Tracking SPA applications with Matomo

Tomas Persson 2021-11-06

Digitalist

https://digitalist.se/matomo

matomo camp

Who I am

https://github.com/tomper00

https://www.linkedin.com/in/tomper00/

Who we are

Digitalist

a digital product agency

that loves Open Source

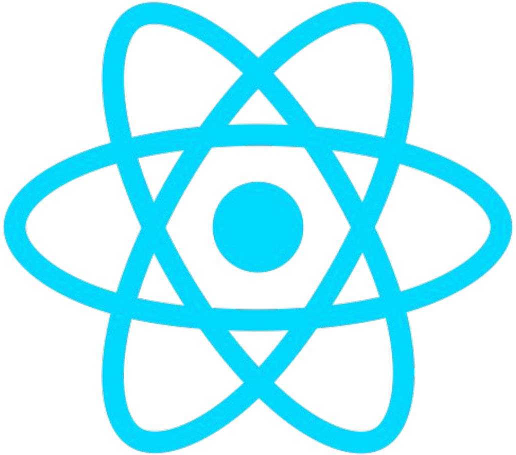
Because we know that good ideas multiply when shared

What is an SPA application

A single-page application is a web application or website that interacts with the user by dynamically rewriting the current web page with new data from the web server, instead of the default method of a web browser loading entire new pages.

Example of SPA frameworks

Angular



React

Vue

SPA tracking - requires us to “fake pageviews”

Pageview tracking is based on “real pageviews” (reloads)

Browser Event: Pageview

In an SPA there is only one Pageview event (when you load the app).

Instead listen to the event
HistoryChange

URL Example

https://example.com/myapp#startpage

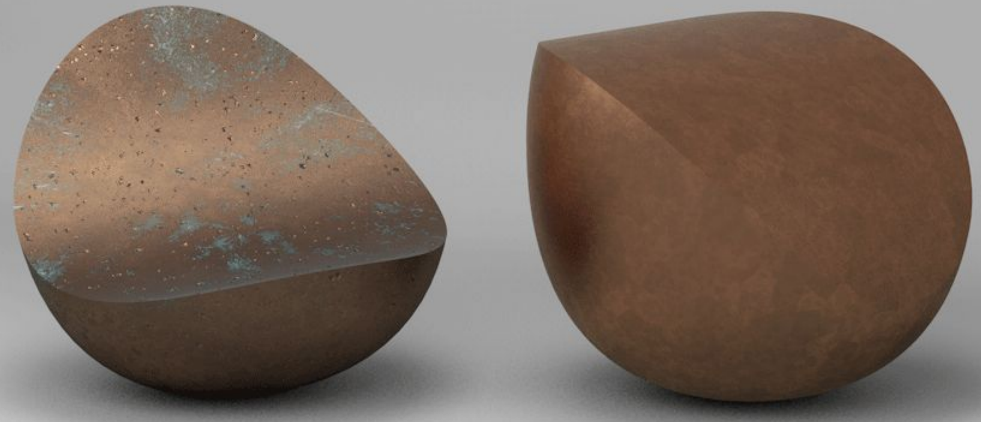
https://example.com/myapp#about-us

https://example.com/myapp#search

Setting up basic tracking

1. **Listen to the event historyChange in the Tag Manager**
2. **Add Matomo script to your apps html**
3. **Create a Pageview tag in Matomo and fire on the historyChange event**

DEMO TIME



https://yog9.github.io/SnapShot/#/SnapScout/mountain

Common challenges

- New title is not set on historyChange
- Url is not rewritten
- Duplicate pageviews (due to duplicate tracking)
- What about event tracking?

New title is not set on historyChange

Developers sometimes forget to set document.title

new **page titles** on url rewrites

leading to:

1. **All pageviews sent report under the same Title**
2. **This is also bad for SEO**

Solution

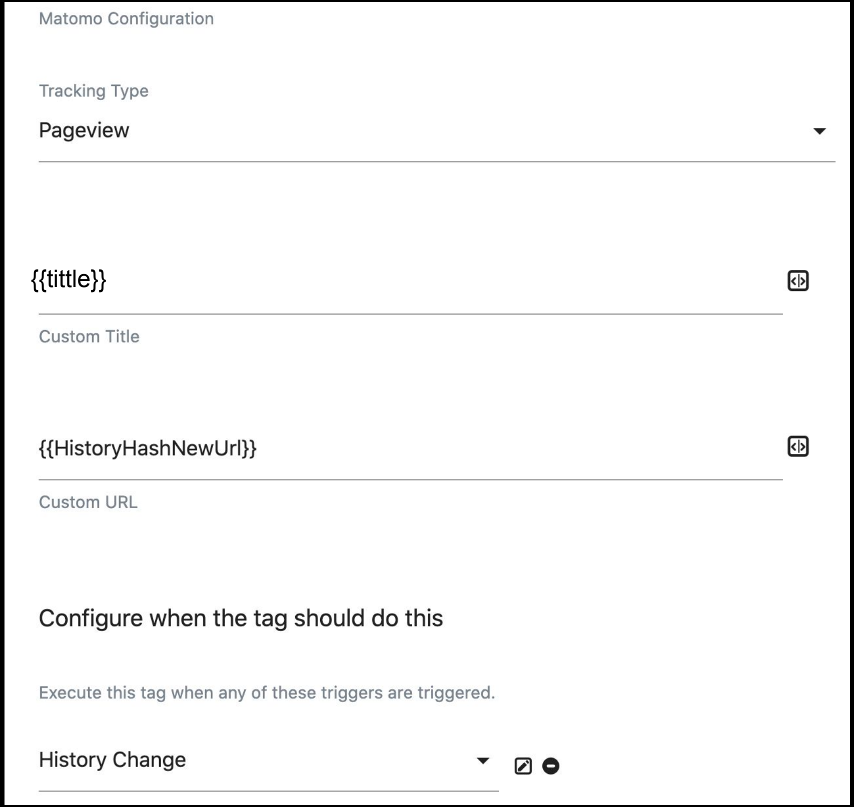
Best: Change the SPA so that titles are correctly set.

Workaround: Write a custom Javascript variable that sets the title based on something else.

(add the url as a title is an easy fallback)

History Change events in the Tag Managern

1. Create a PageView Tag
2. Create a trigger of type History Change
3. Create a variable that reads the page title
4. Use the variabel **{{HistoryHashNewUrl}}** as URL



Matomo Configuration

Tracking Type

Pageview

{{tittle}}

Custom Title

{{HistoryHashNewUrl}}

Custom URL

Configure when the tag should do this

Execute this tag when any of these triggers are triggered.

History Change

Url is not rewritten

Developers sometimes forget to do url routing

leading to:

1. **Hard to track**
2. **Hard to link to context in application**
3. **Bad for SEO**

Solution

Best: Change the SPA so that url routing works

Workaround: Write a custom Javascript that listens to other events pr clicks in the app and fire Fake page views based on these.

Warning: this is really not recommended

Duplicate pageviews (due to duplicate tracking)

The first page in the app gets reported twice

leading to:

1. **To many pageviews in Matomo**

Solution

Best: set up your triggers so that std pageviews are not fired on SPA pages.

Examples:

1. create a variable that detects the SPA in HTML
2. manually add urls where you do not want std pageviews

Control data from the SPA

Sometimes we simply have better control of the data and the events in the Application, then we can use the Tag Manager as a middle man that just routes data between the SPA before it is sent to Matomo.

The benefit of doing so instead of skipping the Tag Manager is that we can still use the Tag Manager to set up filters & exceptions or other things over time.

Send a customEvent and catch data in the TagManagern (example for pageviews)

I VPV

```
var _mtm = _mtm || [];
_mtm.push({
      'event': 'customPageview',
      customURL: '/my-url',
      customTitle: 'My-title'
  });
```

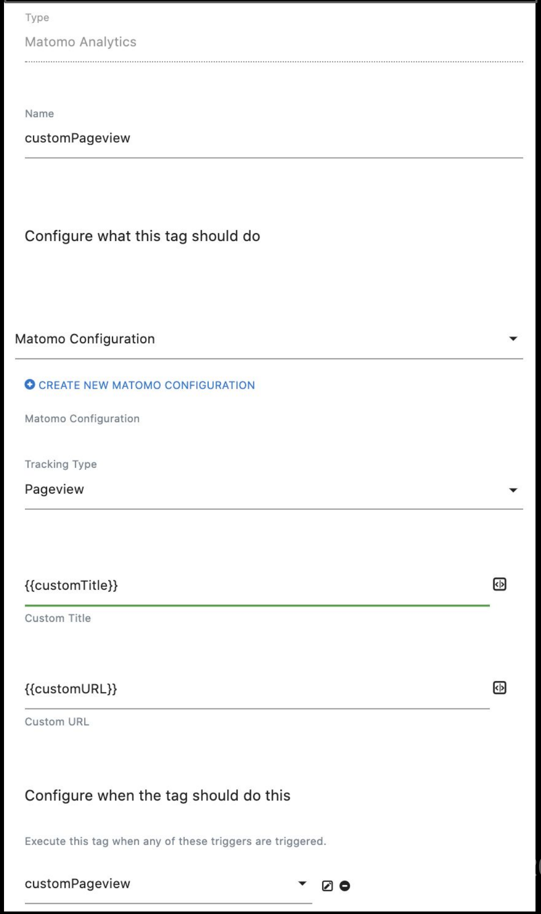
Tagmanager variables

customURL (type Datalayer)
customTitle (type Datalayer)

Tagmanager trigger

customPageview (type customPageview)

TagManager Tag



Type
Matomo Analytics

Name
customPageview

Configure what this tag should do

Matomo Configuration

CREATE NEW MATOMO CONFIGURATION

Matomo Configuration

Tracking Type
Pageview

{{customTitle}}
Custom Title

{{customURL}}
Custom URL

Configure when the tag should do this

Execute this tag when any of these triggers are triggered.

customPageview

Example with Events

```
_mtm.push({
  'event': 'customEvent',
  customEventCategory: <<Event category>>,
  customEventAction: <<Event action>>,
  customEventName: <<Event name>>,
 customEventValue: <<Event value>>
});
```

Tagmanager variables

customEventCategory (type Datalayer)
customEventAction (type Datalayer)
customEventName (type Datalayer)
customEventValue (type Datalayer)

Tagmanager trigger

customPageview (type customEvent)

Read more here
https://developer.matomo.org/guides/spa-tracking