

Poll unit JavaScript events

07/14/2025 9:14 am EDT

(Updated 05/26/2022)

The poll unit triggers custom events during its lifecycle. Your code may listen to these events and perform custom actions when the events occur. Example actions include logging to an analytics platform, modifying the page's content, starting a video, etc.

Custom events

The poll unit uses the JavaScript `CustomEvent` feature to dispatch its events. For most events, some additional information is included in the `detail` property of the event object. The poll unit events bubble up through the ancestors of the poll unit container element. Thus, it is possible and often convenient to listen for events at the topmost level of the document.

To avoid naming conflicts with custom events from your code and other libraries or plugins, the poll unit event types begin with the `civicscience:` prefix.

Using vanilla JavaScript

Here is an example of listening for a poll unit event using vanilla JavaScript.

```
document.addEventListener("civicscience:questionAnswered", function (event) {  
  // handle the event  
  console.log("the user answered question %s", event.detail.id);  
});
```

Question events

The question events can be used to monitor a user answering poll questions in the poll unit.

```
civicscience:beforeFetchingQuestions
```

This event is triggered before the poll unit **requests a batch of questions from the server** to ask the user.

There are no detail properties associated with this event.

```
civicscience:beforeAskingQuestions
```

This event is triggered when the batch of questions to ask has been received from the server, just before the first question will be asked.

This event has one detail property:

- `ids` - an array of question identifiers in the order that they will be asked

```
civicscience:questionAsked
```

This event is triggered each time a question appears in the poll unit to be asked to the user.

This event has three detail properties:

- **id** - the identifier for the question being asked
- **index** - the 0-based index of the asked question in the poll
- **length** - the total number of questions in the poll

```
civicscience:questionAnswered
```

This event is triggered when the user submits an answer to a question.

This event has three detail properties:

- **id** - the identifier for the question that was just answered
- **index** - the 0-based index of the answered question in the poll
- **length** - the total number of questions in the poll

```
civicscience:afterAskingQuestions
```

This event is triggered immediately after the user has answered the last question in the poll and before the results are shown.

This event has one detail property:

- **ids** - an array of question identifiers just answered in the poll, in the order they were answered

Results events

The results events are helpful in monitoring how a user navigates among a series of results pages or "slides" in the poll unit at the end of a poll.

```
civicscience:beforeShowingResults
```

This event is triggered just before the first results slide appears in the poll unit.

This event has one detail property:

- **length** - the total number of results slides

```
civicscience:resultDeparted
```

This event is triggered when the user clicks the back or next button to leave one results slide for another. It is *not* triggered when the user navigates beyond the last results slide to begin a new poll. Instead, this event is triggered:

```
civicscience:afterShowingResults
```

The `civicscience:resultDeparted` event has three detail properties:

- **from** - the 0-based index of the original results slide
- **to** - the 0-based index of the destination results slide
- **length** - the total number of results slides

civicscience:resultArrived

This event is triggered when the user arrives at one results slide from another by clicking a back or next button. It is *not* triggered when the first slide initially appears, in which case this event is triggered instead:

civicscience:beforeShowingResults

The `civicscience:resultArrived` event has three detail properties:

- `from` – the 0-based index of the original results slide
- `to` – the 0-based index of the destination results slide
- `length` – the total number of results slides

Why are there two results navigation event types?

At a glance, supporting both the `civicscience:resultDeparted` and `civicscience:resultArrived` events may seem redundant. For many poll unit configurations, they will indeed be triggered nearly simultaneously.

However, for some instances of poll units with animation effects, there is a delay between the user clicking a navigation button on the original results slide and the appearance of the destination results slide. Providing two separate event types allows you to choose whether your event listener should react to results navigation before or after this delay.

civicscience:afterShowingResults

This event is triggered when the user navigates beyond the last results slide to begin a new poll in the same poll unit visit. Poll unit instances that link to a [carousel site](#) on the last slide do not have this capability, so those instances will never trigger this event.

This event has one detail property:

- `length` – the total number of results slides

Frame events

The frame events can be used to monitor changes in the poll unit iframe.

civicscience:frameResize

This event is triggered when the content of the poll unit has caused the instance to increase/decrease in size to fit its content.

This event has one detail property:

- `height` – the height to which the frame is attempting to change

Event summary

The table below summarizes the events triggered by the poll unit.

event	details	notes
-------	---------	-------

civicscience:beforeFetchingQuestions	none	before the initial server request for getting questions to ask
civicscience:beforeAskingQuestions	ids	after the server responds with questions to ask
civicscience:questionAsked	id , index , length	when a question appears in the poll unit to be asked
civicscience:questionAnswered	id , index , length	when the user answers a question
civicscience:afterAskingQuestions	ids	immediately after the last question has been answered
civicscience:beforeShowingResults	length	immediately before the first result slide is shown
civicscience:resultDeparted	from , to , length	when the user leaves one results slide for another
civicscience:resultArrived	from , to , length	when the user arrives at one results slide from another
civicscience:afterShowingResults	length	when the user navigates past the last results slide to begin a new poll in the poll unit
civicscience:frameResize	height	when the content of the poll unit has caused the instance to increase/decrease in size to fit its content