app.

Step 3: Create on-site modals targeting renewals

The most common renewal tactic is to deliver on-site modals to users with an upcoming renewal. Take the renewal audience and create a modal that triggers when a user is about to renew.

periences to target them. See the slide-out modal on the left side below as an example.

On-side Modals

Further instructions are available for business users and for developers.

Effort	Impact	Developer Required
Low	High	No, but helpful for detailed customization

Step 4: Add targeted email to your renewal journeys

The email inbox is one of the most impactful places to reach your renewal users. First, build an email in your ESP that follows the instructions in your ESP to do so, and this varies considerably by tool.

Effort	Impact	Developer Required
Medium	High	No, but helpful for detailed customization of the email template

Once you have a triggered email built in your ESP, make use of the triggered email feature your ESP likely has.

Whatever ESP you have, you probably do not want to include „Existing Users... in the audience, so we recommend leading users who enter your renewal audience will be sent the automated message. For examples of how you can go a step further, see the Iterative use case.

Step 5: Involve telesales (optional)

Imagine incorporating your message across channels to even include telesales. Your telesales team likely has a strong sales record. There are two ways to incorporate telesales:

Encourage visitors to call inbound.	Deliver a list of leads for targeted outbound.
Effort	Impact
Low	Medium
Medium	No

Encourage Visitors To Call Inbound

Leveraging very simple HTML, you can create click to call campaigns in Orchestrate. Then, users on a mobile phone will be able to click to call.

click-to-call-modal-small

Deliver a List of Leads for Targeted Outbound

Build a list of all the upcoming renewals you care about. Here's an example using several audiences of subscriptions to build a telesales list:

subscription-building-block

Step 6: Incorporate direct mail (optional)

Using the same example as above, consider adding address information to the export and sending it to a fulfillment partner.

direct-mail

As a final note, keep in mind there are many other areas to cover including copy choices, step-by-step optimization, and a playbook, but should be taken into consideration while developing your marketing strategy to retain customers.

Updated over 1 year ago

Personalize your Iterable emails With Lytics content recommendations

Suggest Edits

Lytics content recommendations provide powerful personalized, one-to-one recommendations. Sending content recommendations can help you increase engagement with your brand.

This guide will cover how to implement content recommendations in email with Iterable as the email service provider. We'll use the Lytics data feed to render them inline in your email template. This technique can be combined with other email use cases.

y rich, personalized experience.

Although you will be using the Lytics API, no code is necessary to implement this solution beyond configuring some H

Before you begin

You should have an Iterable account with an audience export running and an HTML email template with a section pr
ndations can include any/all of the following:

Title

Description

Image

URL or link to the document

Author

Date of publication

You can start by setting up your template with some static content. For context, here's how the content recommend
ncludes all of the fields listed above:

content recommendations in email

In addition, make sure you have a Lytics account with the content affinity engine activated. You will also need to gener

Build a content collection

To make recommendations, you will need a content collection. You can see our documentation on building content c
d of the content collection as you will use it in the next step.

Add content recommendation API as a data feed

To populate recommendations in the email, you will use the Iterable feature called a Data Feed to hook into the Lytic

Login to your Iterable account.

Navigate to Integrations > Data Feeds in the side bar.

Click + Create New Feed.

Enter "Lytics Content Recommendations" or some other descriptive title in the Name text input.

Make sure the JSON option is selected from the Format dropdown.

Copy and paste the following URL into the URL text input:

<https://api.lytics.io/api/content/recommend/user/email/{{#urlEncode}}{{email}}{{/urlEncode}}?contentsegment=>

At the end of that URL, paste the ID of the content collection you created in the previous step. This ID is the hash in t

In the Authorization Token field paste a Lytics API token with read permission for content and user search.

Verify that your settings look correct and then click Save Feed.

complete data feed

Enabling this data feed in a template allows Iterable to call the Lytics content recommendation API before each email

}} with the URL encoded version of the recipient's email address. The next step will cover how to access the content t

You can add additional query parameters to the URL in the data feed to control things like the ranking of the recom
commendation API documentation for a list of possible query parameters for this end point.

Modify your email template

Now that the data feed has been configured, it's time to apply it to your email template.

In Iterable open the template in edit mode.

Click the Advanced options tab and configure the following settings.

Check the Enable template generation using a data feed... checkbox.

From the dropdown, select the data feed you created in the previous step.

Check the Cache data feed response (cached for up to 1 hour).

(Optional) Check the Merge the data feed and user contexts checkbox.

advanced options

If you do not select Merge the data feed and user contexts, you will need to use square brackets `[[value]]` instead of `{{value}}`. The examples in this guide assume that this option is checked.

Under the HTML version of the template, click Source to view the HTML source of your template.

Locate the HTML of the recommendation section. You will need to implement templates for each of the fields you will use. For an example of the response example in our API documentation, but in general each field should match the following format:

`{{data.[i].fieldname}}`

Where *i* is the current index of the recommendation (starting at 0), and *fieldname* is one of the following:

title: Title of the document

description: Description of the document

primary_image: URL of the document's image

url: URL of the document (excluding the protocol).

author: Name of the author of the document (this field may not be available for all content types).

created: Date that the article was published.

Here is some example HTML laying out all of these fields for 3 separate recommendations:

HTML

```
<!-- Recommendation 1 -->
<div>

<h5>{{data.[0].title}}</h5>
<span>By {{data.[0].author}} | {{dateFormat data.[0].created format="full"}}</span>
<p>{{data.[0].description}}</p>
<a href="https://{{data.[0].url}}">Read More</a>
</div>
```

```
<!-- Recommendation 2 -->
<div>

<h5>{{data.[1].title}}</h5>
<span>By {{data.[1].author}} | {{dateFormat data.[1].created format="full"}}</span>
<p>{{data.[1].description}}</p>
<a href="https://{{data.[1].url}}">Read More</a>
</div>
```

```
<!-- Recommendation 3 -->
<div>

<h5>{{data.[2].title}}</h5>
<span>By {{data.[2].author}} | {{dateFormat data.[2].created format="full"}}</span>
<p>{{data.[2].description}}</p>
<a href="https://{{data.[2].url}}">Read More</a>
</div>
```

As seen above, you must add the http or https protocol in the href protocol before the URL field. Also the example above is just a simplified version of what you can do. For more information, see an read more about this in Iterable's handlebars documentation.

Once you have finished editing the template, click Save Template. In the next step we'll verify that the data feed is working.

Previewing and verifying the recommendations

Iterable allows you to preview a rendered version of your email in the browser with live data from your data feed. For more information, see the content recommendations.

rendered preview

Once you're happy with the preview, you are ready to send your campaign. You can implement your template into a

Rate Limiting

For best results, please talk to your Iterable representative about setting up a Message Send Rate Limit. The Lytics' C
cond, or 15,000 requests per minute. If the rate-limit is not enabled with Iterable, Lytics may return a 429 HTTP Statu

Troubleshooting

For more information on troubleshooting your templates, see Iterable's documentation on templating. If you're not a
PI, please make sure that the email you are testing with exists as a valid profile in Lytics. You can verify this, by naviga
, and searching for the user you are trying to preview as. The API documentation for content recommendations may
issues persist, feel free to contact support@lytics.com for further assistance in troubleshooting the Lytics API.

Updated over 1 year ago

Build a Custom Personalized Experience With Video Content

Suggest Edits

Video content is a great way to capture your users' attention. Serving your videos to a Lytics audience of users at wh
ngagement.

In this guide, you will learn how to create a widget with a video component. While videos are not natively supported
lback function to load a video with JavaScript. This callback can be applied to an Experience built through the UI with

Video Modal

Follow along with the video guide below or this written guide. You can download the complete JavaScript configurati
examples repository.

Javascript Configuration

Begin with a generic widget configuration.

JavaScript

```
var videoWidget = window.pathfora.Message({  
  id: "video-widget",  
  layout: "modal",  
  className: "video-widget",  
  headline: "iPhone XR available now",  
  msg: "Get yours, or a discount on all previous models today from <strong>MEGASTORE</strong>.",  
  okShow: true,  
  okMessage: "Shop iPhones",  
  cancelShow: false,  
});
```

```
window.pathfora.initializeWidgets([videoWidget]);
```

This config will generate a simple modal, with a headline, message and call to action.

Simple Modal

To add a video to the modal, you can define an onLoad callback function. This function is called just before the modal
function as arguments include the widget's DOM element, which you will append the video element to.

JavaScript

```
var videoWidget = window.pathfora.Message({  
  // config settings ...  
  onLoad: function (event, module) {  
    // add the video element  
  }  
});
```

Start by creating a div element which will later contain your video element. You may want to add a class name such a

of the div with the code for your video. Note that you'll need to escape quotes within the HTML when transforming it.

JavaScript

```
onLoad: function (event, module) {  
  // create a div for the video  
  var videoDiv = document.createElement("div");  
  videoDiv.className = "pf-widget-video";  
  
  // place the video in the div  
  videoDiv.innerHTML = "<iframe src=\"https://www.youtube.com/embed/tG7vx7-3sl0\" frameborder=\"0\" allowfullscreen\">";  
}
```

Now that you have the video element within a div, simply insert the div into the widget. You can access the DOM node for the widget through `module.widget`. From there you will want to select two elements with the class names:

`pf-widget-content` the main div that contains content of the widget.

`pf-widget-text` a child div of `pf-widget-content`, which contains the text content of a widget.

Using the `insertBefore` function, you can insert the div containing your video into the main content div, before the text.

JavaScript

```
onLoad: function (event, module) {  
  // create a div for the video  
  var videoDiv = document.createElement("div");  
  videoDiv.className = "pf-widget-video";  
  
  // place the video in the div  
  videoDiv.innerHTML = "<iframe src=\"https://www.youtube.com/embed/tG7vx7-3sl0\" frameborder=\"0\" allowfullscreen\">";  
  
  // append the div to the widget node  
  var content = module.widget.querySelector(".pf-widget-content");  
  var text = module.widget.querySelector(".pf-widget-text");  
  content.insertBefore(videoDiv, text);  
}
```

With this new config you should now have a modal with a video! But it could probably use CSS to make it look cohesive.

Video Unstyled

Styling the Video Modal

With some additional CSS work, you can style the video to fit your modal. Download the code for this example to see how we styled our modal.

Styled Video Modal

Once you've tested and are happy with the look and feel of your video modal, it's time for the most important step: configuring the widget.

Define the Lytics Audience

The example widget in this guide will be targeted at audience of users with a high content affinity score for iPhones, so we'll configure the widget with the audience slug you wish to serve the widget to.

Alternatively, you may build your widget as a web personalize Experience in the Lytics UI and apply the `onLoad` callback to it.

Updated over 1 year ago

Unlock Additional Web Personalization Features with Lytics API Overrides

Suggest Edits

Lytics web personalize Experiences built through the Lytics UI are powered behind-the-scenes by the Pathfora JavaScript SDK. However, you can also build Experiences using the Lytics UI however such as custom forms, or JavaScript callbacks. Luckily you can apply these features to Experiences built through the Lytics UI.

In this guide, you will learn how to make an API requests to override the configuration of an Experience built in the Lytics UI.

editor and API overrides. The overrides will include the custom form configuration, form error and success states, and more. For more information, see [API Overrides](#).

Output Slideout

You can download the complete API request command for this example from the [Github examples repository](#).

The primary benefit of using the Lytics UI in combination with API overrides instead of the Pathfora SDK is that you can have access to the native Experience, Stage, and Journey reporting features in the Lytics UI without having to build custom integrations.

Building the Experience in Lytics

If you haven't already, you will need to build the base Experience through the Lytics Experience Editor. The goal is to define the desired end state of the widget, and rely on the API overrides to fill in the gaps that are not available for configuration in the UI.

Selecting a Tactic

Selecting the correct tactic is key, as it determines what type of Pathfora widget the Experience outputs. The tactics are:

Drive Traffic - produces a message widget with CTA (okShow set to true).

Capture Leads - produces a form widget.

Present a Message - produces a message widget with no buttons (okShow set to false).

Recommend Content - produces a message widget with content recommendations and variant set to 3.

The example in this guide uses a form widget, thus you will select the Capture Leads tactic.

Experience Editor Steps

You will be dropped into the customize form step of the editor. Since the example form has custom checkboxes set to be visible in the UI. Simply click the Next Step button.

In the design step enter the headline, body, and call to action text. Select the Slideout layout, and under theme you can name custom-tracking-widget.

In the target step, select the audience you wish to show this slideout to. This example uses an audience of high intent visitors.

In the display step configure when and where you want the form slideout to display on your website. This step controls the timing and location of the slideout.

Once you've completed all the steps, you may want to preview your Experience. Though it may not look or function in the Lytics UI, it will be a good foundation configuration which you will augment in the next step.

Legacy Form

You will also need to save the Experience to make API override requests. Be sure to click Save as Draft.

Creating the Override Request

To make the request for the override, you will need the ID of the Experience you just created. You can get this from the Lytics UI by clicking on the Experience and viewing it.

In the URL example below, the ID of the Experience is 2d3e345b2ac24acd9d6d3d33f93516fd:

ID in URL

You will also need a Lytics API token with the Admin role. It may help to save the token as an environment variable for your terminal.

Shell

```
export LIOKEY={api_token_here}
```

You will make a PATCH request to the Experience endpoint with the id of the Experience. This command example uses the ID from the URL above.

Shell

```
curl -s -H "Authorization: $LIOKEY" \
-H "Content-Type: text/json" \
```

```
-XPATCH "https://api.lytics.io/api/experience/{experience_id}" \
-d '{
  JSON PAYLOAD WILL GO HERE
}' | jq '
```

Your config changes will need to be nested in a field called detail_override this field itself is nested in the experience.

JSON

```
{
  "experience": {
    "vehicle": {
      "detail_override": {
        // configuration overrides
      }
    }
  }
}
```

Next you will need to convert the JavaScript configuration into a JSON payload containing the settings which you were

formElements - which customizes the form.

formStates - which sets the success and error states on form submission.

confirmAction - which handles the callback to send the data to a third party.

For the most part, you can translate a config to a JSON override in the same way that the JSON.stringify() function works perfectly for formElements and formStates:

JSON

```
{
  "experience": {
    "vehicle": {
      "detail_override": {
        "formElements": [
          {
            "type": "text",
            "required": true,
            "label": "Email Address",
            "name": "email"
          },
          {
            "type": "checkbox-group",
            "required": true,
            "label": "Which feeds would you like to subscribe to?",
            "name": "subscription_feeds",
            "values": [
              {
                "label": "Beauty & Perfumes",
                "value": "beauty"
              },
              {
                "label": "Electronics",
                "value": "electronics"
              },
              {
                "label": "Fashion",
                "value": "fashion"
              }
            ]
          }
        ]
      }
    }
  },
  "formStates": {
    "success": {
```



```

    "headline": "Success",
    "msg": "Thanks for signing up, you can expect to receive updates in your inbox soon."
  },
  "error": {
    "headline": "Error",
    "msg": "There was an issue submitting your subscription. Please try again or <a href=\"/contact\">contact us</a>."
  }
}
}
}
}
}
}
}

```

But for `confirmAction`, it gets tricky because it contains a JavaScript function:

JSON

```

{
  "experience": {
    "vehicle": {
      "detail_override": {
        // formElements
        // formState

        "confirmAction": {
          "waitForAsyncResponse": true,
          "callback": // callback function here somehow?
        }
      }
    }
  }
}

```

However, the API can accept a JavaScript function as a string. It may help to minify your function first. There are a number of online tools that can help you minify your JavaScript for you. This helps for the API override because it creates a shorter, single line version of the function. This is the minified version of the function:

JavaScript

```

function(e,a,t){if(a.data){var n=a.data.reduce(function(e,a){if(e.hasOwnProperty(a.name)){var t=e[a.name],n=Array.isArray(t)?t:[t];return e},{}),r=new XMLHttpRequest;r.onload=function(){200===this.status?t(!0):t(!1)},r.open("POST","http://yourweb.com/api/subscribe?email="+a.email+"&password="+a.password+"&confirmAction="+JSON.stringify(n)),r.setRequestHeader("Content-Type","application/json;charset=UTF-8"),r.send(JSON.stringify(n))}}

```

To include this function as a string, you will need to escape any double quotes within this before pasting it into your JSON:

JSON

```

{
  "experience": {
    "vehicle": {
      "detail_override": {
        // formElements
        // formState

        "confirmAction": {
          "waitForAsyncResponse": true,
          "callback": "function(e,a,t){if(a.data){var n=a.data.reduce(function(e,a){if(e.hasOwnProperty(a.name)){var t=e[a.name],n=Array.isArray(t)?t:[t];return e},{}),r=new XMLHttpRequest;r.onload=function(){200===this.status?t(!0):t(!1)},r.open(\"Content-Type\\\",\\\"application/json;charset=UTF-8\\\")r.send(JSON.stringify(n))}}"
        }
      }
    }
  }
}

```

Putting the whole thing together, your curl command will look like this:

Shell

```
curl -s -H "Authorization: $LIOKEY" \
-H "Content-Type: text/json" \
-XPATCH "https://api.lytics.io/api/experience/{experience_id}" -d '{
  "experience": {
    "vehicle": {
      "detail_override": {
        "formElements": [
          {
            "type": "text",
            "required": true,
            "label": "Email Address",
            "name": "email"
          },
          {
            "type": "checkbox-group",
            "required": true,
            "label": "Which feeds would you like to subscribe to?",
            "name": "subscription_feeds",
            "values": [
              {
                "label": "Beauty & Perfumes",
                "value": "beauty"
              },
              {
                "label": "Electronics",
                "value": "electronics"
              },
              {
                "label": "Fashion",
                "value": "fashion"
              }
            ]
          }
        ]
      },
      "formStates": {
        "success": {
          "headline": "Success",
          "msg": "Thanks for signing up, you can expect to receive updates in your inbox soon."
        },
        "error": {
          "headline": "Error",
          "msg": "There was an issue submitting your subscription. Please try again or <a href=\"/contact\">contact us</a>."
        }
      },
      "confirmAction": {
        "waitForAsyncResponse": true,
        "callback": "function(e,a,t){if(a.data){var n=a.data.reduce(function(e,a){if(e.hasOwnProperty(a.name)){var t=e[a.name];if(t===a.value)return t;}return e},{}),r=new XMLHttpRequest;r.onload=function(){200===this.status?t(!0):t(!1)},r.open('PATCH','https://api.lytics.io/api/experience/{experience_id}',true),r.setRequestHeader(\"Content-Type\",\"application/json;charset=UTF-8\"),r.send(JSON.stringify(n))}"
      }
    }
  }
}' | jq ''
```

Once you run the command, check that the response from the API includes your changes. You may make subsequent PATCH requests to update your experience.

Testing and Validating the Override

Once you've made the API request, you should be able to view the changes as part of the regular preview process for your Experience. Click Preview and enter the URL you wish to preview the Experience on.

Note: If you are using the same browser session to preview the Experience that you did to create it, you may need to ensure that Lytics is serving the most up to date version of the configuration.

You should be able to see the updated Experience in the preview. If this is not the case there are two scenarios to try.

If your widget is not displaying at all, check the JavaScript console for errors. If you find a formatting error in the transaction override command accordingly. This scenario can happen if you failed to escape quotes properly, for example.

If your widget is not displaying the changes, double check the response from the API and cross-reference your settings to ensure everything is named and formatted correctly.

If the issue persists, you can always contact Lytics Support for additional assistance with debugging.

Updated over 1 year ago

How to Use GTM Tags to Modify Lytics Pathfora Widget Behavior
a Tutorial to modify pathfora behaviour using custom javascript tags in GTM

Suggest Edits

Introduction

Google Tag Manager (GTM) is a powerful tool for managing tags, triggers, and scripts on your website. Lytics offers a variety of ways to personalize and target content to their users. This article will show you how to use GTM to modify the behavior of Lytics widgets using JavaScript and CSS.

Prerequisites

A working knowledge of Google Tag Manager

Basic understanding of JavaScript and CSS

Access to your website's GTM and Lytics accounts

Step 1: Create a JavaScript Tag in GTM for Lytics Embedder

Go to your Google Tag Manager Dashboard and click on "New Tag."

Name the tag "Lytics Embedder."

Choose the tag type as "Custom HTML."

Paste your Lytics Embedder JavaScript code into the HTML field.

For example:

JavaScript

```
// Your Lytics Embedder code here
```

Choose the triggering event for your tag. For instance, you can set it to trigger when a custom event "embedder variable" is fired.

Save and publish the changes.

Step 2: Create a CSS Tag in GTM for Lytics CSS Master

Go back to the GTM Dashboard and create another new tag.

Name the tag "Lytics CSS Master."

Choose the tag type as "Custom HTML."

Insert the CSS code wrapped in style tags into the HTML field.

For example:

HTML

```
<style>
```

```
// Your Lytics CSS Master code here
```

```
</style>
```

Set the trigger to "All Pages Page View" so the CSS will be applied universally.

Save and publish the changes.

Step 3: Testing and Validation

Open your website and inspect the Lytics Pathfora widgets to make sure they are behaving and appearing as intended.

Use browser developer tools to troubleshoot any issues, such as incorrect application of styles or non-functioning JavaScript.

Step 4: Customization and Advanced Uses

You can further customize the Lytics Embedder JavaScript to manipulate widgets based on user behavior or other events. Similarly, modify the Lytics CSS Default to change the look and feel of the widgets to better align with your brand.

Conclusion

Utilizing Google Tag Manager along with Lytics' Pathfora framework can greatly enhance your ability to deliver personalized experiences. Following these steps, you can modify widget behavior and appearance directly from GTM, making your marketing efforts more effective.

Example code

A JavaScript to dynamically alter pathfora behaviour:

Use this code in GTM and associate to a custom variable

```
<script>
// Get the "h2" and "p" elements with their respective class names
// var headlineElement = document.querySelector('.pf-widget-headline');
var messageElement = document.querySelector('.pf-widget-message');

// Get the "img" tag with class name "pf-widget-img"
var imgElement = document.querySelector('.pf-widget-img');

// Create a new "a" tag
var linkElement = document.createElement('a');
linkElement.href = messageElement.textContent;
linkElement.classList.add('pf-widget-custom-href');

// Append the "img" tag inside the new "a" tag
linkElement.appendChild(imgElement.cloneNode(true));

// Replace the "img" tag with the new "a" tag in the DOM
imgElement.parentNode.replaceChild(linkElement, imgElement);
</script>
```

Lytics CSS Default - a JS to update CSS dynamically in Pathfora:

```
<style type="text/css">
/* Lytics-small-modal start */
.lytics-small-modal .pf-widget-content{
  display:none;
}
.lytics-small-modal .pf-widget-img {
  width: 100% !important;
  height: 100% !important;
  float: none !important;
  position: absolute !important;
  top: 0 !important;
  left: 0 !important;
  margin-left: 0 !important;
  border-radius: 0 !important;
}
.lytics-small-modal {
  cursor: pointer;
}
.lytics-small-modal .pf-widget-body {
  width: 100%;
  height: 100%;
}
.lytics-small-modal .pf-widget-close {
  z-index: 1;
}
.lytics-small-modal{
```

```

width: 250px;
height: 250px;
}
/* lytics-small-modal end */
/* lytics-capture-leads styles start */
.lytics-capture-leads .pf-widget, .lytics-capture-leads .pf-widget .pf-widget-body, .lytics-capture-leads .pf-widget-mes
-leads .pf-widget-checkbox{
color: rgb(46, 46, 46) !important;
}
.lytics-capture-leads{
background-color: rgb(235, 177, 17) !important
}
.lytics-capture-leads .pf-widget-headline{
color: white;
}
/* lytics-capture-leads styles end */

/* lytics-preheader-modal styles start */
.lytics-preheader-modal{
background-color: #ffb32a;
min-height: 0;
padding: 0;
box-shadow: none;
}

.lytics-preheader-modal .pf-widget-message{
color: black;
}

.lytics-preheader-modal .pf-widget-close{
color: black;
}

@media (max-width: 600px) {
.lytics-preheader-modal {
display: none;
}
}
/* lytics-preheader-modal styles start */

/* lytics-personalization-coupon styles start */
.lytics-personalization-coupon.pf-widget-modal.pf-widget-variant-2 .pf-widget-img {
float: none !important;
position: absolute !important!important;
top: 20px !important;
left: 5% !important;
margin: 0% !important;
width: 90% !important;
height: auto !important;
border-radius: 0px !important;
}
.lytics-personalization-coupon.pf-widget-modal.pf-widget-variant-2 .pf-widget-content {
background-color: #EEB111 !important;
}
.lytics-personalization-coupon.pf-widget-modal .pf-widget-message {
font-size: 14px !important;
margin: 0 0 0px !important;
}
.lytics-personalization-coupon.pf-widget, .pf-widget .pf-widget-body, .lytics-personalization-coupon.pf-widget-moda
color: black !important;

```

```

}
/* lytics-personalization-coupon styles end */
/* lytics-personalization-coupon black-back styles start */

.lytics-personalization-coupon.black-back.pf-widget-modal.pf-widget-variant-2 .pf-widget-content {
  background-color: black !important;
}
.lytics-personalization-coupon.black-back.pf-widget, .pf-widget .pf-widget-body, .lytics-personalization-coupon.black-back.pf-widget {
  color: white !important;
}
/* lytics-personalization-coupon black-back styles end */

/* lytics-big-modal styles start */
.lytics-big-modal .pf-widget-img{
  position: relative !important;
  margin: 0 !important;
  top: 0 !important;
  left: 0 !important;
  border-radius: 0 !important;
  width: 100% !important;
  height: 100% !important;
  cursor: pointer !important;
}

.lytics-big-modal .pf-widget-text{
  display: none !important;
}

.lytics-big-modal .pf-widget-content{
  padding: 0px !important;
}
.lytics-big-modal .pf-widget-close{
  z-index: 1 !important;
  color: black !important;
}
/* lytics-big-modal styles end */
</style>

```

Updated over 1 year ago

Style your Lytics Web Experiences to Fit Your Brand Guidelines

Suggest Edits

Lytics web Experiences are meant to provide a very simple way for marketers to spin up personalized content on the web. This is a great way to create a consistent look and feel for your website, but if your website has specific style guides, you may prefer to customize the look and feel of your Experience. But if your website has specific style guides, you may prefer to

In this guide, you will learn how to construct a global stylesheet to establish a base set of styles for any Lytics Web Experience. This guide will cover how to write more specific styles for a single unique widget, and how to troubleshoot common issues with

Bar and Slideout Styles

You can download the completed CSS and a JavaScript configuration for testing this example from the [GitHub example](#).

Defining a Global Stylesheet

If you are planning to implement many web Experiences on your site, you will want to establish a stylesheet that will help to generate an Experience or Pathfora config of each layout, and think about what you want to augment for each widget type, slideout, bar, gate, and modal widget to test against.

Slideout Bar Unstyled

Gate Unstyled

Modal Unstyled

The stylesheet in this guide will be kept light, by simply adding a background image to all widgets and tweaking some colors. However, as you can imagine, you can go even further to change any aspect of the widget including the size or positioning. Here is a list of all of the class names of elements used in Pathfora widgets.

Start by adding some general styles you would like to apply to all widgets. `pf-widget` is the class name applied to the widget container. `pf-widget-content` is the first child element. For slideouts, the `pf-widget-content` is essentially wraps the same area as `pf-widget`. For modals, the entire page with a background overlay while `pf-widget-content` contains the modal content itself.

CSS

```
.pf-widget .pf-widget-content {  
  background: url("assets/subtle-background.jpg") no-repeat;  
  background-size: cover;  
  font-family: "Helvetica Neue", "Arial" sans-serif;  
  color: #666;  
  border-radius: 0px;  
  box-shadow: 0 0px 7px 0px rgba(0,0,0,0.3);  
}
```

Bar Unstyled

There's a problem though, widgets with the bar layout don't have a `pf-widget-content` div. So you can adjust the selector to include the bar.

CSS

```
.pf-widget-bar.pf-widget, .pf-widget .pf-widget-content  
Background Image
```

Next apply some additional styles to the call to action buttons. `pf-widget-btn` is the class name used on all Pathfora buttons.

CSS

```
.pf-widget .pf-widget-btn.pf-widget-ok {  
  box-shadow: 0px 1px 1px rgba(0, 0, 0, 0);  
  font-weight: normal;  
  background-color: #5a88ca;  
  border: 0;  
  color: white;  
  font-size: 16px;  
}  
  
.pf-widget .pf-widget-btn.pf-widget-ok:hover {  
  background-color: #666;  
}
```

Lastly add some styles to match the color of the headline with the widget text, and minimize the size of the the widget.

CSS

```
.pf-widget .pf-widget-headline {  
  color: #666;  
}  
  
.pf-widget .pf-widget-footer {  
  height: 0;  
  margin: 0;  
}
```

Bar and Slideout Styles

These styles will cover future widgets you wish to build on the website. As you are building your own stylesheet, if you run into a problem, see the [troubleshooting section](#).

Widget Specific Styles

Using the Experience Editor or the Pathfora SDK, you can define a custom class name for your widget. With this class name, you can write CSS to style the widget. Always preface your selector with the pf-widget class name to ensure that you don't accidentally effect other elements on the page.

CSS

```
.pf-widget.your-custom-class
```

For a full in-depth example of specific widget styles, see the [image customization guide](#).

Troubleshooting and Selector Specificity

A common problem when writing CSS for Pathfora widgets is providing the correct level of specificity when selecting elements. If you write a selector that is too specific, you might not expect to see the change applied but didn't - this section is for you. The default CSS that the Pathfora library ships with is designed to be specific enough to not effect any other part of your website.

The main side effect of this is that developers wishing to override the default styles must pay special attention to the specificity of the selectors they use to override the default styles of the library.

Consider the following example using !important:

CSS

```
.custom-class .pf-content-unit h4 {  
  font-weight: normal !important;  
  font-style: italic;  
}
```

The selector here is short and simple. However, because the bold font-weight is embedded in a deeper selector, it may not be applied. This is a problem because the selector is too specific.

For reference, these is the relevant snippet of styles in the Pathfora CSS:

CSS

```
.pf-widget-variant-3 .pf-content-unit .pf-content-unit-meta h4 {  
  font-weight: 500;  
  margin: 0 0 5px;  
  line-height: 1.4;  
  color: #444;  
}
```

Now consider how you would need to write this CSS without using !important:

CSS

```
.pf-widget.custom-class.pf-widget-variant-3 .pf-content-unit .pf-content-unit-meta h4 {  
  font-weight: normal;  
  font-style: italic;  
}
```

This is a ridiculously long selector! However the advantage is you no longer need to know which styles need to be overridden.

Both techniques are valid and will result in the same changes to your widget. Feel free to use whatever method is easiest for you. If you are using a CSS pre-processor that allows for nested selectors, what would longer selectors may be easier to interpret.

If you're not a fan of !important, continue reading for an in-depth look at how to easily derive the required level of specificity.

Utilizing Chrome Developer Tools

The best way to determine how to override a default style is to look the conflicting style from the CSS of the library. You can find the Pathfora CSS in the [Pathfora GitHub repository](#) which is open source, but it may be easier to debug your CSS using the Inspect Element feature in Google Chrome.

Inspect Element

Locate the relevant element in your widget by clicking on it while in the element selection mode in developer tools. You'll find the element in the Elements tab. Examine the Styles tab for that element to see where the conflicting style is. Once you've located it, you can remove the style or modify it to your liking.

CSS

.pf-widget-variant-3 .pf-content-unit .pf-content-unit-meta h4

Then simply add the custom class of your widget accordingly to the selector:

CSS

.pf-widget.custom-class.pf-widget-variant-3 .pf-content-unit .pf-content-unit-meta h4

Add your rules, save your stylesheet, and test that the changes in the browser. Remember, you can always consult the documentation for more information on how to use the Personalize Experiences.

Updated about 1 year ago