



This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

Journey JavaScript SDK

Use Events methods with web actions

Contents

- [1 Event methods for web actions](#)
- [2 Media types, lifecycle states, and code examples using Events methods](#)
- [3 Event types for web actions](#)
- [4 Capture more data with Genesys widgets](#)

Use Events methods to subscribe to events that occur during the lifecycle of Genesys Predictive Engagement web actions such as web chats and content offers. This raw data can be streamed to third-party analytics platforms or to tag management platform data layers for use in analytics and reporting platforms.

Important

This article only applies to customers using web chat. If you are a Genesys Cloud CX customer, we encourage you to use the new web messaging feature to replace web chat.

Event methods for web actions

To capture information about events that occur during the lifecycle of a web action, use the Events methods shown in the following table.

| Method | Description |
|--------|--|
| on | Subscribe to start receiving events for a given type of web action in a given state. |
| once | Receive events for only the first occurrence of a given type of web action in a given state. |
| off | Unsubscribe to stop receiving events for a given type of web action in a given state. |

Media types, lifecycle states, and code examples using Events methods

| Media type | Lifecycle states | Code examples |
|--------------|--------------------------|--|
| contentoffer | Content offers lifecycle | Examples: Events methods with content offers |
| webchat | Web chat lifecycle | Examples: Events methods with web chats |

Event types for web actions

The following table lists the events that you can use with Events methods for web actions. Event information returned includes the action state, customer ID, session ID, and action map ID.

| Event | Data type | Description |
|------------------|-----------|--|
| actionId | UUIDv4 | Unique Id for a specific Predictive Engagement action. |
| actionState | String | Current state of the action. For example, offered. |
| actionMediaType | String | The engagement type. For example, webchat. |
| actionMapId | UUIDv4 | Id of the action map that qualified/triggered this action. |
| actionMapVersion | Integer | Version of the action map. |
| customerId | UUIDv4 | Stable identifier of the customer. For example, a cookie Id. |
| customerIdType | String | The specific type of customer identifier (always "cookie"). |
| sessionId | UUIDv4 | Identifier of the customer's current web session. |
| errorCode | Integer | Status code for any exceptions caught during presentation of the action. |
| errorMessage | String | Error message for any exceptions caught during presentation of the action. |

Capture more data with Genesys widgets

You can use the Web Action Events API with Genesys Widgets commands to enrich events with more data that may be useful.

For example, the Genesys Cloud CX `conversationId` may be useful in an analytics context. For more information on the Widgets API, see [API Commands](#).

```
ac('on', 'webchat:all', (evt) => {
  _genesys.widgets.bus.command('WebChatService.getSessionData').then((data) => {
    if (data.conversationId) {
      evt.conversationId = data.conversationId;
    }

    someAnalyticsProvider.send(evt);
  })
});
```

Important

The data that `WebChatService.getSessionData` returns differs based on your Genesys platform.