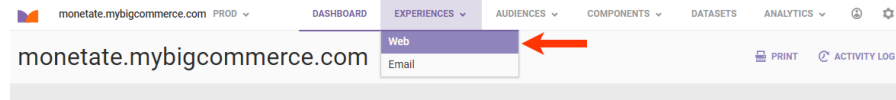


Create an Omnichannel Experience

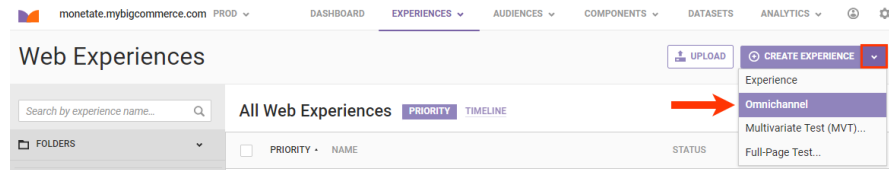
Regardless of the channel that you want to serve an experience on (mobile app, server-side integration for Web, etc.), the process for creating an Engine API experience is the same with the exception of the actions available for the experience.

Follow these steps to configure an Engine API experience.

1. Click **EXPERIENCES** in the top navigation bar, and then select **Web**.



2. Click the arrow to the right of **CREATE EXPERIENCE**, and then select **Omnichannel**.

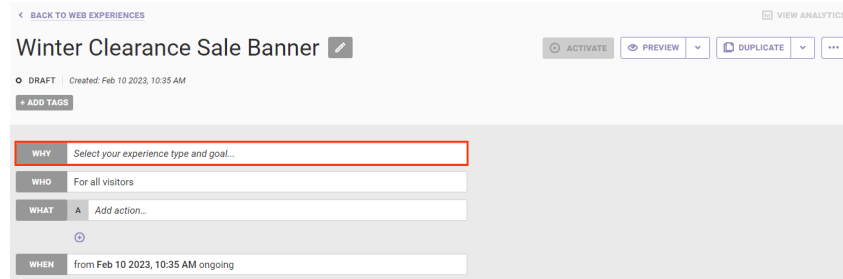


3. Name the experience.



4. Configure the WHY settings.

- a. Click **WHY**.



- b. Select an experience type and then click **NEXT**.

WHY

WHO

WHAT

WHEN

1 Why are you running this experience?

RECOMMENDED

To optimize my goal with Machine Learning. **MACHINE LEARNING** **100% EXP**

Automated Personalization **AUTO PERSONALIZATION**

Use Automated Personalization to serve the best-fit variant to each individual customer.

Highest ROI better for long-running experiences with high traffic volume

Dynamic Testing **DYNAMIC TESTING**

Use Dynamic Testing to monitor experience results in real time and automatically allocate more traffic to the winner.

Better for short-running experiences with limited traffic

Note: This experience will start with an even distribution between variants until the engine begins to learn.

NEXT

c. Select a goal metric and then click **NEXT**. Refer to [Event/Metric Mapping in Target, Event, & Condition Mapping](#) for a table of default metrics and their respective associated Engine API events.

1 Why are you running this experience?

To serve everyone the same experience. **100% EXP**

2 Goal

Which metric do you want to optimize with this experience?

★ Select a goal metric below

Search for metric by name...

GOAL	METRIC	
★	Conversion rate	<input type="text" value="DEFAULT"/>
★	New customer acquisition rate	<input type="text" value="DEFAULT"/>
★	Add to cart rate	<input type="text" value="DEFAULT"/>
★	Cart abandonment rate	<input type="text" value="DEFAULT"/>
★	Bounce rate	<input type="text" value="DEFAULT"/>
★	Revenue per session	<input type="text" value="DEFAULT"/>
★	Average page views	<input type="text" value="DEFAULT"/>

BACK **NEXT**

d. Configure as necessary the secondary metrics, and then click **NEXT**.

2 Goal

★ Conversion rate


3 Confirm Secondary Metrics

These metrics will be tracked as you optimize your goal.

⊕ ADD METRIC

Cart abandonment rate	DEFAULT
Bounce rate	DEFAULT
Revenue per session	DEFAULT
Average page views	DEFAULT
Average time on site	DEFAULT
Average order value	DEFAULT
Example Click Event	
mouseoverevent	

BACK NEXT



e. Confirm that the WHY summary statement is accurate, and then click **SAVE**.

4 Confirm

I would like to **serve the same experience to each customer to measure Conversion rate** and **9** additional metrics.

SAVE CANCEL


5. Configure the WHO settings. Refer to [Target Type Mapping in Target, Event, & Condition Mapping](#) for a table of target types and their respective associated Engine API events.
- a. Click **WHO** and then click **ADD TARGET**.

WHY to increase lift for all traffic

WHO For all visitors

WHAT A Add action...

WHEN from Feb 10 2023, 10:35 AM ongoing

WHO  ADD TARGET

- b. Click a category on the Target Type panel, select a target, and then configure it as necessary. See [WHO Targets in Omnichannel Experiences](#) in this documentation for more information.

WHO TARGET TYPE RETURN TO TARGET LIST

Target Types

Landing	Location	Behavior	Technographics
US Demographics	Weather	Named Segments	Custom Targets
Datasets	Audiences	Audience Discovery	AdLink

6. Configure the WHAT settings.

- a. Click **WHAT** and then click **ADD ACTION**.

BACK TO WEB EXPERIENCES VIEW ANALYTICS

Winter Clearance Sale Banner ACTIVATE PREVIEW DUPLICATE

DRAFT Created Feb 10 2023, 10:35 AM

+ ADD TAGS

WHY to increase lift for all traffic

WHO For all visitors

WHAT **A** Add action...

WHEN from Feb 10 2023, 10:35 AM ongoing

WHAT ADD ACTION

- b. Click a category on the Action Type panel, select an action template, and then configure it as necessary. See [Available Action Types](#) for more information about the action templates typically available for Engine API clients. See [Using Action Conditions](#) in this documentation for guidance on configuring action conditions.

WHAT ACTION TYPE RETURN TO WHAT

Action Type

HTML Recommendation Social Proof Product Finder

Badges Other

7. Configure the WHEN settings.

- a. Click **WHEN** and then click **START AND STOP TIME**.

WHY to increase lift for all traffic

WHO For all visitors

WHAT **A** show Omni HTML

WHEN from Feb 10 2023, 10:35 AM ongoing

WHEN BACK

START AND STOP TIME from February 10, 2023 at 10:35 AM ongoing

- b. Configure the start and, optionally, stop date and time as well as dayparting and time zone settings as necessary, and then click **SAVE**. See [Configure Experience Timing](#) for more information.

WHEN START AND STOP TIME RETURN TO WHEN

Start and Stop Time

Select a start and stop time for this experience, based on the retailer's time zone (America/Dominica).

Start Experience End Experience

Start date Start time End date End time

2/10/23 10:35 AM

Ongoing Specific end date

Day Parting

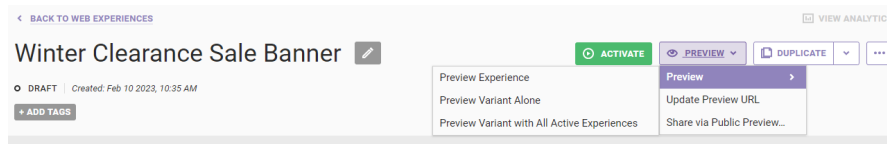
To turn this experience on for only some parts of the day, first choose the time zone context (your timezone or the visitor's timezone), and then select the hours when you would like the experience to be on.

Time Zone Context

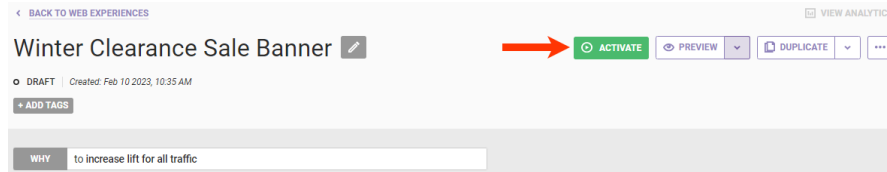
NONE

SAVE CANCEL

8. Click **PREVIEW** to test the experience. See [Previewing an Experience](#) for more information.



9. Click **ACTIVATE** when you're ready to deploy the experience.



See [Calling the Engine API](#) for more information about Engine API request calls for Omnichannel experiences.

WHO Targets in Omnichannel Experiences

Engine API experiences support most WHO targets that you can use in an experience built for a JavaScript tag implementation, with the exception of a few that rely on data specific to Monetate tag-based integrations.

In Monetate tag implementations, many of the targets are evaluated based on information passed via the request header from the browser (information such as IP address, used to determine the location and related information, and user agent, from which device type, and operating system are derived).

In a server-side implementation, these are not available to Monetate and must be provided in the body of the Engine API request, as shown in this example.

```
{
  "channel": "a-1d121c07/p/monetate-api.myshopify.com",
  "monetateId": "5.275061469.1543777025730",
  "events": [
    {
      "eventType": "monetate.decision:DecisionRequest",
      "requestId": "11111",
      "includeReporting": true
    },
    {
      "eventType": "monetate.context:UserAgent",
      "userAgent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_13_6) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/68.0.3422.100 Safari/537.36"
    },
    {
      "eventType": "monetate.context:ipAddress",
      "ipAddress": "193.51.100.0"
    }
  ]
}
```

Using Action Conditions

Similar to JavaScript tag-based implementations, you can apply action conditions to the actions you configure in Engine API experiences.

For example, if you want an action to appear on a particular page type, click **ADD CONDITION**, expand the options of the **Page type** category, and then select **Page type =**.

WHAT ACTION TYPE OTHER OMNI JSON < RETURN TO SELECT ACTION

Omni JSON

Returns JSON
This action contains no built-in targeting.

Required Inputs

JSON


```
{ }
```

JSON to be returned

Optional Inputs

Apply this action when the following conditions are met:

+ ADD CONDITION v

Page type >	Page type = 
URL >	Page type ≠
Time >	
Cart value >	
Landing page >	
Page breadcrumb >	
Page category >	
Product detail page >	
Product list page >	

Type the page type into the field to complete the action condition. The page type is defined by the `setPageType` data you pass to the platform.

Here is the request body with `pageType` specified in the `PageView` event:

```
{
  "channel": "a-1d121c07/p/monetate-api.myshopify.com",
  "monetateId": "5.275061469.1543777025730",
  "events": [
    {
      "eventType": "monetate:decision:DecisionRequest",
      "requestId": "111111",
      "includeReporting": true
    },
    {
      "eventType": "monetate:context:PageView",
      "pageType": "home",
      "url": "test/foo"
    }
  ]
}
```

Refer to [Action Condition Mapping in Target, Event, & Condition Mapping](#) for a table of action condition options and their respective associated Engine API events.

Previewing an Experience

The experience preview has two functionalities:

- **Experience Preview:** This preview mimics experience evaluation in full so anyone who views it is evaluated as if they are a normal site visitor. To qualify for the experience, you must satisfy any targets and action conditions associated with the experience. The Engine API also decides what split you should see and then serves it.
- **Variant Preview:** This preview allows you to see the content returned by a specific variant. Anyone who views it sees the actions associated with the chosen variant when they satisfy the action conditions. Variant preview disregards any targets set in the experience.

The experience preview loads in a new browser window or tab. The URL is typically for the account's home page

and includes a `preview` query parameter. The value for the parameter is an encoded preview token.

Here is an example of a preview URL:

```
http://www.monetate.com/?  
preview=6.3.ejyrVkpMTs4vz5ujz0xRsji1tjTTUUpOB7HNzczNjcyBPBAnrzQnR0cptajAycrQ1NDAwtjc2NRQRym9CCRQCwDQbBOe.xKAgglhOuGG4Vivm5OP7GBFLW!
```

In a JavaScript tag-based implementation, the Monetate tag recognizes the `preview` parameter, adds it to its own call to the Monetate platform, and then applies the relevant actions to preview.

However, for Engine API integrations you must provide the preview token to the Engine API call.



The preview token is valid for 1 hour.

Here is a code sample for passing the `preview` parameter.

```
{  
  "preview": "preview token",  
  "events": [  
    {  
      "pageType": "home",  
      "eventType": "monetate.context:PageView"  
    }  
  ]  
}
```