UNCLASSIFIED

Ontario

GOOGLE ANALYTICS 4

DIGITAL INNOVATION AND PRODUCT DIVISION

MPBSD

PRIVACY IMPACT ASSESSMENT

VERSION 1.01-APPROVED

JUNE 18, 2024

Under the direction of the

MPBSD Privacy Office

MPBSDPO Project Number: 2023-46

Record of Changes

REVISION	DESCRIPTION	ISSUE DATE
0.1	Initial review with project/program area	August 11, 2023
0.2	Review with Privacy and project/program area	December 14, 2023
0.3	Third Review by Privacy	January 8, 2024
0.4	Fourth Review by Privacy	January 23, 2024
0.5	Fifth Review by Privacy	February 7, 2024
0.6	Sixth Review by Privacy	February 12, 2024
0.7	Seventh Review by Privacy	February 21, 2024
1.0	Approved by MPBSD Head of Privacy	April 30, 2024
1.01	Minor update to clarify Custom Events	June 10, 2024
1.02	Program area requested another change	June 18, 2024

Acknowledgements:

PIA was written by Justin DUCKETT, (MPBSD Digital Innovation and Product Division) and Arshak MAVISAKALYAN (MPBSD Privacy Office).

PIA was reviewed by Peter HOPE-TINDALL and Matthew R. SMITH (MPBSD Privacy Office).

Notes:

This project is saved at: < P:\PRIVACY\PROJECTS - INF–CMP–002 (R)\2023 MPBSDPO Projects\PR2023-46 - MPBSD - Google Analytics 4 >

Ontario Public Service Mission:

Taking pride in strengthening Ontario, its places, and its people.

MPBSD Privacy Office Mission:

We are the MPBSD privacy experts and champions. We are proud and passionate stewards and protectors of our clients' personal information. Our goal is to ensure our clients receive excellent customer service and excellent privacy protection! We give our colleagues honest and helpful privacy advice and direction.

Table of Contents

[This page intentionally blank]

1. Summary

In October 2020, Google announced an update to their Google Analytics product - Google Analytics 4 (GA4). The current version of Google Analytics (known as Universal Analytics) will sunset and be replaced by Google Analytics 4 on July 1, 2023 for free customers and July 1, 2024 for paid enterprise customers.

Various OPS projects/program areas use Google Analytics to collect data about how users (an entity, usually a person that starts a session on a website or an app, by visiting a website and/or downloading and using interacting with an app) and engage with websites and apps. This data helps to identify potential performance issues and improvements on websites/apps and to obtain aggregate insights into website/app use.

This PIA will cover a high level overview of how GA4 works, an analysis of key considerations when using GA4, provide recommendations to best mitigate potential privacy risks, and provide a suggested guideline for default implementation of GA4 that can be referred to and used by all OPS projects/program areas that will be utilizing this analytics tool. Establishing OPS-wide default guidelines for the use of GA4 have many significant benefits which include:

- Eliminating the duplication of effort of writing a new PIA for every website and application that is built. Especially when only a basic implementation of Google Analytics 4 is required.
- Improving privacy outcomes, by applying the same web analytics standards to every website using Google Analytics 4.
- Being able to support OPS projects/program areas which need to add the ontario.ca Google Analytics 4 tracking snippet to their application, but don’t have the required skill set to do so.
- The ability to enable cross domain tracking across different government domains when needed (through using the same GA4 tracking snippet across multiple domains)
- A standard approach would ensure that the right tracking codes, configurations, and measurement goals are consistently applied across all OPS websites and apps, resulting in accurate and reliable insights into user interactions, website performance, and marketing effectiveness.
- Many additional benefits described in Section 3.2.

This PIA sets out a baseline level of GA4 use/functionality that is recommended by default. GA4 offers many additional features and functionalities going beyond the scope of the recommended parameters addressed in this PIA. Therefore, use of those would need to be discussed and consulted with the respective privacy teams/departments of any given project/program area of any ministry within the OPS.

2. Introduction

2.1.1 Definitions and descriptions

Term	***Description***
API	Application programming interface — a software intermediary that allows two applications to talk to each other.
BigQuery	Google's fully managed, serverless data warehouse that enables scalable analysis over petabytes of data.
Consent Management Platform (CMP)	A technology that websites use to obtain the legal consents from users to process their personal data, typically through cookies and trackers in operation on the domain.
Data Thresholds	Data thresholds are applied to prevent anyone viewing a report or exploration from inferring the identity or sensitive information of individual users based on demographics, interests, or other signals present in the data.
Firebase	A set of backend cloud computing services and application development platforms provided by Google.
Google Analytics 4 360	Paid, subscription-based enterprise version of Google Analytics 4.
Google Analytics 4 account	An access point for GA4, which contains all properties and reports.
Google Ads	Google's online advertising platform that allows you to create online ads to reach audiences who are interested in the products and services one may offer.

Google Analytics 4 property	A property represents a grouping of data from a website and/or app in Google Analytics 4.
Google Cloud	A suite of cloud computing services that provides a series of modular cloud services including computing, data storage, data analytics, and machine learning, alongside a set of management tools.
Google Tag	The Google tag (gtag.js) is a single tag one can add to a website to use a variety of Google products and services (e.g., Google Ads, Google Analytics).
Google Tag Manager (GTM)	GTM allows to quickly and easily update measurement codes and related code fragments collectively known as tags on a website or a mobile app.
Measurement ID	A measurement ID in Google Analytics is a unique identifier for a web data stream (which is a website registered within Google Analytics).
SDK	Software development kit is a set of tools to build software for a particular platform. These tools also allow an app developer to build an app which can integrate with another program.

2.2 How Google Analytics 4 Works

2.2.1 Flow of Data

The flow of data in Google Analytics 4 involves several stages, starting from the data collection on a website or app to the reporting interface. Here is a step-by-step description of the data flow in GA4:

1. User Interaction:

Users interact with a website or mobile app, performing actions such as page views, clicks, form submissions, and other events.

2. GA4 Tracking Code:

The GA4 tracking code (also known as the Google Tag (gtag.js), is a piece of JavaScript code provided by Google Analytics 4), embedded in a website or app, captures user interactions. This code is provided by Google Analytics and is unique to individual GA4 property (a property represents a grouping of data from a website and/or app in Google Analytics).

3. Data Collection:

The GA4 tracking code collects data on user interactions and sends this data to the Google Analytics servers for processing.

4. Event and Parameter Configuration:

Within the GA4 property, a project/program area can configure events and parameters to track specific user interactions or gather additional information about users and their actions (such as through the use of the Exploration feature explained in Section 2.2.7).

5. User Identification:

Users are normally identified through a ClientID (please see Section 2.2.5 for more information). If you have implemented UserID tracking (UserID tracking is further explained in Section 2.2.5), GA4 can associate user interactions with specific user identities. This feature allows for cross-device tracking.

6. Data Aggregation:

GA4 aggregates the collected data, organizing it into meaningful reports and metrics. This includes information about user demographics, events, conversions, and more.

7. Reports and Insights:

The aggregated data is presented in the GA4 reporting interface. Here, you can access various reports, including user engagement, audience demographics, conversion tracking, and more. The reports provide insights into user behavior (i.e. pages visited, actions taken, and interactions with content such as clicks and conversions) on websites and apps.

8. Integration with Other Platforms:

GA4 can be integrated with other Google products and third-party platforms, such as Google Ads, Google BigQuery (BigQuery is a cloud data warehouse that can run highly performant queries of large datasets), and others. As an example, this can be done by linking GA4 and Google Ads accounts in the Admin tab, under Product Links. This integration allows for a more comprehensive analysis of user interactions and marketing efforts.

9. User Segmentation:

GA4 enables you to segment users based on various criteria, allowing you to analyze specific user groups separately. This segmentation can be based on demographics (if Google signals are activated (please see Section 2.2.6 for more on Google signals)), behavior, technology usage, and more.

10. Export and Sharing (Optional):

You have the option to export GA4 data or share specific reports with stakeholders. This can be done through the GA4 interface or by using Google Analytics APIs for more customized data extraction. (Note: GA4 data can be exported/downloaded as a PDF or CSV file). [1]

2.2.2 How GA4 is implemented

[1] Google Analytics Help < https://support.google.com/analytics/answer/9317657?hl=en >

Implementing Google Analytics 4 involves several steps, and the specific details may vary depending on whether you are implementing it on a website or a mobile app. Here are general guidelines for implementing GA4:

Implementation on a Website:

1. Create a Google Analytics 4 Property:
   - Log in to the Google Analytics 4 account.
   - Create a new GA4 property by following the setup wizard.

2. Obtain the GA4 Measurement ID:
   - After creating the property (within a property, you can view reports and manage data collection, attribution, privacy settings, and product links. An account can contain one or more properties), obtain the Measurement ID, a unique identifier for the GA4 property. It can be found in the Admin section of the GA4 property.

3. Add the GA4 Tracking Code:
   - Insert the GA4 tracking code into the <head> section of each page on the website. The tracking code is a JavaScript snippet provided by GA4.
   - The code typically includes a <script> tag to load the gtag.js library and a configuration script with the Measurement ID. The Measurement ID is a unique identifier for a web data stream and acts as a critical link, connecting a website to the corresponding data stream in Google Analytics 4.

4. Verify Implementation:
   - After adding the tracking code, it is important to visit the website to test whether GA4 is operating as necessary.

Implementation on a Mobile App:

1. Create a Google Analytics 4 Property:

- Similar to the website implementation, a new GA4 property must be created in the Google Analytics 4 account.

2. Integrate the GA4 SDK:

   - Integrate the GA4 SDK into the mobile app. The SDKs are available for various platforms, including iOS and Android.

3. Set Up GA4 Configuration in the App:

   - Configure the GA4 SDK with the Measurement ID and other settings in the app code.

4. Track Events and Screens:

   - Implement code to track events and screen views based on user interactions in the app.
   - The gtag function or specific SDK methods can be used to send events and screen views to GA4.

5. Verify Implementation:

   - Run the app on a device or simulator/emulator and use debug tools to ensure that events and screens are being recorded correctly.

It's essential to refer to the official Google Analytics 4 documentation for the most up-to-date information and detailed implementation instructions. Additionally, following guidelines and restrictions provided by Google's Terms and Conditions is strongly recommended, as failure to do so may pose many risks.

2.2.3 Privacy controls in GA4

GA4 offers the following privacy features to reduce any potential risks that may arise with its use. Google provides the instructions on how to set these up on their website.

1. Disable Google Analytics 4 data collection (web and app)

   a. GA4 can programmatically disable data collection from websites, Android and iOS apps.
   b. Visitors to the website can also opt out from data collection using the Google Analytics 4 opt-out browser add-on.

2. Disable collection of granular location and device data
   a. Google Analytics 4 collects this data by default. (Details on what data is collected is in Section 2.2.8)

3. Redact email addresses and user-defined URL query parameters
   a. The data-redaction feature helps to prevent the inadvertent collection of PII in the form of email addresses and URL query parameters. Data redaction uses text patterns to identify likely email addresses across all event parameters and the URL query parameters that are included as part of the event parameters page_location, page_referrer, page_path, link_url, video_url, and form_destination.

4. Disable collection of Google signals data
   a. When you activate Google signals (Please see Section 2.2.6), GA4 gives the option to enable or disable collection of those signals on a per-region basis.

5. Disable advertising features (web and app)
   a. GA4 can programmatically disable data collection for advertising features from websites, Android and iOS apps.

      Note: Google Analytics 4 Advertising Features is a collection of features that takes advantage of the Google advertising cookies which allow to do things like:

      - Create Remarketing Audiences (a remarketing audience is a list of cookies or mobile-advertising IDs that represents a group of individuals whose re-engagement may lead to a conversion) based on specific behavior (i.e. pages visited, actions taken, and interactions with content such as clicks and conversions), demographic, and interest data (based on cookies), and share those lists with Google Ads.
      - Use demographic and interest data in your Google Analytics 4 reports.

- Create Segments (a segment is a subset of analytics data that can be used in Explorations (a collection of advanced techniques that go beyond standard reports which provide deeper insights of analytics data, explained further in Section 2.2.7)) based on demographic and interest data.

6. Disable advertising personalization
   a. GA4 can programmatically disable data collection for advertising personalization from websites, Android and iOS apps.

      Note: Advertising personalization refers to the ability to tailor ads to website visitors' preferences. These preferences are based on the data that is collected from Google Analytics 4. Data that is collected through the Google Analytics 4 for Firebase (set of backend cloud computing services and application development platforms provided by Google) default implementation includes:
      - Number of users and sessions
      - Session duration
      - Operating systems
      - Device models
      - Geography
      - First launches
      - App opens
      - App updates
      - In-app purchases

7. Set the retention period for the Google Analytics 4 data that is collected
   a. For Google Analytics 4 properties, you can set the retention period of user-level data (i.e. user identifier data for each event, such as the ClientID (Please see Section 2.2.5) to:
      i. 2 months
      ii. 14 months
   b. For all other event data (e.g. pageviews, clicks, file downloads, etc.), you can set retention to:
      i. 2 months
      ii. 14 months
      iii. 26 months (Paid version (GA4 360) only)
      iv. 38 months (Paid version (GA4 360) only)

v. 50 months (Paid version (GA4 360) only)

2.2.4 IP Addresses, Location Data and Device Data

Google Analytics 4 does not log or store individual IP addresses.

As a default, GA4 provides coarse geo-location data by deriving the following metadata from IP addresses: City (and the derived latitude, and longitude of the city), Continent, Country, Region, Subcontinent (and ID-based counterparts (a place ID is a textual identifier that uniquely identifies a place)). These features can be disabled. If disabled, GA4 does not collect the following data:

- City
- Browser minor version
- Browser User-Agent string
- Device brand
- Device model
- Device name
- Operating system minor version
- Platform minor version
- Screen resolution

IP addresses are immediately discarded after location data is inferred and IP address data is never logged or stored on Google Analytics 4 databases.[2]

In Universal Analytics, Google provided the ability to mask IP addresses. The IP-masking feature in Universal Analytics set the last octet of IPv4 (32-bit address that is usually represented in dotted decimal notation, with a decimal value representing each of the four octets (bytes) that make up the address) user IP addresses and the last 80 bits of Ipv6 (the most recent version of the Internet Protocol (IP)) addresses to zeros in memory shortly after being sent to Google Analytics 4. The full IP address was never written to disk in this case.

2.2.5 Unique Identifiers

[2] Per Google Analytics Help website - <https://support.google.com/analytics/answer/12017362?hl=en#:~:text=Google%20Analytics%204%20does%20not%20log%20or%20store%20individual%20IP,and%20ID%2Dbased%20counterparts).>

GA4 assigns unique identifiers to distinguish unique users ("user" is in reference to device/browser) and their sessions on websites (via a ClientID or UserID) and to identify unique installations of apps and compute user metrics (via an app-instance ID).

- **For websites – ClientID**

  Google Analytics 4 stores a unique, randomly generated identifier called a ClientID in a first-party cookie named _ga to distinguish unique users (i.e. ClientID) and their sessions on websites. Technically, the ClientID identifies a specific device / web browser pairing and links all session / usage data generated by the same device / web browser pairing. Usage / session data continues to be collected under the same ClientID each time the same device / browser pairing returns to a website until the GA cookie is deleted by the user or it expires 2 years after the last website visit. The ClientID does not enable the identification of unique users across different browsers or devices. That is the purpose of the GA UserID and to an extent Google Signals.

- **For apps and mobile devices – App-Instance ID**

  Firebase SDK (Software Development Kit) is a set of tools, libraries, and APIs provided by Google as a comprehensive solution for developing web and mobile applications. Google Analytics 4 data is collected for mobile applications via the Google Firebase SDK. The Google Analytics 4 for Firebase SDK automatically generates and assigns an app-instance identifier to each instance of an app. This is known as the app-instance ID. Google Analytics 4 uses this ID to identify unique installations of an app and compute user metrics. If someone downloads an app on their phone and visits the app multiple times, the app-instance ID will stay the same. That's how it is able to capture repeated visits by the same user. If the same person downloads the same app on a different device like an iPad, then the app-instance ID will be different. If the same person deletes the app on their phone, then redownloads the app, it will generate a different app-instance ID. [3]

On iOS, the SDK uses the Advertising Identifier (IDFA) (random device identifier assigned by Apple to a user's device) as the app-instance ID if it is available. If it is not available, the SDK collects the Vendor Identifier (a unique, alphanumeric identifier assigned by Apple to all apps on a single device that are from the same publisher/vendor). If the IDFA becomes available later on, the SDK stops collecting the Vendor Identifier.

On Android, the SDK collects the Advertising ID (a unique, user-resettable ID for advertising, provided by Google Play services) by default. The Advertising ID is replaced with a string of zeros when a user opts out of personalization in their Android Settings.

- **UserID**

  Google Analytics 4 includes a UserID feature which is another unique identifier, beyond the ClientID, that makes it possible to associate session data across multiple browser / device pairings with a single user. For example, without a UserID, a user that visits ontario.ca on 3 separate occasions on 3 separate devices will appear as 3 separate, unrelated ClientIDs. With a UserID, it is possible to link those 3 separate ClientIDs to the same, single user. Google Analytics Help website does not provide a specific answer whether apps can also utilize this feature, but rather only refers to "different sessions and … various devices and platforms".[4]

2.2.6 Google signals

[3] January 24, 2024 – Email from Justin Duckett, with subject line, "RE: Extra notes on server side tagging and app instance id"

[4] Google Analytics Help website < https://support.google.com/analytics/answer/9213390?hl=en >

Google signals are session data from sites and apps that Google associates with users who have signed in to their Google accounts, and who have turned on Ads Personalization. Google signals is not automatically enabled. Activating Google signals in a GA4 property imports demographics and interest data from users who are signed in to a Google account when they visit a website provided they have consented to sharing this information (individual consent is turned on by default, but can be withdrawn by visiting My Ad Center page[5]) by enabling Ads Personalization within their Google account. In particular this includes:

- ○ Age bracket of the user (18-24, 25-34, 35-44, 45-54, 55-64, and 65+)
- ○ Gender of the user
- ○ The interests of the user (such as Arts & Entertainment, Games, Sports)

The inclusion of demographics and interests data can increase the risk that users could become identifiable. Google attempts to mitigate this through Data Thresholds[6] (a feature available in both signals and explorations (please see Section 2.2.6 and Section 2.2.7 respectively)) which withholds information where there are low user counts returned for a specified reporting date range. This is intended to prevent GA4 users from inferring the identity of individual users based on demographics, interests, or other signals present in the data. These thresholds are system-defined by Google and can't be adjusted by GA4 customers. Google has confirmed that they do not externalize the exact numbers of users needed before thresholding is disabled.[7]

Google signals also enables GA4 customers to identify users across different devices and sessions. When users turn on Ads Personalization, Google is able to develop a holistic view of how those users interact with an online property from multiple browsers and multiple devices. For example, GA customers can see how users browse a site from a phone, and later return on a tablet or laptop. In this way, it provides a similar capability to the UserID.

[5] My Ad Center < https://myadcenter.google.com/ >

[6] Google Analytics Help website < https://support.google.com/analytics/answer/9383630 >

[7] Google Analytics Help website < https://support.google.com/analytics/answer/9383630?hl=en >

2.2.7 Explorations

Explorations is a collection of advanced techniques that go beyond standard reports to help uncover deeper insights about user behavior.

The Explorations feature within GA4 enables organizations to pull event information for any given user identifier (i.e. ClientID) via the User Explorer report or the User Activity report. These features allow organizations to analyze and export event level data for a single user identifier, including how and when that user was acquired, summary metrics for that user, and a timeline of their activities on a site or an app.

User Explorer allows for the selection of specific groups of users, such as users who engaged with the GA4 property on both the app and website, and drill down on each individual user's activities.

OPS projects/program areas generally use analytics for aggregate analysis of website usage, i.e. not requiring a granular analysis of the activity of any specific user during the time spent on a website or an app. It is understood that using this functionality may be necessary in certain cases i.e. for troubleshooting. However, it must not be used to target individuals, to attempt to establish their identities or to take actions that affect users individually.

2.2.8 Data Collection

GA4 has a predetermined list of automatically collected events which are triggered by the basic user interactions on a website or an app. Please refer to the Appendix 6.3 for a full list of events.

This PIA recommends that the following parameters for data collection are set up as a default implementation of Google Analytics 4 across the OPS.

- The unique ClientID (websites) and app-instance ID (apps) identifiers – but **not** the UserID (see the GA4 functionality not approved section below).
- Users' approximate geographic location (city, country, region, continent, subcontinent) derived from their IP addresses.
- Information about users' devices including:
    - Web browser used (web implementations only)

- Browser minor version
- Browser user agent string
- Device brand
- device category
- device model (app implementations only)
- device name
- Operating system and minor version
- Platform minor version
- Screen resolution

- Users' language based on the language setting of the device OS (such as en-us or pt-br).
- Information relating to a user's activity.
  - The Automatically Collected Events (Appendix 6.3), the Optional Enhanced Measurement Events (Appendix 6.4) and the data collection for apps via Firebase (Appendix 6.5) are all approved by this PIA.
  - Custom Events (Appendix 6.6) are not approved by default. If OPS projects/program areas set up Custom Events in GA4, they should seek additional approval from their respective privacy office.
- App Information (only applicable for app implementations):
  - The app store from which the app was downloaded and installed
  - The App Version – versionName (Android) or the Bundle version (iOS)
  - New / Established parameter – New = user first opened the app within the last 7 days;
  - Established = User first opened the app more than 7 days ago.

2.2.9 Cross Device Tracking

The UserID feature lets you associate one's own identifiers with individual users so you can connect their behavior across different sessions and on various devices and platforms. Google Analytics 4 interprets each UserID as a separate user, which provides a more accurate user count and a more holistic story about a user's relationship with the website.

In Google Analytics 4 the UserID feature lets you manually assign your own identifiers with individual users so you can connect their behavior across different sessions and devices. To send UserIDs to Google Analytics 4, you need to create a unique ID for each user on your own and assign the IDs to your users. This is typically done when a user is logging in on a website/app.

2.2.10 Consent Mode

Consent mode lets you communicate your users' cookie or app identifier consent status to Google. Tags adjust their behavior and respect users' choices.

Consent mode interacts with the Consent Management Platform (CMP) or custom implementation for obtaining visitor consent, such as a cookie consent banner. Consent mode receives users' consent choices from your cookie banner or widget and dynamically adapts the behavior of Google Analytics 4, Ads, and third-party tags that create or read cookies.

When visitors deny consent, instead of storing cookies, tags send pings to Google. If you are using Google Analytics 4, Google fills the data collection gaps with conversion modeling and behavioral modeling.

Although Consent Mode is a feature that is available in GA4, there will be no need for its use in the near future within the context of the OPS.

2.2.11 Exporting Data

Google Analytics 4 data can be exported from GA4 through a series of reports. The file format is either CSV or PDF.

User-level and event-level data can also be exported to BigQuery for broader analysis.

2.2.12 Storage of Analytics Data

Google Analytics 4 uses regional data centers to ensure web and mobile app user measurements are sent to Google as quickly and securely as possible.

When Google Analytics 4 establishes a connection with the closest available Google data collection center (i.e., data centres that support Google's various enterprises, some of which host the GA4 analytics servers), measurement data is sent to Google over an encrypted HTTPS connection.

At the collection centers, the data is further encrypted before it's forwarded to Google Analytics 4 processing servers and made available to on the Google Analytics 4 platform.

It appears there are no data centers in Canada at this time, and most data centers are located in the USA. [8] (Note: Google Cloud does operate data centres across the world, including those in Toronto ON and Montreal QC, which support BigQuery and house its data.[9] This means that in the event a BigQuery Export is set up through GA4, a significant amount of both user-level and event-level raw data collected through GA4 can be exported to BigQuery.[10]

At this time, it appears that data collected by GA4 is stored on multiple computers, in different locations, thus not allowing for the possibility to select a specific data centre(s) to house the relevant data, quoting the loss of data due to “a single point of failure” as one of the reasons.[11]

[8] Google Data Centers < https://www.google.com/about/datacenters/locations/ >

[9] Google Cloud < https://cloud.google.com/about/locations >

[10] Google Analytics Help < https://support.google.com/analytics/answer/12769371?hl=en&ref_topic=9359001&sjid=3843367609826826903-NC >

[11] Google Data Centers < https://www.google.com/about/datacenters/data-security/ >

2.3 Differences between Universal Analytics and Google Analytics 4

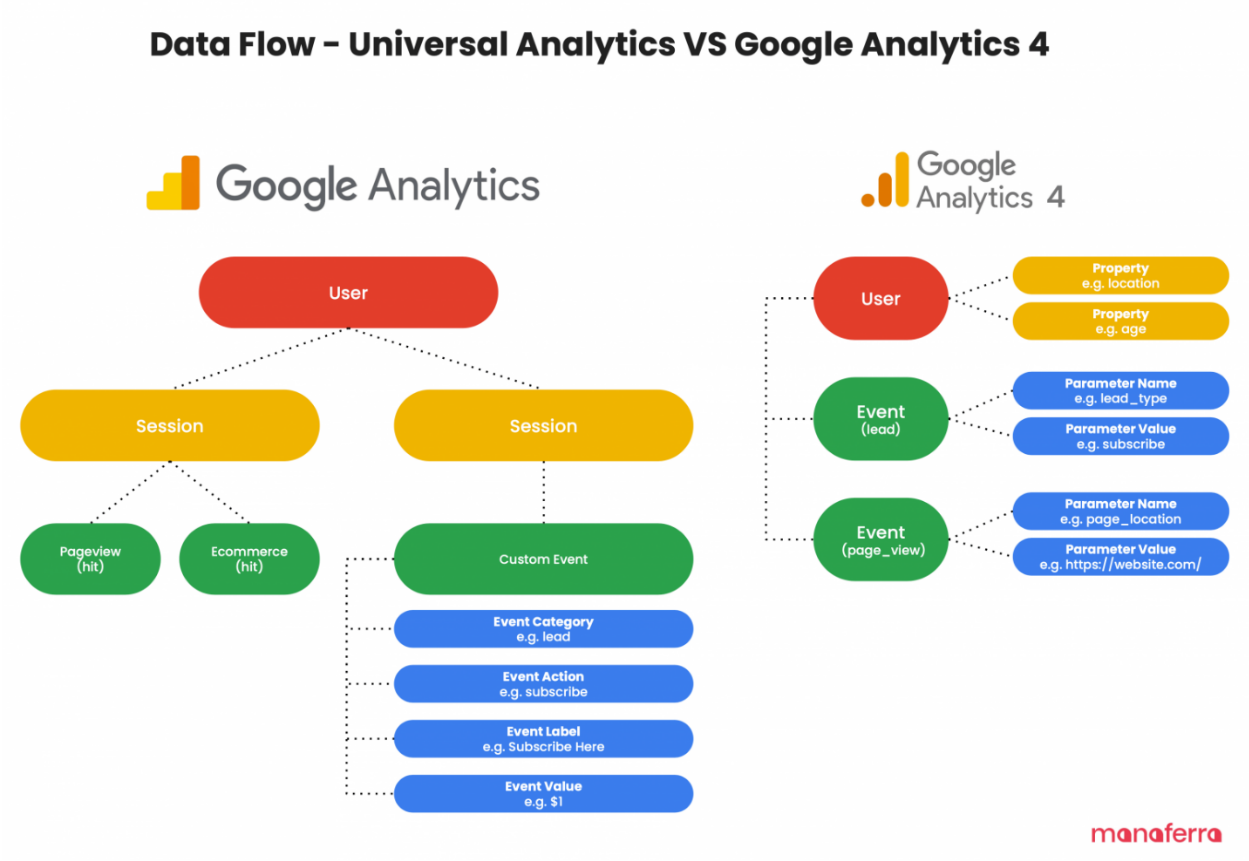

Data Flow - Universal Analytics VS Google Analytics 4
Google Analytics
User
Session
Session
Pageview (hit)
Ecommerce (hit)
Custom Event
Event Category e.g. lead
Event Action e.g. subscribe
Event Label e.g. Subscribe Here
Event Value e.g. $1
Google Analytics 4
User
Property e.g. location
Property e.g. age
Event (lead)
Parameter Name e.g. lead_type
Parameter Value e.g. subscribe
Event (page_view)
Parameter Name e.g. page_location
Parameter Value e.g. https://website.com/
manaferra

Figure 1 - Data Flow - Universal Analytics VS Google Analytics 4[12]

Data Model:

[12] Graphic from Manaferra < https://www.manaferra.com/google-analytics-4-for-universities/ >

Universal Analytics (GA4's predecessor) uses a session-based data model, which means it tracks user interactions within a specific session or visit to a website. It focuses on metrics like pageviews, sessions, and conversions.

Google Analytics 4 uses an event-based data model. Instead of focusing on sessions, it tracks individual events or actions performed by users on a website or app. These events can include clicks, form submissions, video views, and more. GA4 provides a more granular understanding of user behavior beyond traditional pageviews.

User-Centric Approach:

Universal Analytics primarily relies on third-party cookies to identify and track individual users. It assigns a unique identifier, known as the ClientID, to each user's device or browser. However, it doesn't easily connect user interactions across different devices or platforms.

GA4 takes a more user-centric approach by using multiple signals like first party cookies, ClientIDs, and other identifiers to track users across devices and platforms. It aims to provide a more complete picture of user behavior by consolidating data from different touchpoints.

Reporting and Analysis:

In Universal Analytics, reporting primarily revolves around predefined reports and dimensions. Customization options are available but may require additional implementation effort.

GA4 introduces a more flexible and customizable reporting structure. It offers a "BigQuery Export" feature that allows users to export raw, unaggregated data to Google BigQuery for advanced analysis and custom reporting. This provides more opportunities for in-depth analysis and data exploration.

Privacy and Consent:

Google provides very limited information on the specific details on how it handled data collected by Universal Analytics but does quote privacy features such as IP masking and browser opt-out ad-on.[13]

GA4, on the other hand, provides a clearer picture of what it offers with its built-in privacy features. It provides options for website owners to configure data collection

[13] Google Analytics Help < https://support.google.com/analytics/answer/2838718?hl=en >

settings, including cookie consent controls and data retention periods. It also offers more flexibility in terms of respecting user preferences and compliance with privacy regulations.[14]

Data Retention:

Data retention in Universal Analytics is subject to the account's settings and can range from a minimum of 14 months to indefinite retention.

GA4 allows users to configure their data retention settings more granularly. It offers options to set data retention periods ranging from 2 months to 50 months, providing organizations with more control over how long their data is retained. Data collected from GA4 can be downloaded onto a computer to be used offline for retention purposes.

In summary, Google Analytics 4 introduces a more advanced data model, focuses on user behavior and events, provides better cross-device tracking, and incorporates built-in privacy features compared to Universal Analytics. The goal is to provide a more comprehensive understanding of user behavior, improve insights, and align with evolving privacy standards.

[14] Google Analytics Help < https://support.google.com/analytics/answer/9019185?hl=en#zippy=%2Cin-this-article >

[This page intentionally blank]

3. Analysis

3.1 General Comments

This privacy analysis assesses the privacy implications of Google Analytics 4, a widely used web analytics platform provided by Google. GA4 enables website and app owners to analyze user behavior, measure website performance, and gain insights for marketing optimization. This analysis aims to evaluate the privacy features, data handling practices, and potential risks associated with GA4 implementation.

GA4 collects various types of data, including user interactions, events, conversions, and device information. The primary purpose is to analyze user behavior, improve website performance, and measure marketing effectiveness. While GA4 provides flexibility for customization, website owners must be mindful of data minimization principles and ensure that only necessary data is collected to fulfill specific objectives. The data collected by Google Analytics 4 is not to be used to collect personally identifiable information. Additionally, the data is not to be used for personalisation purposes i.e. to tailor what an individual user sees based on an user-level analysis of their analytics traffic.

As GA4 has only been released for less than a year, it is important to keep in mind that not all information that is available can be relied upon with absolute certainty without extensive prior testing of the product by the community at large. Much like with any new product, or an update to one, it may have its vulnerabilities and bugs that may have not yet been discovered. Therefore, vigilance is crucial and all projects/program areas using GA4 must regularly inspect its operational integrity.

Website visitors may be notified of the collection of analytics data by the website owner through their privacy policy, cookie consent banner, GA4 Consent Mode, user notification in mobile apps, and through other methods. This is an important step in promoting transparency and fostering trust.

GA4 allows for cross-device tracking through the UserID feature, helping to better understand user behavior across different devices. Although GA4 aims to ensure that individual users cannot be personally identified across devices, this elevates the risk nevertheless.

3.1.1 Risks of identification and reidentification

One of the biggest privacy considerations with any project is the risk of an individual being identified or reidentified based on the data that is collected. GA4 has three main identifiers: ClientID, app-instance ID and UserID. ClientID and app-instance ID are randomly generated strings that act as a pseudonymous identifier and pseudonymously identifies a browser instance. These get stored in the browser's cookies or the app itself, so subsequent visits to the same site or app can be associated with the same user. On the other hand, UserID presents a cross-platform, cross-device view of how users interact with an app or a website. Therefore, the risk of being identified or reidentified is much higher with the UserID feature enabled, as it tracks data across different platforms and devices.

Additionally, Google signals, which imports demographics and interest data from users who are signed in to a Google account when they visit a website provided they have consented to sharing this information can pose similar risks of being identified, as certain key details can be extrapolated from their profile leading to identification/reidentification.

Deploying GA4 using Google Tag Manager (GTM) introduces additional privacy and security risks including:

- The top permission level ("Publish") enables a user to create, edit and publish which means there is no segregation of responsibilities for users with that role.
- GTM simplifies tag deployment and therefore there is a risk that "tags" can be deployed to introduce tracking functionality, such as social media retargeting, that is outside of the recommended scope i.e. web analytics.
- If the Google Tag is used, GTM provides an option to collect user-provided data such as email addresses, names, addresses, phone numbers etc. Please see Appendix 6.2 for more information about this feature.

GA4 assigns unique identifiers to distinguish unique users ("user" is in reference to device / browser) and their sessions on websites (via a ClientID) and to identify unique installations of apps and compute user metrics (via an app-instance ID). It is not generally possible to establish users' real-life identities via GA4 (assuming any personal data in page titles or URLs is redacted). All OPS projects/program areas which plan to use GA4 should consider any risk scenarios whereby it might be possible to do this. For example, systems may collect other personal data via web forms or account sign ups that could be matched to the GA data to establish individuals' real-life identities.

3.2 Benefits of OPS-wide standardization of GA4 parameters

This PIA should serve as a guideline on best practices for the use of GA4 by any OPS project/program area. That being said, each individual project/program area bears full responsibility for their use of GA4 and must take every reasonable step to protect individual privacy.

The MPBSD Privacy Office always strives to provide the most objective and constructive advice on all projects that come into contact with personal information. Many ministry projects and program areas may choose to use Google Analytics 4, therefore implementing a standard approach for the use of Google Analytics 4 across the organization offers a number of benefits for all OPS stakeholders using this product.

Some of the benefits are:

1. Consistency and Efficiency: Establishing a standardized approach ensures consistency in the implementation and usage of Google Analytics 4 throughout the OPS. This means that all departments follow the same guidelines, settings, and reporting practices, unless otherwise requested. It leads to a unified understanding of metrics, data collection, and reporting methodologies. It streamlines workflows, reduces duplication of efforts, and enhances productivity.

2. Accurate and Reliable Insights: A standard approach ensures that the right tracking codes, configurations, and measurement goals are consistently applied across websites, apps, or digital properties. This results in accurate and reliable insights into user behavior, website performance, and marketing effectiveness. By aligning tracking methods and metrics, organizations can trust the data generated by Google Analytics 4 to make data-driven decisions. Consistency in data collection and reporting enhances the organization's ability to measure and improve key performance indicators (KPIs), identify trends, and optimize marketing campaigns effectively.

3. Streamlined Governance and Compliance: A standardized approach to using Google Analytics 4 facilitates streamlined governance and compliance with privacy legislations and internal policies. By defining clear guidelines, organizations can ensure that data collection and usage adhere to legal and ethical standards. It enables consistent implementation of data protection measures, such as anonymization and user consent mechanisms, in line with applicable regulations like the Freedom of Information and Protection of Privacy Act (FIPPA). Standardization also aids in maintaining data security practices, access controls, and audit trails, ensuring compliance with internal and external requirements.

4. Knowledge Sharing and Collaboration: With a standard approach to Google Analytics 4, knowledge sharing and collaboration across ministries will be much easier. Stakeholders can learn from each other's experiences, best practices, and insights gained from using Google Analytics 4 in different departments or projects. This promotes a culture of data-driven decision-making and empowers individuals to leverage analytics effectively. By sharing reports, dashboards, and insights generated from Google Analytics 4, OPS projects/program areas can align their goals, strategies, and actions, resulting in improved performance, enhanced marketing campaigns, and better customer experiences.

5. Scalability and Growth: As the government expands its digital presence, launches new websites or apps, or explores new marketing channels, having a predefined approach ensures that Google Analytics 4 can be seamlessly integrated into these new initiatives. It reduces the learning curve for OPS projects/program areas involved and allows for efficient tracking, analysis, and reporting from the outset. Standardization enables the organization to scale its analytics capabilities, adapt to changing business needs, and leverage data to drive continuous improvement and innovation.

In summary, establishing a standard approach for the use of Google Analytics 4 across the OPS brings consistency, efficiency, accuracy, governance, collaboration, and scalability. It ensures that data is collected and analyzed consistently, leading to reliable insights that support informed decision-making.

4. Recommendations

We recommend that projects that use GA4 use the following configurations for default implementation.

4.1 Primary Recommendations

The following recommendations on implementation of GA4 to limit the impact on privacy. However, we remind that every project/program area is responsible for their own use of GA4 and should consult with their own privacy office regarding their own implementation of GA4.

1. **JavaScript Code in GA4**

To make Google Analytics 4 ‘function’, JavaScript code is downloaded and executed from Google. It may be potentially possible in the future that this executable code could be updated to change the functionality of Google Analytics in a privacy-invasive manner.

While not specific to any specific project or implementation, we recommend that regular monitoring and testing of Google Analytics 4 be undertaken by GSIC to ensure that no privacy invasive functionality is introduced in the future because of changes made by Google to the JavaScript code that is downloaded and executed to implement Google Analytics 4. Testing should be performed at regular intervals to confirm and verify the flow of information and continued expected functionality of Google Analytics 4.

2. **Certain GA4 Functionality should be limited**

We recommend against the use of the following GA4 features. If any OPS projects/program areas wish to use these additional features, they must seek additional advice and approval from their respective privacy office:

- **Google signals**

  Activating Google signals in a GA4 property imports demographics and interest data from users who are signed in to a Google account when they visit a website provided they have consented to sharing this information by enabling Ads Personalization within their Google account. In particular this includes:

  - Age bracket of the user (18-24, 25-34, 35-44, 45-54, 55-64, and 65+)
  - Gender of the user
  - The interests of the user (such as Arts & Entertainment, Games, Sports)

- **UserID**

  Google Analytics 4 includes a UserID feature which is another unique identifier, beyond the ClientID, that makes it possible to associate session data across multiple browser / device pairings with a single user. For example, without a UserID, a user that visits ontario.ca on 3 separate occasions on 3 separate devices will appear as 3 separate, unrelated ClientIDs. With a UserID, it is possible to link those 3 separate ClientIDs to the same, single user.

- **Custom Events**

  Custom Events collect additional information that GA4 does not collect automatically. They are defined manually so you can collect information about an interaction that's important to your business. They are made up of an event name, parameter names and parameter values. Please refer to section 6.6 for a full description of custom events.

An example of a custom event may include tracking a field error, so you can understand and fix pain points for your users. This custom event might include the following name and parameters:

event name: field_error

parameter 1 name: fieldName

parameter 1 value: a value is passed in with the text of the field title eg *'Search for a service*'

parameter 2 name: errorMessage

parameter 2 value: a value is passed in with the text of the error description eg '*Field must not be empty'*

Custom events can pose a privacy risk if not managed properly. For instance, a custom event can collect Personally Identifiable Information (PII) if not set up properly. Implementing a custom event in GA4 is only recommended following these best practices:

- All of Google's recommendations to avoid sending PII are followed. (see https://support.google.com/analytics/answer/6366371)
- No page URL's or page titles that contain PII are collected as parameter values in custom events.
- Website visitors sometimes enter PII into search boxes and form fields. No fields that ask for PII are collected as parameter values in custom events. Eg names, email addresses, credit card numbers, etc.
- In the event that any OPS projects/program areas may require to set up additional events in GA4 and are unsure if it meets privacy requirements or if the event requires the collection of PII, they should seek additional advice from their respective privacy office.

3. **Privacy concerns related to URLs**

We recommend that all OPS projects/program areas verify via testing that all personal data is being successfully removed from page titles, URLs or any other place where it may appear. PII includes information such as names, emails, etc.

If PII is passed in URLs, Google recommends two solutions:

Updating URL schemes. Developers of the site should remove the PII from the link and use identifiers or tokens to associate the verification email with the user account.

Update event tracking. If there is any possibility of your URLs, URL parameters, or titles containing PII, you'll need to remove it. You can add analytics.js code to change the URL before it gets sent to Google Analytics 4.

Any personal data in page titles or URLs must be redacted before being sent to GA4. For example a URL might be in the form of *www.ontario.ca/update_details/joe_smith/confirmation* and the Title field might be "Update Details Joe Smith Confirmed".

Please see Appendix 6.1 where GA4 provides instructions on how to do this. Projects/program areas must verify via testing that all personal data is being successfully removed from page titles, URLs or any other place where it may appear. OPS projects/program areas must also implement reasonable controls to detect if personal data is being included in the analytics data and handle such events as security incidents in accordance with their incident management processes.

4. **Google Tag Manager**

Google Tag Manager (GTM) is a free tag management system provided by Google. It allows website owners and marketers to manage and deploy various tracking codes, analytics tools, and other snippets of code on their websites without directly modifying the website's source code.

GTM allows for different levels of user access, which can pose a risk if not managed properly. For instance, a user with too much access can accidentally or maliciously modify tags, leading to incorrect data collection, website functionality disruption, or exposure to sensitive data.

Deploying GA4 via GTM is recommended with the implementation of the following:

- Custom HTML, Custom JavaScript and all tags other than Google Analytics 4 tags should be restricted by implementing a code snippet on webpages.

```
<script>
window.dataLayer = window.dataLayer || [];
dataLayer.push({
  'gtm.allowlist': ['<id>', '<id>', ...],
  'gtm.blocklist': ['<id>', '<id>', '<id>', ...]
});
</script>
```

- This code snippet should not be editable by the GTM users. Therefore, any change to the tracking functionality on a web page will require GTM users to configure it in GTM and another group (probably developers) to change the code snippet.
- The number of Administrators and users with “Publish” permissions should be kept to an absolute minimum.
- MFA (Multi Factor Authentication) should be enabled on all GTM accounts.
- Applicable Privacy Notices and Cookie Policies must clearly state that users are only tracked for analytics purposes.
- OPS projects/program areas must ensure that their GTM users are made aware of the restrictions around what GTM can be used for i.e., analytics only.
- The settings that enable the collection of user-provided data such as email addresses, names, addresses, phone numbers etc. must not be enabled.

4.2 Additional Recommendations

1. We recommend that all OPS projects/program areas conduct periodic audits of GA4 configurations as well as deployment within any website or app. This should include verifying that the tracking code is correctly implemented on all relevant pages or screens.

2. We recommend that GA4 accounts are set up with individual permissions based on roles and responsibilities, which are reviewed and managed regularly, ensuring that only authorized individuals have access to the GA4 property, and that their access levels are appropriate for their roles.

3. We recommend that all projects/program areas use tools such as GA4 DebugView to test and debug, to ensure that events and data are being tracked correctly, especially after making changes to a website or an app.

4. We recommend that all projects/program areas remain up-to-date with new features and updates to GA4 both prior to starting new projects and also while maintaining existing ones. This is critical in ensuring the proper use and maintenance of the analytics tool.

5. We recommend staff involved with the implementation and maintenance should receive training or knowledge transfer from/to relevant team members so that they understand how to navigate and use GA4 effectively. We recommend that all staff that will be working with GA4 receive the appropriate training and coaching to ensure proper understanding of the product.

6. We recommend keeping track of changes in privacy legislations and regulations and ensuring that the GA4 implementation complies with relevant data protection laws at all times.

7. We recommend maintaining documentation that outlines the GA4 configuration, tracking code placements, customizations, and any other relevant details will contribute towards a more accountable and responsible approach of using GA4. This documentation can also be valuable for onboarding new team members and troubleshooting issues, if any arise.

8. We recommend that all project/program areas that plan on using GA4 must review this overarching GA4 PIA.

9. We recommend that all project/program areas using GA4 need to conduct their own privacy analysis and consult with their respective privacy office as appropriate. A template (see Section 5.) has been provided to assist with this process for inclusion in any project specific PIA.

10. Any project/program area is responsible for their own privacy compliance and how they use GA4, and they have the responsibility to inform their decision makers and obtain the necessary approvals for any risks specific to their own GA4 implementation.

5. Template for other PIAs

The following template has been prepared to assist projects/program areas that will be using Google Analytics 4 and provides a list of recommendations to assist with the appropriate implementation and maintenance of the analytics tool. This template may be inserted into any project specific PIA to record GA4 use-case configuration.

#	Feature	Recommendation	Recommendation Followed? (Y/N) / Comments
		Privacy Invasive Functionality	
1	Google signals	We recommend against using Google signals.	
3	UserID	We recommend against using UserID.	
4	Custom Events	If Custom Events are required, follow best practices: 1. Follow Google's recommendations to avoid sending PII are followed. (see https://support.google.com/analytics/answer/6366371) 2. No page URL's or page titles that contain PII are collected as parameter values in custom events. 3. No fields that ask for PII are collected as parameter values in custom events. Eg names, email addresses, credit card numbers, etc. 4. Consult your privacy office if unsure as to whether this meets privacy requirements and/or if the event requires the collection of PII.	

5	PII in URLs	We recommend that all projects/program areas verify via testing that all personal data is being successfully removed from page titles, URLs or any other place where it may appear. PII includes information such as names, emails, etc.	
6	Google Tag Manager	We recommend that all projects/program areas deploy GA4 via GTM with the implementation of the code snippet provided in Section 4.1. We also recommend the following: 1. The number of Administrators and users with “Publish” permissions should be kept to an absolute minimum. 2. MFA (Multi Factor Authentication) should be enabled on all GTM accounts. 3. Applicable Privacy Notices and Cookie Policies must clearly state that users are only tracked for analytics purposes. 4. OPS projects/program areas must ensure that their GTM users are made aware of the restrictions around what GTM can be used for i.e., analytics only. 5. The settings that enable the collection of user-provided data such as email addresses, names, addresses, phone numbers etc. must not be enabled.	
Additional Recommendations			
7	Audits	We recommend that all OPS projects/program areas conduct periodic audits of GA4 configurations as well as deployment within any website or app. This should include verifying that the tracking code is correctly implemented on all relevant pages or screens.	
8	Individual permissions and role-based access	We recommend that GA4 accounts are set up with individual permissions based on roles and responsibilities, which are reviewed and managed regularly, ensuring that only authorized individuals have	

		access to the GA4 property, and that their access levels are appropriate for their roles.	
9	Testing and debugging	We recommend that all projects/program areas use tools such as GA4 DebugView to test and debug, to ensure that events and data are being tracked correctly, especially after making changes to a website or an app.	
10	Keeping up to date with GA4	We recommend that all projects/program areas remain up-to-date with new features and updates to GA4 both prior to starting new projects and also while maintaining existing ones. This is critical in ensuring the proper use and maintenance of the analytics tool.	
11	Training and awareness	We recommend staff involved with the implementation and maintenance should receive training or knowledge transfer from/to relevant team members so that they understand how to navigate and use GA4 effectively. We recommend that all staff that will be working with GA4 receive the appropriate training and coaching to ensure proper understanding of the product.	
12	Legislative framework	We recommend keeping track of changes in privacy legislations and regulations and ensuring that the GA4 implementation complies with relevant data protection laws at all times.	
13	Documentation	We recommend maintaining documentation that outlines the GA4 configuration, tracking code placements, customizations, and any other relevant details will contribute towards a more accountable and responsible approach of using GA4. This documentation can also be valuable for onboarding new team members and troubleshooting issues, if any arise.	
14	Review of PIA	We recommend that all project/program areas that plan on using GA4 must review the overarching GA4 PIA.	

15	Individual privacy analysis	We recommend that all project/program areas using GA4 need to conduct their own privacy analysis and consult with their respective privacy office as appropriate.	
16	Approvals	Any project/program area is responsible for their own privacy compliance and how they use GA4, and they have the responsibility to inform their decision makers and obtain the necessary approvals for any risks specific to their own GA4 implementation.	

6. Appendix

6.1 Best practices to avoid sending Personally Identifiable Information (PII)

Avoid sending PII to Google when collecting Analytics data.

To protect user privacy, Google policies mandate that no data be passed to Google that Google could use or recognize as personally identifiable information (PII). PII includes, but is not limited to, information such as email addresses, personal mobile numbers, and social security numbers. Because laws across countries and territories vary, and because Google Analytics can be used in many ways, consult an attorney if you are in doubt whether certain information might constitute PII or not.

Learn more about what Google considers PII.

When implementing Analytics on a property, follow the best practices in this article to reduce the risk of passing PII to Google.

In this article:

UserIDs

Page URLs and titles

PII entered by users

Data Import

Analytics features and privacy risk

Geolocation

AdSense

UserIDs

Before using UserIDs, read Best practices for UserID

Page URLs and titles

The basic Analytics page tag collects the page URL and page title of each page that is viewed. PII is often inadvertently sent in these URLs and titles. Both the URL path and parameters must be free of PII. If there is any possibility of your URLs, URL parameters, or titles containing PII, you'll need to remove it.

You can configure data redaction in Analytics to remove email addresses on a best-effort basis and to remove URL query parameters that you specify. Data redaction is configured in the Admin section of Analytics and doesn't require that you write any code. It is only available for web data streams. Learn more about Data redaction.

You can also add analytics.js code to change the URL before it gets sent to Analytics. For example, to alter the URL to "example.com/example?a=b":

```
ga('set', 'location', 'http://example.com/example?a=b');
```

See the developer reference.

Similarly, you can alter the page title before it gets sent to Analytics. For example, to change the title to "New Title":

```
ga('set', 'title', 'New Title');
```

See the developer reference.

There are additional strategies to avoid sending PII through URLs. To learn more, read Best practices to avoid sending PIIin the AdSense help center.

PII entered by users

Website visitors and users sometimes enter PII into search boxes and form fields. Be sure to remove PII from user-entered information before it is sent to Analytics.

Data Import

Read the Upload data use policy before using Data Import or uploading data to Analytics.

Analytics features and privacy risk

Special care should be taken to ensure no PII such as names, social security numbers, email addresses, or any similar personal identifiers, or data that permanently identifies a particular device such as a mobile phone's unique device identifier (if such an identifier cannot be reset) is sent to Analytics when using these following features:

UserID override

All custom dimensions

Campaign dimensions: Source, Medium, Keyword, Campaign, Content

Be sure not to include PII in custom campaign parameters utm_source, utm_medium, utm_term, utm_campaign, and utm_content.

Site search dimensions: Site Search Term and Site Search Category

Event dimensions: Event Category, Event Action, Event Label

Geolocation

If collecting geolocation info, ensure it is not GPS or fine-grained location information, as this could lead to reasonable inference of the individual. "Fine-grained location" information for Analytics is defined as any area less than 1 square mile, including any lat/long data. In some instances, such as in the UK, zip code can map to a single residence and thus cannot be passed to Analytics.

AdSense

If you use AdSense, read and follow the Best practices to avoid sending PII in the AdSense help center.

Source: < https://support.google.com/analytics/answer/6366371?hl=en#zippy=%2Cin-this-article >

6.2 About enhanced conversions

Note: The global site tag (gtag.js) is now the Google tag. With this change, new and existing gtag.js installations will get new capabilities to help you do more, improve data quality and adopt new features – all without additional code. Learn more About the Google tag.

Enhanced conversions is a feature that can improve the accuracy of your conversion measurement and unlock more powerful bidding. It supplements your existing conversion tags by sending hashed first-party conversion data from your website to Google in a privacy-safe way. The feature uses a secure one-way hashing algorithm called SHA256 on your first-party customer data, such as email addresses, before sending to Google. You can learn more about Google's conversion modelling solutions.

You can set up enhanced conversions using the Google tag, Google Tag Manager or Google Ads API.

Note: Google is committed to protecting the confidentiality and security of your data. We'll keep your data confidential and secure using the same industry-leading standards we use to protect our own users' data. We only report aggregated conversions. You can read more about enhanced conversions customer data policies and how we use your data.

When a customer completes a conversion on your website, you may receive first-party customer data such as an email address, name, home address and/or phone number. This data can be captured in your conversion tracking tags, hashed, sent to Google in its hashed form and then used to enhance your conversion measurement.

Depending on which type of enhanced conversions you use, the hashed data will be used in different ways to improve your measurement:

Enhanced conversions for web	Enhanced conversions for leads
Relevant for advertisers who want to track sales and events that happen on a website.	Relevant for advertisers who want to track sales that happen off a website (for

Enhanced conversions for web	Enhanced conversions for leads
	example, phone or email) from website leads.
Improves measurement of online conversions.	Improves measurement of offline transactions that came from a website lead or visitor.
Allows you to send hashed first-party, user-provided data from your website when a user converts. The data is then used to match your customers to Google accounts, which were signed-in to when they engaged with one of your ads.	Allows you to use hashed, first-party user-provided data from your website lead forms for offline lead measurement. When you upload your leads, the provided hashed information is used to attribute back to the Google Ad campaign.

Source: < https://support.google.com/google-ads/answer/9888656?hl=en-GB >

6.3 Automatically collected events

Automatically collected events are triggered by basic interactions with your app and/or site (as indicated under the event name in the table below). As long as you use the Google Analytics for Firebase SDK or gtag.js, you don't need to write any additional code to collect these events.

Analytics collects events for Android ***and*** iOS apps unless otherwise stated. The names and parameters of these events can be helpful when accessing your raw event data in BigQuery.

Note

The following parameters are collected by default with every event, including custom events. The automatically collected events that don't have parameters listed in the table below only receive the following parameters:

- ***language***
- ***page_location***
- ***page_referrer***
- ***page_title***
- ***screen_resolution***

The value assigned to event parameters must be 100 characters or fewer.
The ***page_title*** parameter must be 300 characters or fewer.
The ***page_referrer*** parameter must be 420 characters or fewer.
The ***page_location*** parameter must be 1,000 characters or fewer.

If you override the ***page_location*** parameter, make sure that the URL path is valid. If you assign an invalid URL path, the ***Page location*** dimension will be empty. You can use the Campaign URL Builder to check whether a URL path is valid.

Event	**Automatically triggered...**	**Parameters**
ad_click (app)	when a user clicks an ad Publisher events coming from AdMob via the Google Mobile Ads SDK or Ad Manager via the Ad Manager integration This event is not exported to BigQuery.	ad_event_id
ad_exposure (app)	when at least one ad served by the Mobile Ads SDK is on screen This event does not appear in reports and is not exported to BigQuery.	firebase_screen, firebase_screen_id, firebase_screen_class, exposure_time
ad_impression (app)	when a user sees an ad impression Publisher events coming from AdMob via the Google Mobile Ads SDK or Ad Manager via the Ad Manager integration This event is not exported to BigQuery.	ad_event_id, value*
ad_query (app)	when an ad request is made by the Mobile Ads SDK This event does not appear in reports and is not exported to BigQuery.	ad_event_id

ad_reward (app)	when a reward is granted by a rewarded ad served by the Mobile Ads SDK	ad_unit_code, reward_type, reward_value
adunit_exposure (app)	when an ad unit served by the Mobile Ads SDK is on screen This event does not appear in reports and is not exported to BigQuery.	firebase_screen, firebase_screen_id, firebase_screen_class, exposure_time
app_clear_data (app)	when the user resets/clears the app data, removing all settings and sign-in data Android only	
app_exception (app)	when the app crashes or throws an exception The event is sent when you integrate Firebase Crashlytics.	fatal, timestamp, engagement_time_msec
app_remove (app)	when an application package is removed (uninstalled) from an Android device Android only This event is different from the ***Daily uninstalls by device*** and ***Daily uninstalls by user*** metrics, which are both reported by Google Play Developer Console. The ***app_remove*** event counts the removal of application packages, regardless of the installation source, and the count changes depending on the date range you are using for the report. The ***Daily uninstalls by device*** and ***Daily uninstalls by user*** metrics count the removal of application packages only when they were installed	

	from Google Play, and are reported on a daily basis.	
app_store_refund (app)	when an in-app purchase is refunded by Google Play Android only This event is not exported to BigQuery.	product_id, value, currency, quantity
app_store_ subscription_cancel (app)	when a paid subscription is cancelled in Google Play Android only Requires an initial subscription that was made on or after July 1, 2019. This event is not exported to BigQuery.	product_id, price, value, currency, cancellation_reason
app_store_ subscription_convert (app)	when a free-trial subscription is converted to a paid subscription This event is set as a default conversion. Requires an initial subscription that was made on or after July 1, 2019. An initial free-trial subscription is logged as an in_app_purchase with the subscription parameter set to true. This event is not exported to BigQuery.	product_id, price, value, currency, quantity
app_store_ subscription_renew (app)	when a paid subscription is renewed This event is set as a default conversion.	product_id, price, value, currency, quantity, renewal_count

	Requires an initial subscription that was made on or after July 1, 2019. This event is not exported to BigQuery.	
app_update (app)	when the app is updated to a new version and launched again The previous app version id is passed as a parameter. This event is conceptually different from the Daily upgrades by device metric, which is reported by Google Play Developer Console. An upgrade refers to the updating of the application binary, whereas an app_update event is triggered upon the subsequent launch of the upgraded app.	previous_app_version
click (web)	each time a user clicks a link that leads away from the current domain By default, outbound click events will occur for all links leading away from the current domain. Links to domains configured for cross-domain measurement will not trigger outbound click events. The parameters populate the following dimensions: • ***Link classes*** (from link_classes) • ***Link domain*** (from link_domain) • ***Link ID*** (from link_id) • ***Link URL*** (from link_url)	link_classes, link_domain, link_id, link_url, outbound (boolean)

	- ***Outbound*** (from outbound) Collected by default via enhanced measurement.	
dynamic_link_app_open (app)	when a user re-opens the app via a dynamic link **Note**: This event is being phased out as Firebase Dynamic Links is currently deprecated. Learn more about these changes.	source, medium, campaign, link_id, accept_time
dynamic_link_ app_update (app)	when the app is updated to a new version and is opened via a dynamic link Android only **Note**: This event is being phased out as Firebase Dynamic Links is currently deprecated. Learn more about these changes.	source, medium, campaign, link_id, accept_time
dynamic_link_ first_open (app)	when a user opens the app for the first time via a dynamic link **Note**: This event is being phased out as Firebase Dynamic Links is currently deprecated. Learn more about these changes.	source, medium, campaign, link_id, accept_time
error (app)	logged in place of an event that can't be logged because it is invalid in some way _err (firebase_error), _ev (firebase_error_value), and _el (firebase_error_length) parameters have additional information. This event does not appear in reports and is not exported to BigQuery.	firebase_error, firebase_error_value

file_download (web)	when a user clicks a link leading to a file (with a common file extension) of the following types: • document • text • executable • presentation • compressed file • video • audio This event is collected by default via enhanced measurement. See the file extensions that trigger the event	file_extension, file_name link_classes, link_id, link_text, link_url
firebase_campaign (app)	when the app is launched with campaign parameters	source, medium, campaign, term, content, gclid, aclid, cp1, anid, click_timestamp, campaign_info_source
firebase_in_app_ message_action (app)	when a user takes action on a Firebase In-App Message	message_name, message_device_time, message_id
firebase_in_app_ message_dismiss (app)	when a user dismisses a Firebase In-App Message	message_name, message_device_time, message_id
firebase_in_app_ message_impression (app)	when a user sees a Firebase In-App Message	message_name, message_device_time, message_id
first_open (app)	the first time a user launches an app after installing or re-installing it This event is not triggered when a user downloads the app onto a	previous_gmp_app_id, updated_with_analytics, previous_first_open_count, system_app, system_app_update, deferred_analytics_collection,

	device, but instead when he or she first uses it. To see raw download numbers, look in Google Play Developer Console or in iTunesConnect. Supports measuring first_open conversions for users who accept Apple's iOS 14 app-tracking prompt.	reset_analytics_cause, engagement_time_msec
first_visit (app, web)	the first time a user visits a website or launches an Android instant app with Analytics enabled	
form_start (web)	the first time a user interacts with a form in a session Note: You can only use the parameters in your reports if you create custom dimensions for them. Collected by default via enhanced measurement.	form_id, form_name, form_destination
form_submit (web)	when the user submits a form Note: You can only use the parameters in your reports if you create custom dimensions for them. Collected by default via enhanced measurement.	form_id, form_name, form_destination, form_submit_text
in_app_purchase (app)	when a user completes an in-app purchase, including an initial subscription, that is processed by the Apple App Store or Google Play Store	product_id, price, value, currency, quantity, subscription, free_trial, introductory_price

	The product ID, product name, currency, and quantity are passed as parameters. This event is triggered only by versions of your app that include the Google Analytics for Firebase SDK. **Android:** To see in-app purchase data for Android apps, link Analytics to Google Play. Note that Analytics doesn't automatically measure paid-app purchase revenue. Also, your reported revenue in Google Analytics may differ from the values you see in the Google Play Developer Console. Analytics ignores events that are flagged as invalid or tests. Learn more about testing Google Play billing. **iOS:** Note that Analytics doesn't automatically measure paid-app purchase revenue and refunds. Analytics ignores events that are flagged as invalid or sandbox.	
notification_dismiss (app)	when a user dismisses a notification sent by Firebase Cloud Messaging (FCM) Android only	message_name, message_time, message_device_time, message_id, topic, label, message_channel

notification_foreground (app)	when a notification sent by FCM is received while the app is in the foreground	message_name, message_time, message_device_time, message_id, topic, label, message_channel, message_type
notification_open (app)	when a user opens a notification sent by FCM	message_name, message_time, message_device_time, message_id, topic, label, message_channel
notification_receive (app)	when a notification sent by FCM is received by a device when the app is in the background Android only	message_name, message_time, message_device_time, message_id, topic, label, message_channel, message_type
os_update (app)	when the device operating system is updated to a new version. The previous operating system version id is passed as a parameter	previous_os_version
page_view (web)	each time the page loads or the browser history state is changed by the active site Collected by default via enhanced measurement.	page_location (page URL), page_referrer (previous page URL), engagement_time_msec
screen_view (app)	when a screen transition occurs and any of the following criteria are met: • No screen was previously set • The new screen name differs from the previous screen name	firebase_screen, firebase_screen_class, firebase_screen_id, firebase_previous_screen, firebase_previous_class, firebase_previous_id, engagement_time_msec

	- The new screen-class name differs from the previous screen-class name - The new screen id differs from the previous screen id	
scroll (web)	the first time a user reaches the bottom of each page (i.e., when a 90% vertical depth becomes visible) Collected by default via enhanced measurement.	engagement_time_msec
session_start (app, web)	when a user engages the app or website A session ID and session number are generated automatically with each session and associated with each event in the session. Learn more	
user_engagement (app, web)	when the app is in the foreground or webpage is in focus for at least one second. Learn more	engagement_time_msec
video_complete (web)	when the video ends For embedded YouTube videos that have JS API support enabled. Collected by default via enhanced measurement.	video_current_time, video_duration, video_percent, video_provider, video_title, video_url, visible (boolean)
video_progress (web)	when the video progresses past 10%, 25%, 50%, and 75% duration time For embedded YouTube videos that have JS API support enabled. Collected by default via enhanced measurement.	video_current_time, video_duration, video_percent, video_provider, video_title, video_url, visible (boolean)

video_start (web)	when the video starts playing For embedded YouTube videos that have JS API support enabled. Collected by default via enhanced measurement.	video_current_time, video_duration, video_percent, video_provider, video_title, video_url, visible (boolean)
view_search_results (web)	each time a user performs a site search, indicated by the presence of a URL query parameter Collected by default via enhanced measurement.	search_term, optionally 'q_<additional key="">' (where <additional key=""> matches an additional query parameter you specify to be collected under advanced settings) Note: This event only sends the unique_search_term parameter when it has a value of 1 (i.e. when the string is unique to that session).

Source: < https://support.google.com/analytics/answer/9234069?hl=en >

6.4 Enhanced measurement events

[GA4] Enhanced measurement events

Discover how to enable and disable enhanced measurement events and learn more about which parameters are collected for each event.

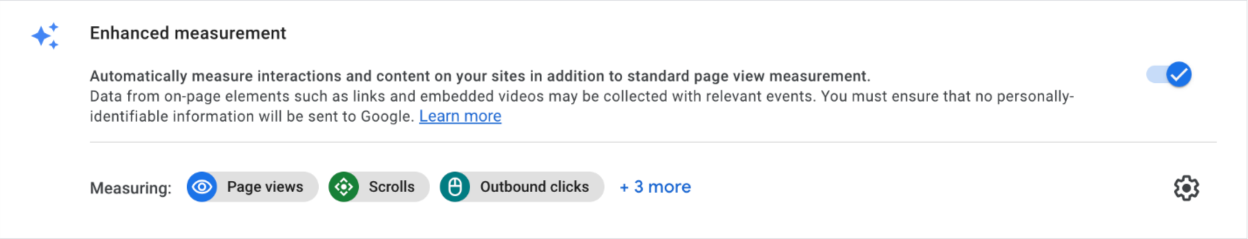
Enhanced measurement lets you measure interactions with your content by enabling options (events) in the Google Analytics interface. No code changes are required. When you enable these options for a web data stream, your Google Analytics tag starts sending events right away.

Before turning on the enhanced measurement feature, be sure you understand each option and what enhanced data will be collected. You can also turn off specific measurement options in settings.

You're required to ensure that no personally identifiable information is collected.

Enable or disable enhanced measurement events

1. In **Admin**, under ***Data collection and modification***, click **Data streams**.
2. Click the name of your data stream.
3. Under ***Enhanced measurement***, slide the switch **On** to enable all options. Click ⚙ to edit individual options as needed.



Enhanced measurement

Automatically measure interactions and content on your sites in addition to standard page view measurement.
Data from on-page elements such as links and embedded videos may be collected with relevant events. You must ensure that no personally-identifiable information will be sent to Google. Learn more

Measuring: Page views Scrolls Outbound clicks + 3 more

If you use the Google tag on your website, you also need to make sure that each event is enabled for automatic event detection for your Google tag. By default, all event types are enabled. Learn more about your Google tag settings

Events measurement and parameters

The following table explains when events are triggered, and which parameters are collected for each event. You can find enhanced data about each triggered event in the Events report within the Engagement topic. Click the event name in the report for more information on the event.

Measurement option / event	Triggered...	Parameters
Page views page_view	each time the page loads or the browser history state is changed by the active site This event is collected automatically. You cannot turn off collection. An advanced setting on this option controls whether the event is sent based on browser-history events. This measurement option listens for pushState, popState, and replaceState. The event populates the ***Views*** metric. The parameters populate the following dimensions: • ***Page location*** (from page_location) • ***Page referrer*** (from page_referrer)	page_location (page URL), page_referrer (previous page URL)
Scrolls scroll	the first time a user reaches the bottom of each page (i.e., when a 90% vertical depth becomes visible) The event populates the ***Percent scrolled*** dimension.	No parameters are collected

Measurement option / event	Triggered...	Parameters
Outbound clicks click	each time a user clicks a link that leads away from the current domain By default, outbound click events will occur for all links leading away from the current domain. Links to domains configured for cross-domain measurement will not trigger outbound click events. The parameters populate the following dimensions: • ***Link classes*** (from link_classes) • ***Link domain*** (from link_domain) • ***Link ID*** (from link_id) • ***Link URL*** (from link_url) • ***Outbound*** (from outbound)	link_classes, link_domain, link_id, link_url, outbound (boolean)
Site search view_search_results	each time a user is presented with a search results page, as indicated by the presence of a URL query parameter By default, the event is triggered based on the presence of one of the following 5 query parameters in the URL: • q • s • search • query • keyword You can optionally configure this event to look for other URL query parameters. The search_term parameter populates the ***Search term*** dimension.	search_term, optionally ‘q_<additional key="">’ (where <additional key=""> matches an additional query parameter you specify to be collected under advanced settings). Note: This event only sends the unique_search_term parameter when it has a value of 1 (i.e. when the string is unique to that session).
Video engagement video_start video_progress video_complete	For YouTube embedded videos that have JS API support enabled, the following events are triggered: • video_start: when the video starts playing • video_progress: when the video progresses past 10%, 25%, 50%, and 75% duration time • video_complete: when the video ends The parameters populate the following dimensions: • ***Video provider*** (from video_provider) • ***Video title*** (from video_title) • ***Video URL*** (from video_url) • ***Visible*** (from visible)	video_current_time, video_duration, video_percent, video_provider, video_title, video_url, visible (boolean)

Measurement option / event	Triggered...	Parameters
File downloads file_download	when a user clicks a link leading to a file (with a common file extension) of the following types: - document - text - executable - presentation - compressed file - video - audio File extensions that match the following regex will trigger the event: pdf|xlsx?|docx?|txt|rtf|csv|exe|key|pp(s|t|tx)|7z|pkg|rar|gz|zip|avi|mov|mp4|mpe?g|wmv|midi?|mp3|wav|wma The parameters populate the following dimensions: - ***File extension*** (from file_extension) - ***File name*** (from file_name) - ***Link classes*** (from link_classes) - ***Link ID*** (from link_id) - ***Link text*** (from link_text) - ***Link URL*** (from link_url)	file_extension, file_name, link_classes, link_id, link_text, link_url
Form interactions form_start form_submit	'form_start': the first time a user interacts with a form in a session 'form_submit': when the user submits a form You can use these two events to see how many users started to fill out a form and compare the information to users who submitted the form. Note: You can only use the parameters in your reports if you create custom dimensions for them.	form_start - form_id: HTML id attribute of the <form> DOM element - form_name: HTML name attribute of the <form> DOM element - form_destination: URL to which the form is being submitted form_submit - form_id: HTML id attribute of the <form> DOM element - form_name: HTML name attribute of the <form> DOM element - form_destination: URL to which the form is being submitted - form_submit_text: text of the submit button, if present

Source: < https://support.google.com/analytics/answer/9216061?hl=en >

6.5 Data collection

Data collection

Google Analytics for Firebase collects certain information in its default implementation.

Google Analytics for Firebase data collection

The type of information collected through the Google Analytics for Firebase default implementation includes:

- Number of users and sessions
- Session duration
- Operating systems
- Device models
- Geography
- First launches
- App opens
- App updates
- In-app purchases

See a full list of the default events and user properties collected by Google Analytics for Firebase.

Identifying devices

The Google Analytics for Firebase SDK library uses an app-instance identifier to identify a unique installation of the App.

When using the SDK, an app-instance identifier gets generated at the app level.

By default, the Firebase SDK collects identifiers for mobile devices (for example, Android Advertising ID and Advertising Identifier for iOS) and utilizes technologies similar to cookies.

On iOS, the SDK collects the Advertising Identifier if it is available. For IDFA to be available, a developer has to link in the AdSupport.framework library.

If the Advertising Identifier is unavailable, the SDK collects the Vendor Identifier. If the Advertising Identifier becomes available after the Vendor Identifier was reported, the SDK stops collecting the Vendor Identifier.

By default, on Android the SDK collects the Advertising ID.

Source: < https://support.google.com/analytics/answer/11593727?hl=en >

6.6 Custom Events

[GA4] Custom events

Collect additional information that Google Analytics does not collect automatically

A user sometimes performs an action you want to analyze in your reports that isn't one of the automatically collected or recommended events. In these cases, it may make sense to implement a custom event.

A custom event is an event that you define so you can collect information about an interaction that's important to your business.

For example, while Google Analytics records when a user views a page, you may want to know when a user makes a donation, interacts with a new feature, lands on a confirmation page, or renames a file. In these cases, you likely want to implement a custom event.

Watch a video about custom events

Before you create a custom event

Before you create a custom event, make sure the event you want to create isn't already collected through an automatically collected event or recommended as a recommended event. It's always better to use an existing event because these events automatically populate dimensions and metrics that are used in your reports.

How to implement a custom event

You can implement a custom event in a few different ways, depending on how you set up your website or app measurement. These includes:

- gtag.js (for websites)
- Tag Manager (for websites)
- Google Analytics for Firebase (for mobile apps)

The specific implementation details differ depending on how you set up your website or app measurement. Refer to the developer documentation (liked above) for more details on how to implement a custom event.

The anatomy of a custom event

A custom event consists of the following parts:

- The custom event name
- The custom event parameters associated with the custom event

Event name

The name of a custom event is whatever name you choose for the event. The name should describe what you intend to measure with the event. For example, if you're measuring donations, the name might be "donate."

Before you name a custom event, make sure the name adheres to the event naming rules (e.g., the name is case sensitive, cannot be a reserved name, and starts with a letter) and event naming limits (i.e., the name must be fewer than 40 characters in length) to ensure that Google Analytics collects and processes the event.

When someone triggers the custom event on your website or app, the event name is used to count how many times a user performed the action. For example, if someone makes a donation, then the count of the donate event would increase by 1. Any additional information about the donation should be captured as custom event parameters.

Event parameters

The event parameters of a custom event provide more information about the action that took place. For example, you could measure whether the action was successful, when the event occurred, or choices users made during the interaction.

A parameter consists of key-value pairs. Each pair includes these parts:

- The parameter name, which describes the information you're collecting
- The parameter value, which is the value associated with the parameter in that interaction

The parameter name is the same name used across sessions, while the parameter value should get updated depending on what the user does in the session. For example, if one user makes a donation of $1.00, then the 'value' parameter would be '1.00', while if another user makes a donation of $2.00, then the 'value' parameter would be '2.00'. This allows you to consistently measure the same information across sessions.

Custom dimensions and metrics

To access the different values assigned to an event parameter in your reports, you should create a custom dimension or metric. A custom dimension or metric lets you see the information you collected from an event parameter. For example, if you set up a 'value' event parameter, you could create a custom metric called 'Value' that allows you to see each value assigned to the event parameter. Learn more about custom dimensions and metrics.

Cardinality and system limits

While you set up a custom event parameter, be mindful of the number of possible values that can be assigned to the parameter.

Each parameter can have a number of values assigned to it. For example, a 'mobile' custom event parameter might have two potential values – 'true' or 'false'. Other parameters might have any number of values assigned, such as 'page_location', which could have a different value for every URL on your website.

When you create a custom dimension to see categorical information like page_location, the number of possible values matters. Dimensions with more than 500 possible values are considered high-cardinality dimensions. Reports and explorations that contain high-cardinality dimensions may be affected by Google Analytics system limits, which can lead to values getting rolled up in an (other) row or data sampling may get triggered.

Additionally, review the number of custom dimensions and metrics you can create before creating custom event parameters. If you send more event parameters than these limits, you will need to choose which of these event parameters you want to use as custom dimensions and metrics.

Event collection limits

There's no limit on the number of custom events you can create for a web data stream (i.e., a website). However, you can collect up to 500 distinct events per mobile app user per day. For example, you might see 700 distinct events if you have two different users on different app instances who each trigger different events.

The automatically collected and enhanced measurement events don't count towards these limits.

Review the other event collection limits before creating your own events.

Conversions

Any custom event can be marked as a conversion. For example, if donations are important to the success of your business, you might mark the custom event 'donate' as a conversion.

Sometimes, the action you want to measure is partially captured through an existing event, but you want to narrow the scope of the event to measure an action that's important to your business. For example, Analytics already measures pageviews automatically, but you may want a separate custom event for when someone views a confirmation page.

Instead of creating a custom event in these cases, you could create a copy of the event or modify the event directly, from within Google Analytics. Then, you could mark the event as a conversion. This allows you to create events quickly without needing to update your website code.

However, make sure to communicate these changes with the person who makes changes to your website code if that isn't you. Otherwise, you may overcount the interaction if they too create a custom event for that interaction.

See the events in your reports

After you implement a custom event and Google Analytics collects the event, you can use the Events report in the Reports section to see how many times the event was collected and other data about the event in the specified date range.

You can also select an event name to open a more detailed report about that event, including details about the parameters, demographics about the users who triggered the event, and how many users triggered each event (and the associated parameters) in realtime.

Source: < https://support.google.com/analytics/answer/12229021?hl=en >

7. Approvals

- This document accurately documents the Google Analytics 4 functionality as described to the MPBSD Privacy Office.
- The MPBSD Privacy Office is responsible for the recommendations contained in this document.
- This PIA must be maintained by MPBSD and updated as required.

Peter HOPE-TINDALL (MPBSD)

Digitally signed by Peter HOPE-TINDALL (MPBSD)
Date: 2024.06.18 10:25:25 -04'00'

2024-06-18

Peter HOPE-TINDALL
Head of Privacy - MPBSD

Date

- This document accurately documents the Google Analytics 4 functionality.
- The MPBSD Digital Innovation and Product Division has reviewed and understood the recommendations contained in this document.
- This PIA must be maintained by MPBSD and updated as required.

Spencer Daniels

Digitally signed by Spencer Daniels
Date: 2024.06.18 11:47:38 -04'00'

Spencer DANIELS
Assistant Director – Product-Ontario.ca, Digital Innovation and Product Division, MPBSD

Date

[This page intentionally blank]

*** END OF DOCUMENT ***