

ID Synchronization

Monetate identifies a single person across different devices and domains using ID synchronization. This process occurs within a Customer View by matching multiple Monetate IDs that share the same Person ID. The Person ID is observed by the platform via an ID Collector that tracks the Person ID within cookies, JavaScript variables, query parameters, and HTML elements created and configured within a Customer View. ID synchronization provides a complete understanding of the customer across desktop websites, mobile websites, and websites of sister brands no matter what channel, device, or browser the customer uses.

In this screenshot, you can see three different Monetate IDs matched to one customer.

These Monetate IDs are joined because they share the same Person ID. In this example, all three Monetate IDs have the same Person ID, `1234`, and are associated with Jane Doe.

The ID used in a dataset upload must match the case of the value collected in an on-site ID.

How ID Synchronization Works

When you define a Person ID, Monetate must know where to collect the value of the Person ID for each customer. ID Collectors added to a Customer View setup tell the platform where to collect the Person ID value, such as in a cookie, a JavaScript variable, an HTML element, or a URL query parameter. Each ID Collector must collect the same value for each customer for ID synchronization to work.

The ID synchronization process itself links the collected Person ID for each customer with the Monetate ID. Monetate identifies customers across devices and domains by synchronizing all the Monetate IDs—one per device/browser combination—with the same Person ID value seen across devices and domains. For more information about ID Collectors, see [Client-Side ID Collectors](#).

You may have a different ID Collector for each of your domains, and you may have more than one collector for a single domain. The most important consideration is the Person ID value that's collected. All the Person ID values must map to the same customers across all the ID Collectors. For example, if Monetate detects a customer with a Person ID value of `1234` using a cookie-based ID Collector on a login page, that same customer should have a Person ID of `1234` using a query parameter-based ID Collector.

Consider Jane Doe, the customer whose ID was linked across devices in the screenshot. For this example, she accesses your main site and logs in on her iPhone. The ID Collector `onSiteID` should collect the Person ID `1234`. If Monetate doesn't already recognize Person ID `1234`, then it assumes that this unrecognized ID is for a new customer. This screenshot shows how the resulting entry may look.

Later, Jane clicks through to your site from a Monetate email link on her Mac, which triggers the query parameter ID collector `emailClickThrough` to collect a Person ID. Because the visitor is still Jane and because Person IDs are based on people, the platform collects the Person ID `1234`. It then recognizes the ID and joins the two Person IDs that share the value `1234`. It continues to join Person IDs as your customers access your site from different

locations and devices.

Personally Identifiable Information

You could upload to Monetate certain pieces of personally identifiable information (PII) in a [Customer Attributes dataset](#), such as customers' email addresses.

Monetate does not filter or restrict the customer data processed in a Customer Attributes dataset that you upload. You are responsible for ensuring that the data you upload complies with all applicable data privacy laws, regulations, and policies.

See the [Consumer Data Privacy](#) documentation category for more information about the data privacy compliance tools that Monetate offers its clients to help them comply with various data privacy laws and regulations.

Using an Existing ID Collector

You can use the ID Collector you have from your datasets for ID synchronization as long as it collects Person IDs based on people. Follow these steps to use an existing ID Collector.

1. Click **COMPONENTS** in the top navigation bar, and then select **Customer View**.
2. Click the name of the Customer View.
3. Click **COLLECTION**.
4. If the Customer View has multiple collection methods or multiple websites, find the tile for the entry to which you want to add the ID Collector. Click **ADD OR CREATE ID COLLECTOR**.
5. Find the ID Collector you want to add in the Add an ID Collector modal, and then click **ADD**.

If you haven't used Customer Attributes datasets or if you have but the ID Collector represents a session ID, a browser ID, or something other than a person, create a new ID Collector that's based on a person for ID synchronization. See [Upload a Customer Attributes Dataset](#) for more information.

The Utility of the Customer View

The Customer View can offer a complete picture of a single customer, which then serves as the backbone for improving cross-screen behavioral segmentation, customer analytics, and more. Additionally, the Customer View setup is foundational for onboarding any omnichannel customer data into the Monetate platform so that you can use all customer data within the platform for cross-device testing and experience segmentation, Automated Personalization context (for Engine customers only), analytics, and more. Consider these two examples of how you

can benefit from using the Customer View.

Use Case One

On his home computer, Mike views a specific brand that makes him eligible for one of your company's Web experiences, but then he clears his browser's cookies. If your company is not using Customer ID Manager, then Mike would no longer be eligible for the experience and all his behavioral data would be lost. He would be treated as a new customer the next time he visited your site.

However, if your company is using Customer ID Manager, then Monetate can recognize when a new visitor is actually associated with a known person according to his Person ID and then reference Mike's past behavioral data and his browsing experience remains consistent.

Use Case Two

On the way to work, Jane is shopping on her tablet and is placed in experiment group A for one of your company's Web experiences. She then switches to her desktop when she gets to work. Cross-device testing ensures that once Jane is identified on one device and placed in experiment group A, she is placed in the same experiment group when she arrives on your site on a second device. As a result, Jane is placed into experiment group A on her work computer as well and sees the same, consistent experience on both devices.