

Qubit.

- **Set up Qubit on Shopify**

April 2021



In this article...

We'll cover the steps involved in setting up Qubit technology on your Shopify store.

Overview

From a technical perspective, the setup can be divided into two steps:

1. Add the `smartserve.js` script to the `<head>` of every page
2. Emit events based on visitor browsing behavior

`qubit-events.liquid` and `qubit-events-transaction.liquid` handle both of these tasks for you using Shopify templates.

We have also bundled the UV API, which sets up an API as a conduit for events. Any script can emit events using the API, even before `smartserve.js` loads, and those events can be read and subscribed to by any script using the API.

INFO: `qubit-events.liquid` and `qubit-events-transaction-liquid` provide the sample logic that can be copied across to your Shopify templates.

Setup tasks

Perform these tasks in the following order:

Step 1 - Update the `smartserve.js` script

► Update the following line at the top of `qubit-events.liquid` and `qubit-events-transaction.liquid` with your unique Id:

```
<script src="//static.goqubit.com/smartserve-XXXX.js" async defer></script>
```

Where:

- xxxx is a unique Id provided to you by Qubit

Step 2 - Copy the qubit-events.liquid file to the Snippets directory

- ▶ Copy *qubit-events.liquid* to the **Snippets** directory, located in *Themes > Template Editor > Snippets*

Step 3 - Add {% include 'qubit-events' %}

- ▶ At the bottom of the <head> in *theme.liquid* in the **Layouts/** directory, add the following line:

```
{% include 'qubit-events' %}
```

Step 4 - Insert ecProduct event

If your products have different size and color variants, then an additional product event needs to be emitted when the variant is changed. Due to the way Shopify themes work, it is not possible to template this as each theme can override the switcher.

- ▶ Insert the following event code into the function which handles the variant change:

```
uv.emit('ecProduct', {
  eventType: 'detail',
  product: {
    productId: "{{ product.id | json }}",
    sku: {{ product.selected_or_first_available_variant.sku | default:
product.selected_or_first_available_variant.id | json }},
    name: {{ product.title | json }},
    manufacturer: {{ product.vendor | json }},
    stock: {{ product.selected_or_first_available_variant.inventory_quantity
}},
    price: {
      currency: {{ shop.currency | json }},
      value: {{ product.price | money_without_currency | remove: ',' }}
    },
    url: window.location.protocol + "://" + window.location.hostname + "{{
product.url }}",
    description: {{ product.description | strip_html | strip_newlines |
truncate: 2048 | json }},
    categories: ["{{ product.type}}"],
```

```
images: [ window.location.protocol + "{{ product | img_url: '400x' }}" ]
  }
})
```

Liquid variables are included and assume product is the object of the new variant.

WARNING: If you have been given a namespace prefix for your property, prefix it to `ecProduct`.

Step 5 - Copy and paste the contents of `qubit-events.liquid`

The Shopify template system does not allow code to be added to the confirmation page, but it does provide a separate mechanism to achieve what is required.

► Copy and paste the contents of `qubit-events-transaction.liquid` into the text box found at *Settings > Checkout > Additional content & scripts*

Additional information

- There are a lot of inline JavaScript comments within the templates. If you do not want these to be publicly accessible, we recommend removing them
- `qubit-events.liquid` and `qubit-events-transaction.liquid` handle the vast majority of the implementation. However, if you're looking to do something more custom, additional coding may be required
- Any questions, please reach out to support@qubit.com