MEASURE CAMP

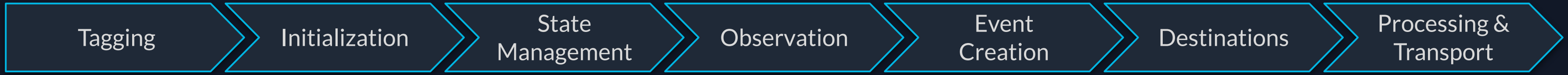
HELSINKI

Challenge it: one implementation to rule them all?

elbwalker/walker.js

Data collection with walker.js

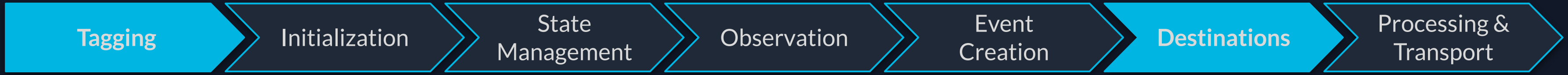
Once upon a dataLayer.push



Tagging	Initialization	State Management	Observation	Event Creation	Destinations	Processing & Transport
Measurement Plan	Loading State	Consent	Loading	Properties	Consent Permissions	Setup & Init
Configuration	Event Listener	User IDs	Clicking	Context	Custom Settings	Mapping
HTML Attributes	Mutation Observer	Event Queue	Scrolling	Globals	Event Settings	Queueing
	Updates	Timings	Timing	Misc		Sending
		Misc	Custom			

Data collection with walker.js

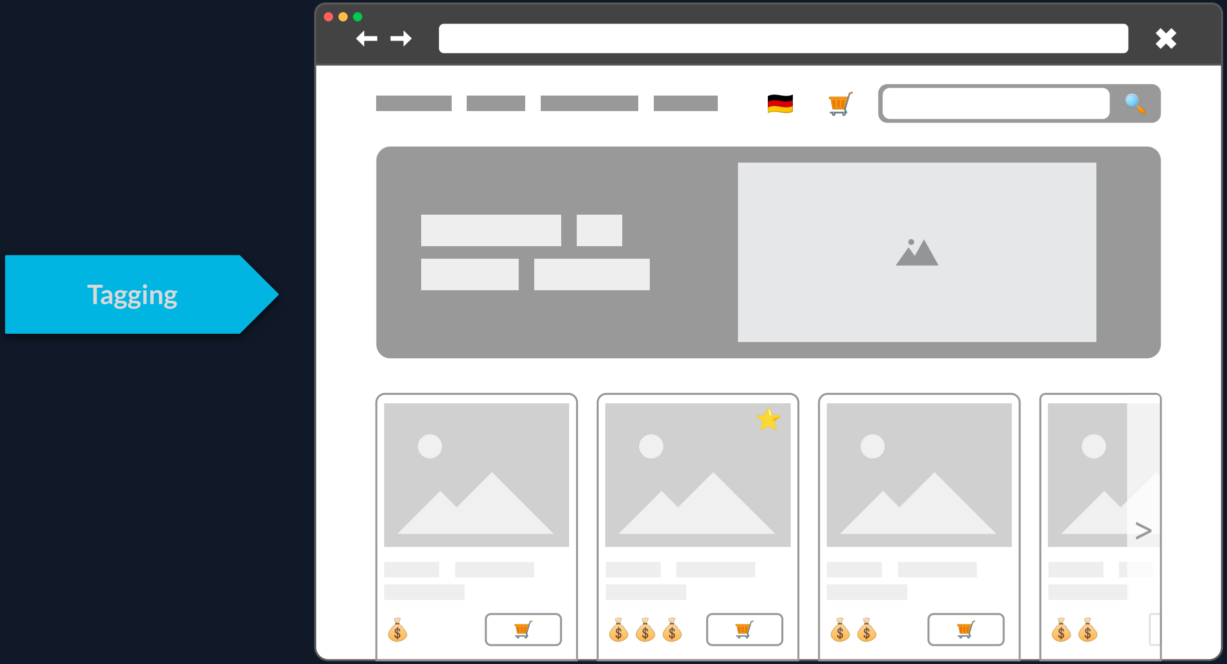
Once upon a dataLayer.push



Tagging	Initialization	State Management	Observation	Event Creation	Destinations	Processing & Transport
Measurement Plan	Loading State	Consent	Loading	Properties	Consent Permissions	Setup & Init
Configuration	Event Listener	User IDs	Clicking	Context	Custom Settings	Mapping
HTML Attributes	Mutation Observer	Event Queue	Scrolling	Globals	Event Settings	Queueing
	Updates	Timings	Timing	Misc		Sending
		Misc	Custom			

Data collection with walker.js

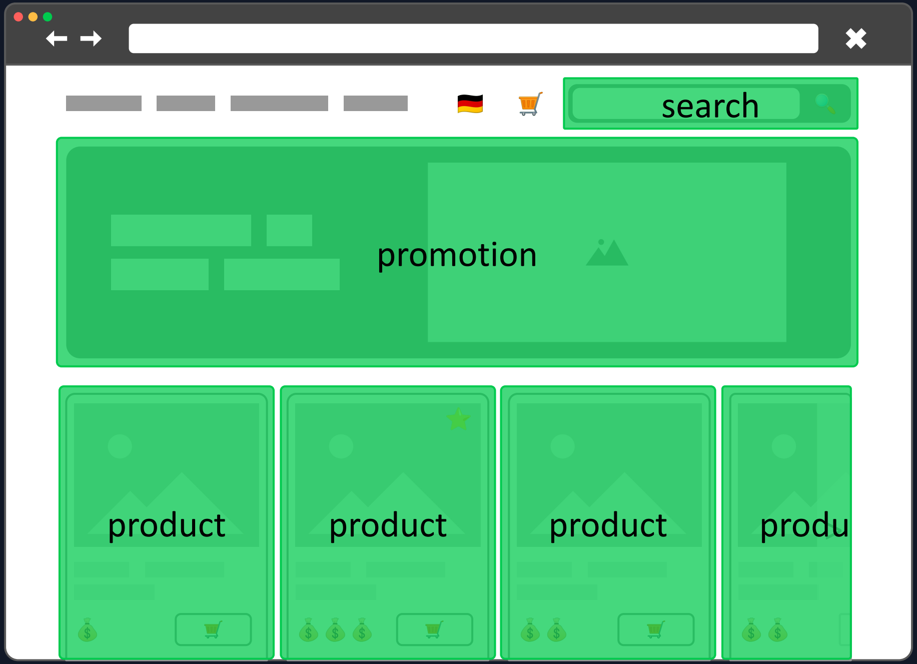
How to tag a webpage



Data collection with walker.js

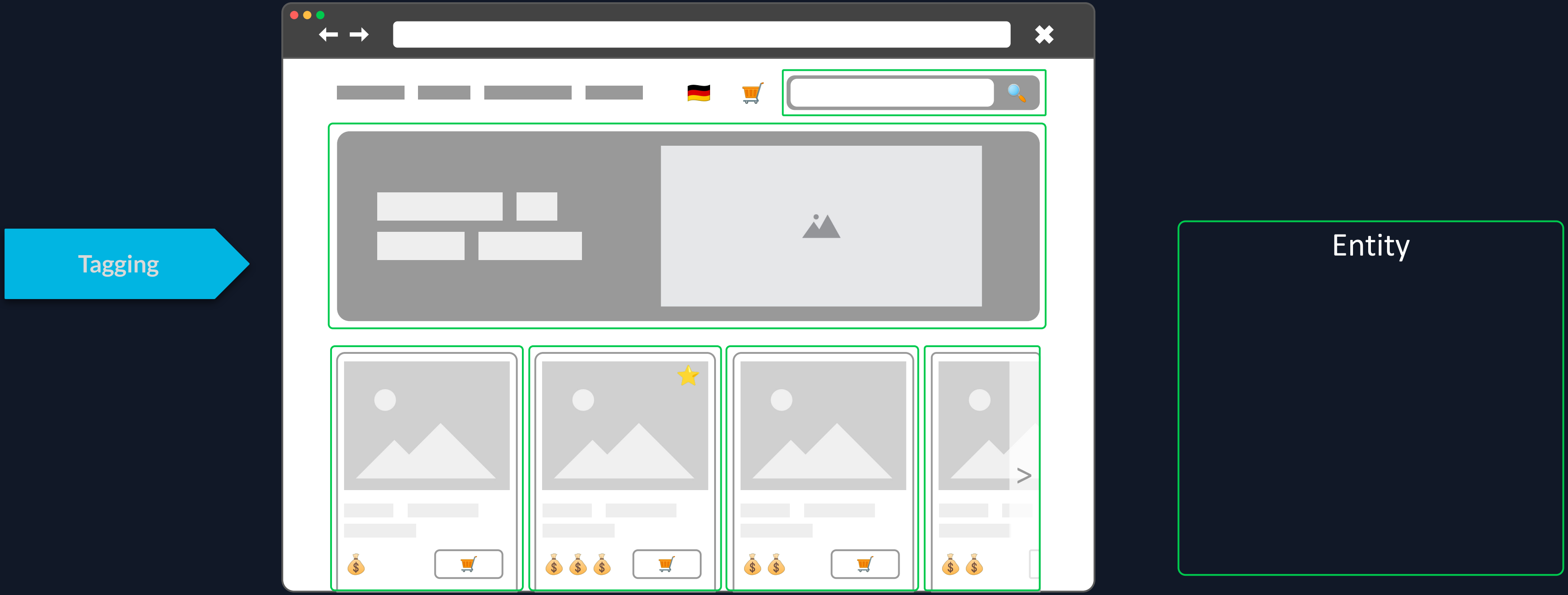
How to tag a webpage

Tagging



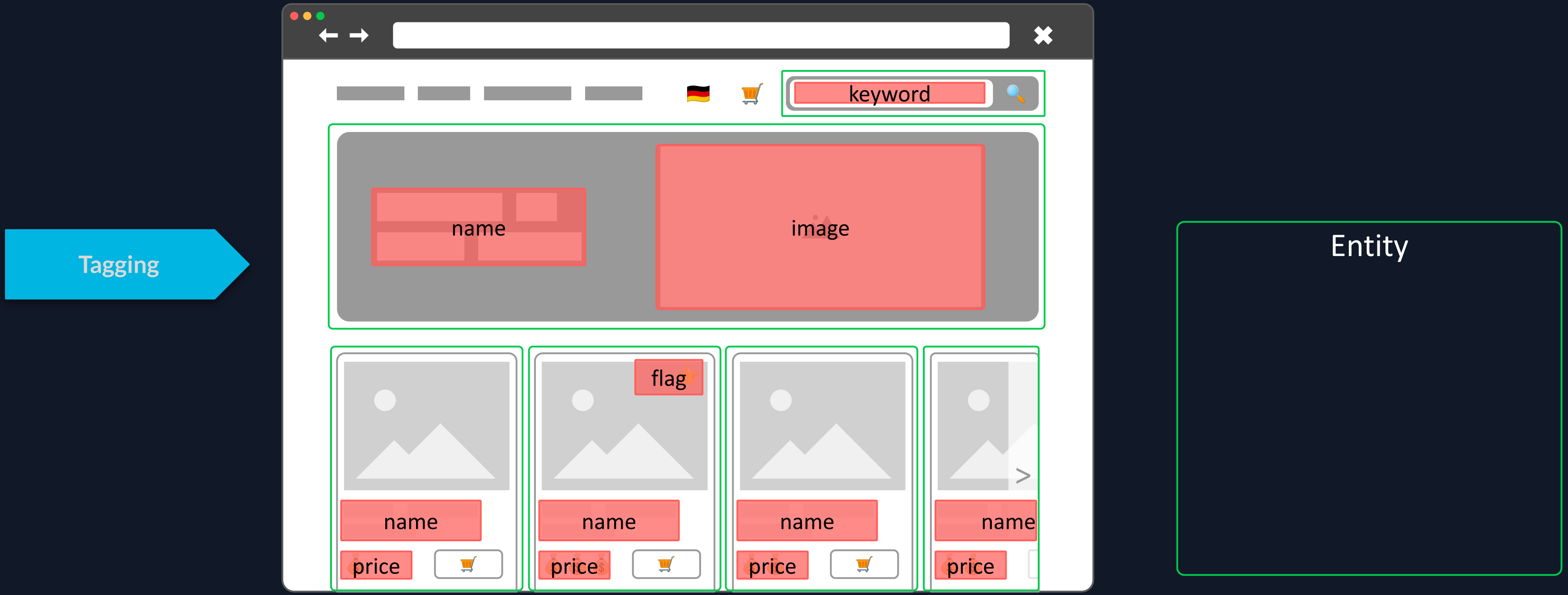
Data collection with walker.js

How to tag a webpage



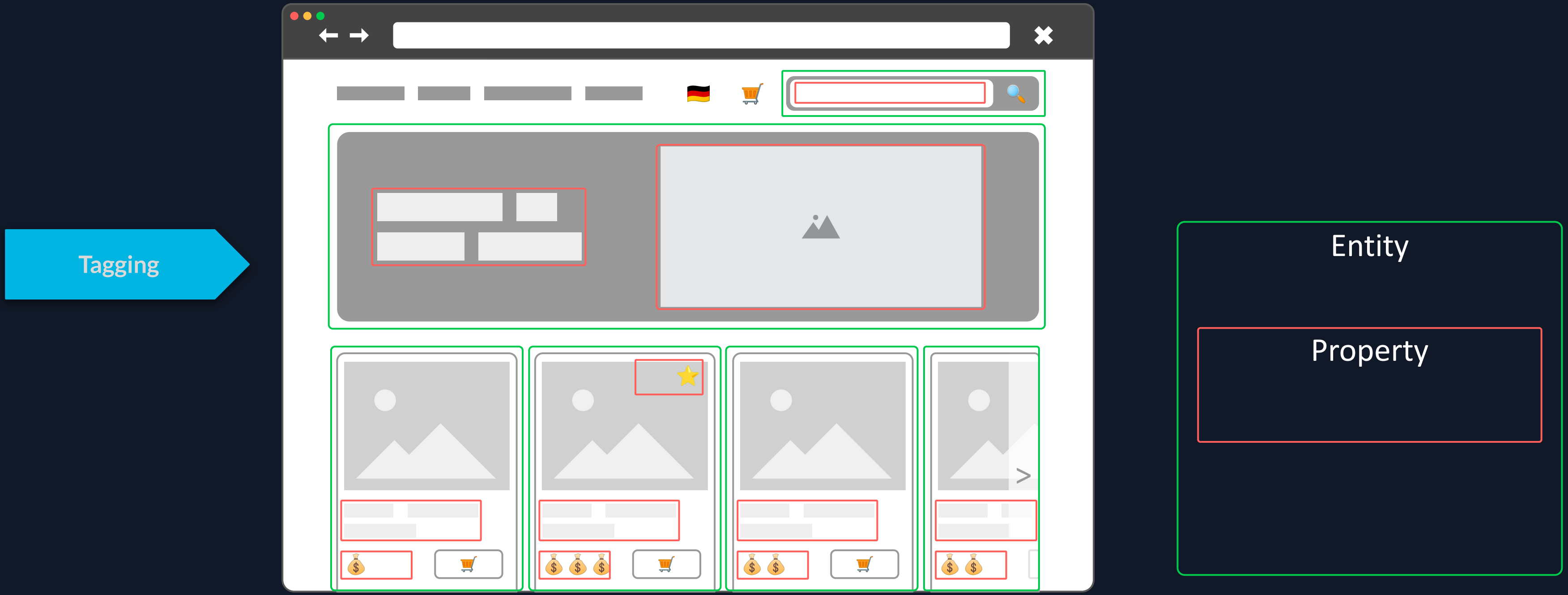
Data collection with walker.js

How to tag a webpage



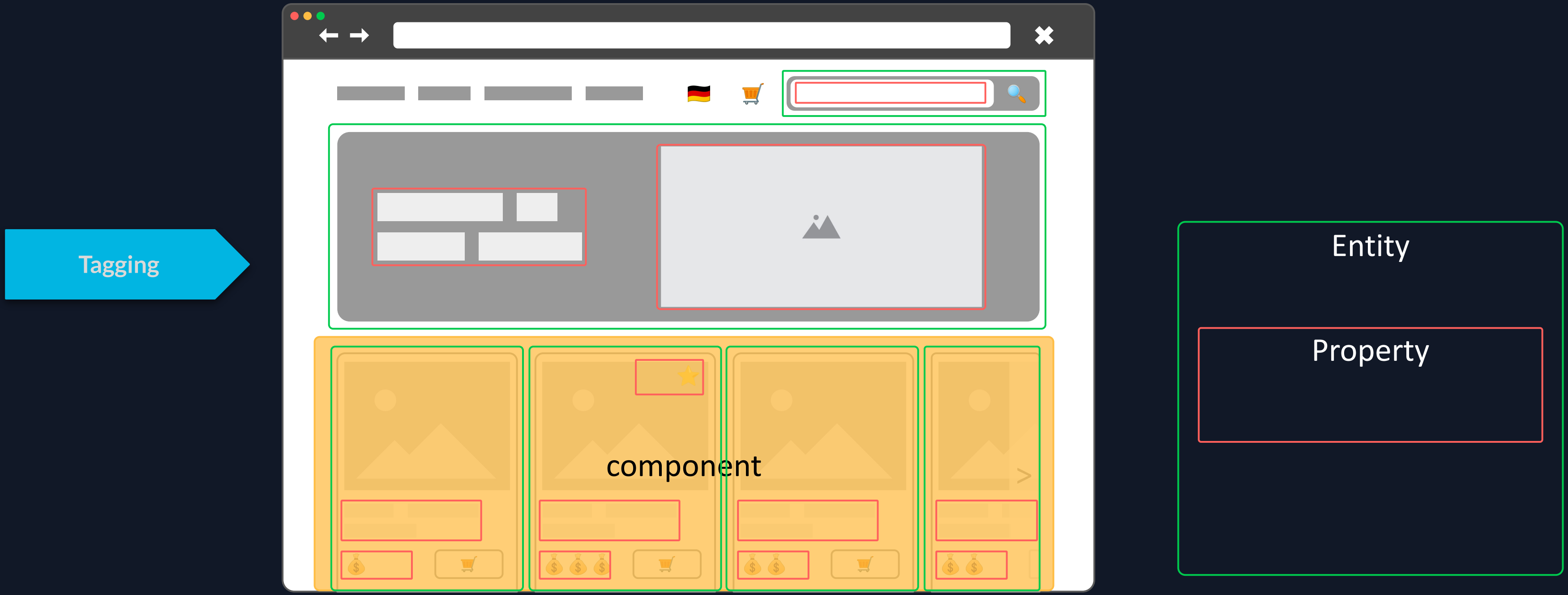
Data collection with walker.js

How to tag a webpage



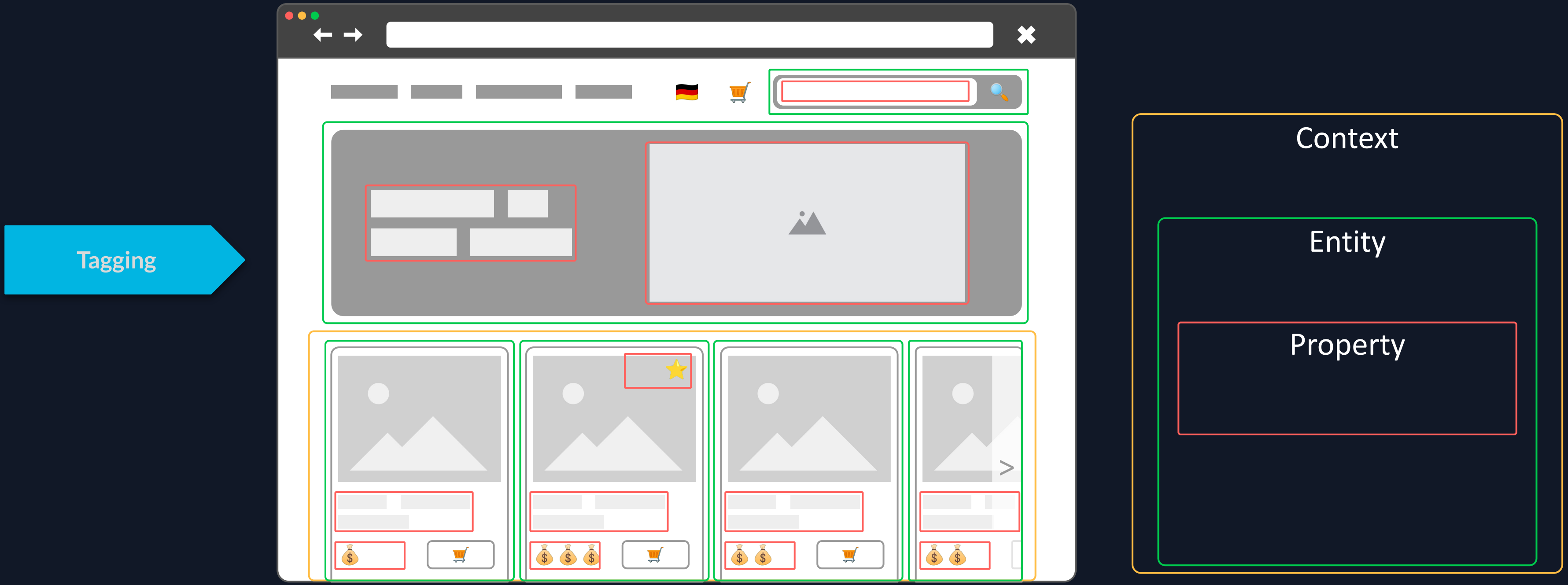
Data collection with walker.js

How to tag a webpage



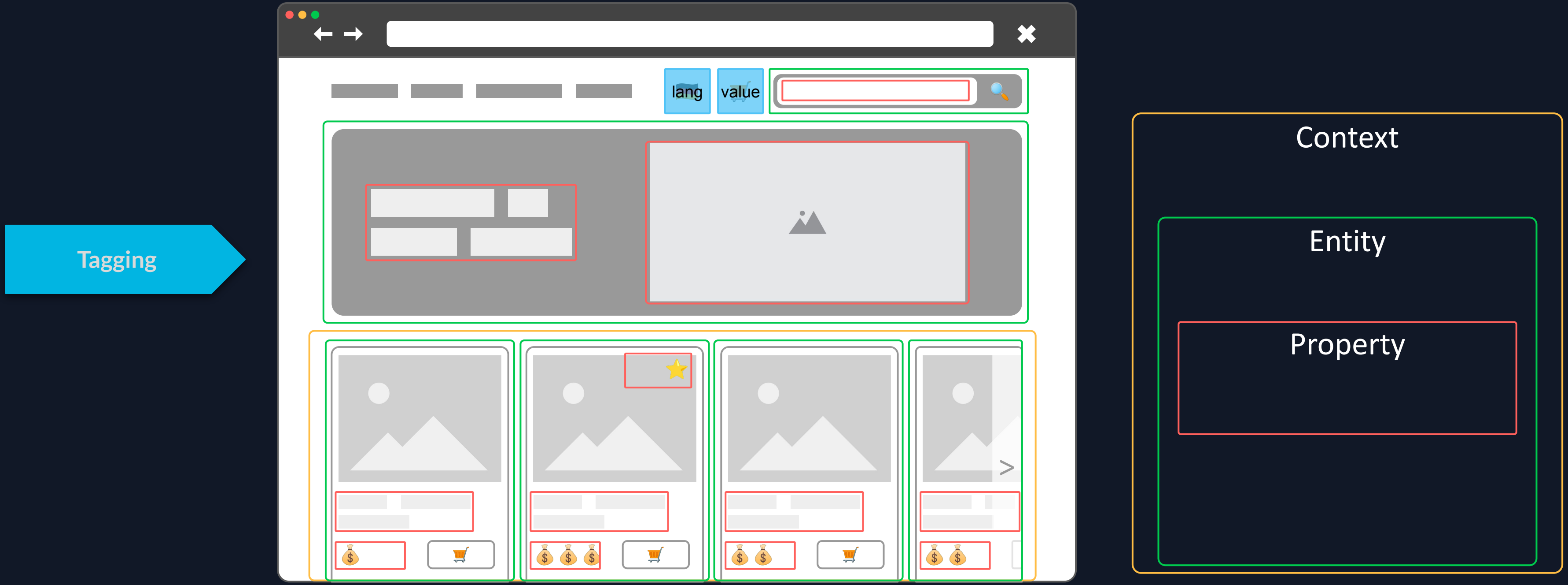
Data collection with walker.js

How to tag a webpage



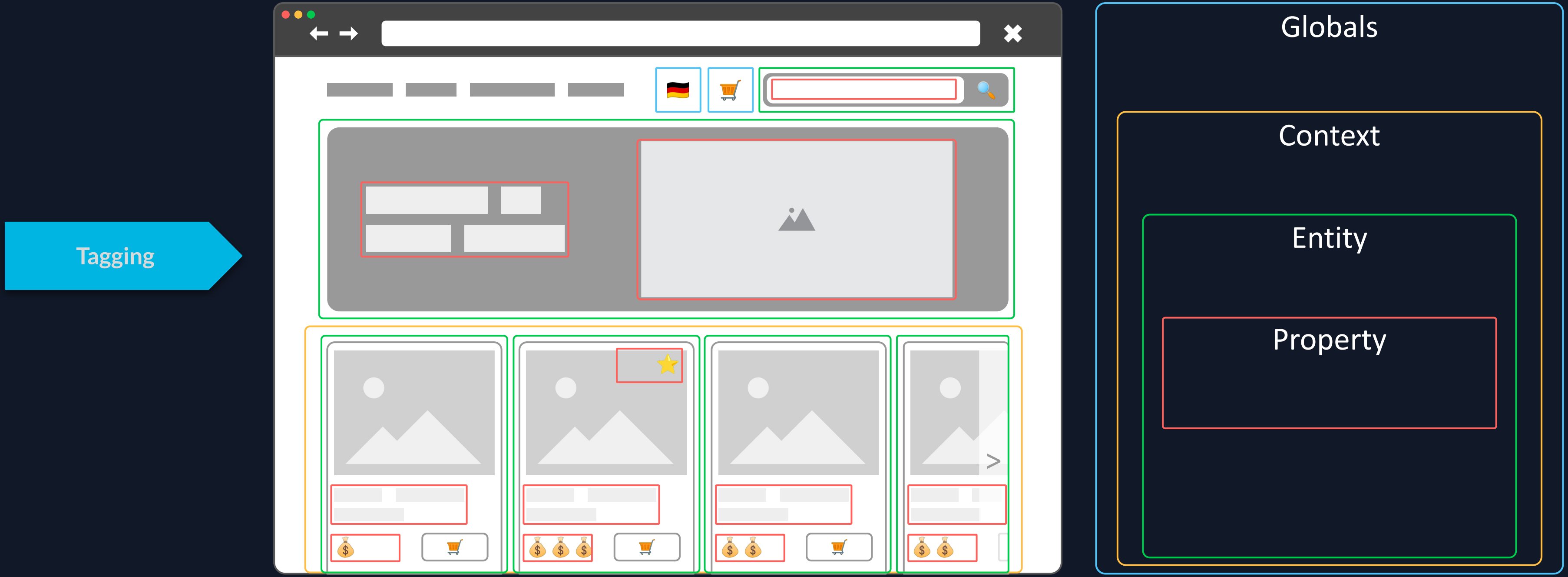
Data collection with walker.js

How to tag a webpage



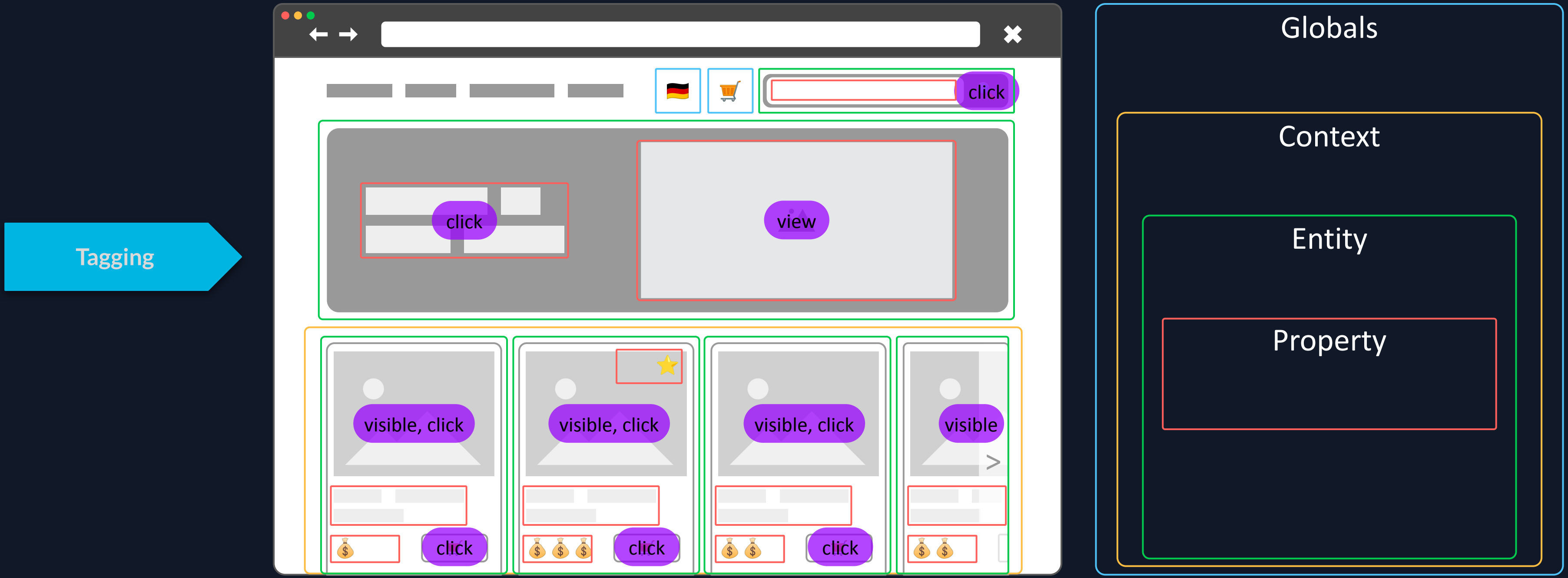
Data collection with walker.js

How to tag a webpage



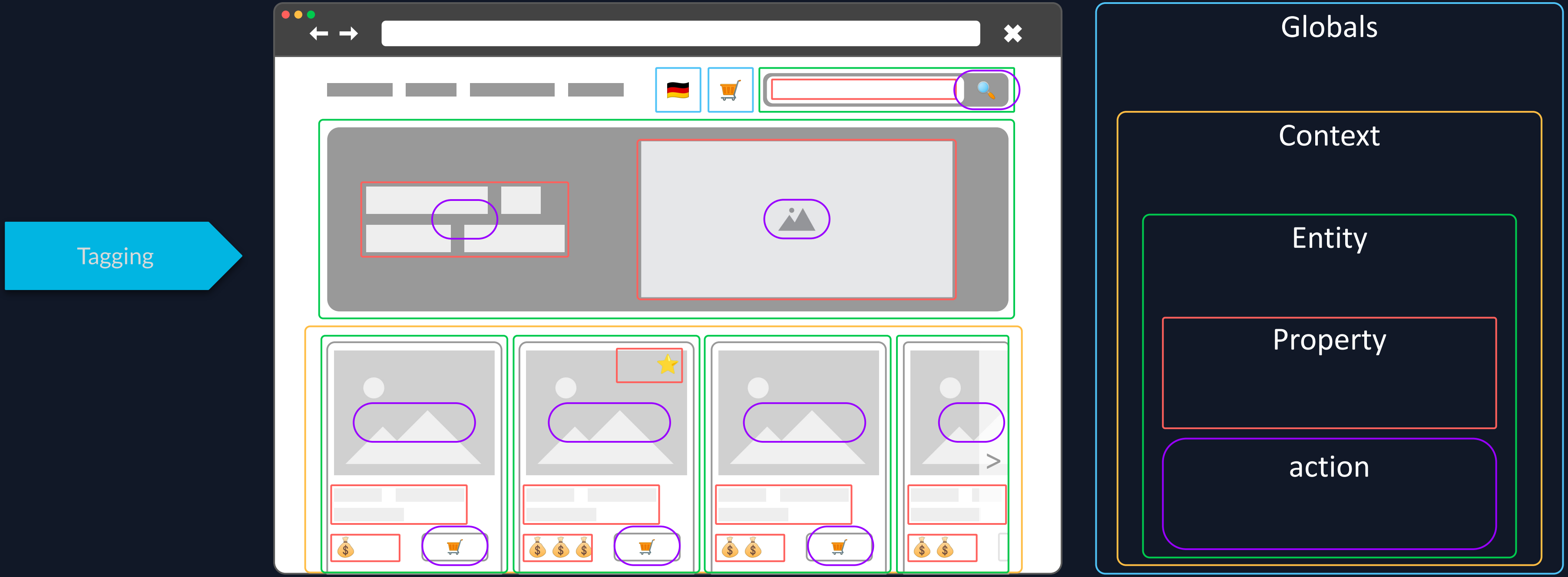
Data collection with walker.js

How to tag a webpage



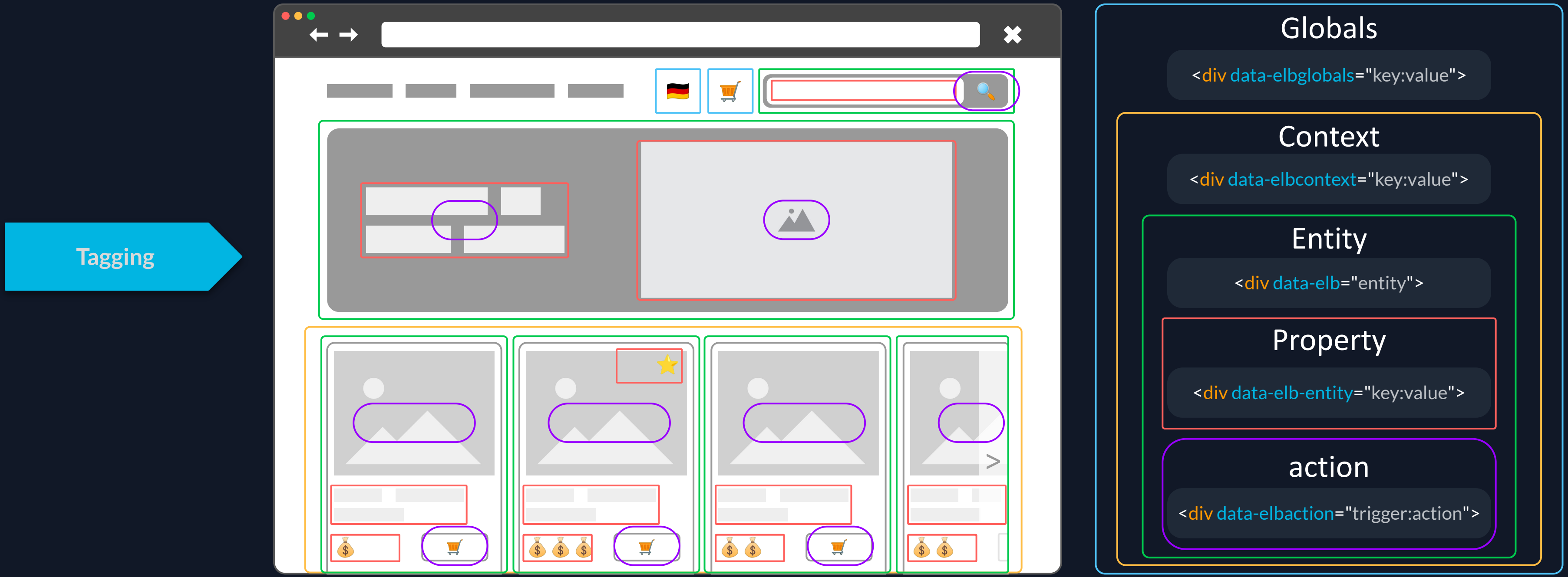
Data collection with walker.js

How to tag a webpage



Data collection with walker.js

How to tag a webpage



Globals

`<div data-elbglobals="key:value">`

Context

`<div data-elbcontext="key:value">`

Entity

`<div data-elb="entity">`

Property

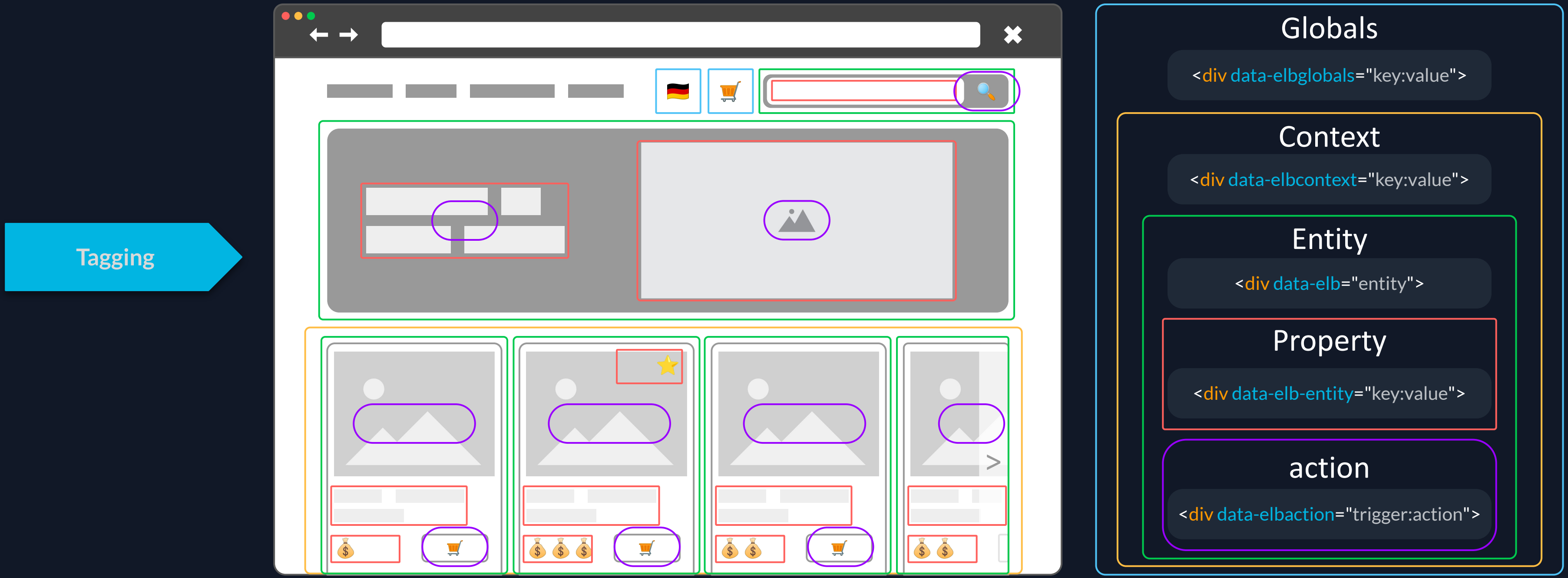
`<div data-elb-entity="key:value">`

action

`<div data-elbaction="trigger:action">`

Data collection with walker.js

How to tag a webpage



Globals

`<div data-elbglobals="key:value">`

Context

`<div data-elbcontext="key:value">`

Entity

`<div data-elb="entity">`

Property

`<div data-elb-entity="key:value">`

action

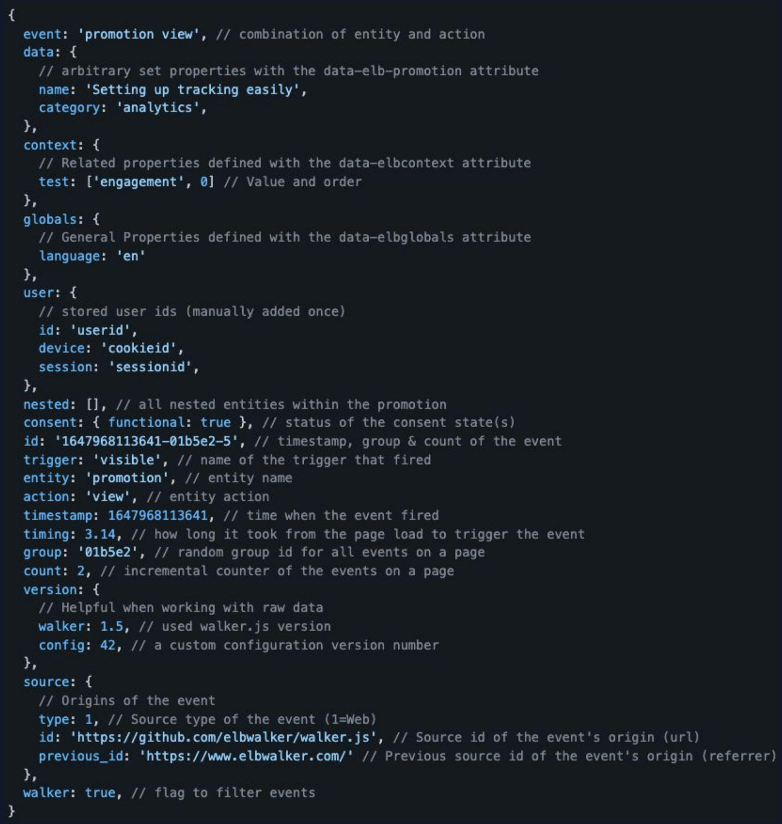
`<div data-elbaction="trigger:action">`

search start, promotion visible, promotion click, product visible, product click, product add

Data collection with walker.js

How to tag a webpage

Tagging



```
{
  event: 'promotion view', // combination of entity and action
  data: {
    // arbitrary set properties with the data-elb-promotion attribute
    name: 'Setting up tracking easily',
    category: 'analytics',
  },
  context: {
    // Related properties defined with the data-elbcontext attribute
    test: ['engagement', 0] // Value and order
  },
  globals: {
    // General Properties defined with the data-elbglobals attribute
    language: 'en'
  },
  user: {
    // stored user ids (manually added once)
    id: 'userid',
    device: 'cookieid',
    session: 'sessionid',
  },
  nested: [], // all nested entities within the promotion
  consent: { functional: true }, // status of the consent state(s)
  id: '1647968113641-01b5e2-5', // timestamp, group & count of the event
  trigger: 'visible', // name of the trigger that fired
  entity: 'promotion', // entity name
  action: 'view', // entity action
  timestamp: 1647968113641, // time when the event fired
  timing: 3.14, // how long it took from the page load to trigger the event
  group: '01b5e2', // random group id for all events on a page
  count: 2, // incremental counter of the events on a page
  version: {
    // Helpful when working with raw data
    walker: 1.5, // used walker.js version
    config: 42, // a custom configuration version number
  },
  source: {
    // Origins of the event
    type: 1, // Source type of the event (1=Web)
    id: 'https://github.com/elbwalker/walker.js', // Source id of the event's origin (url)
    previous_id: 'https://www.elbwalker.com/' // Previous source id of the event's origin (referrer)
  },
  walker: true, // flag to filter events
}
```

Home:
https://www.elbwalker.com/

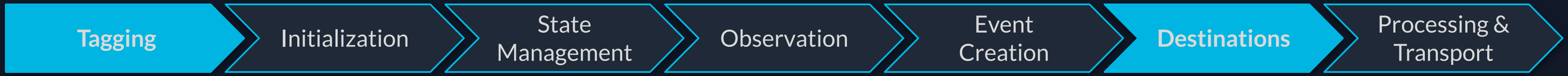
Tagging:
https://docs.elbwalker.com/master/basic-idea

Event concept:
https://www.elbwalker.com/blog/elbwalker-event-concept/

Repo:
https://github.com/elbwalker/walker.js

Data collection with walker.js

Adding destinations



Tagging

- Measurement Plan
- Configuration
- HTML Attributes

Initialization

- Loading State
- Event Listener
- Mutation Observer
- Updates

State Management

- Consent
- User IDs
- Event Queue
- Timings
- Misc

Observation

- Loading
- Clicking
- Scrolling
- Timing
- Custom

Event Creation

- Properties
- Context
- Globals
- Misc

Destinations

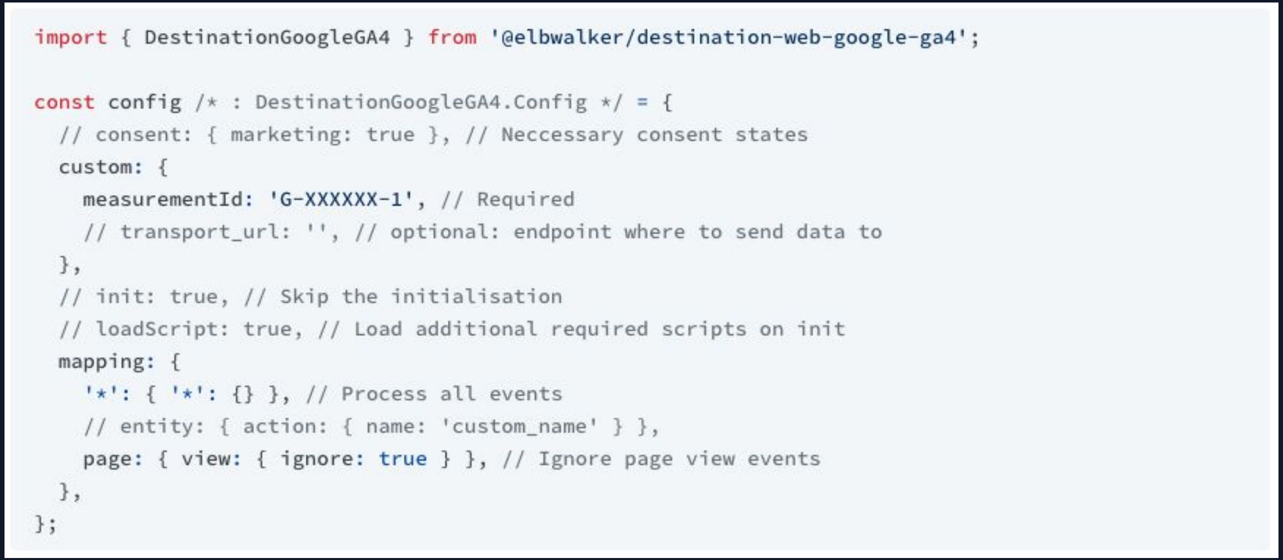
- Consent Permissions
- Custom Settings
- Event Settings

Processing & Transport

- Setup & Init
- Mapping
- Queueing
- Sending

Data collection with walker.js

Adding destinations

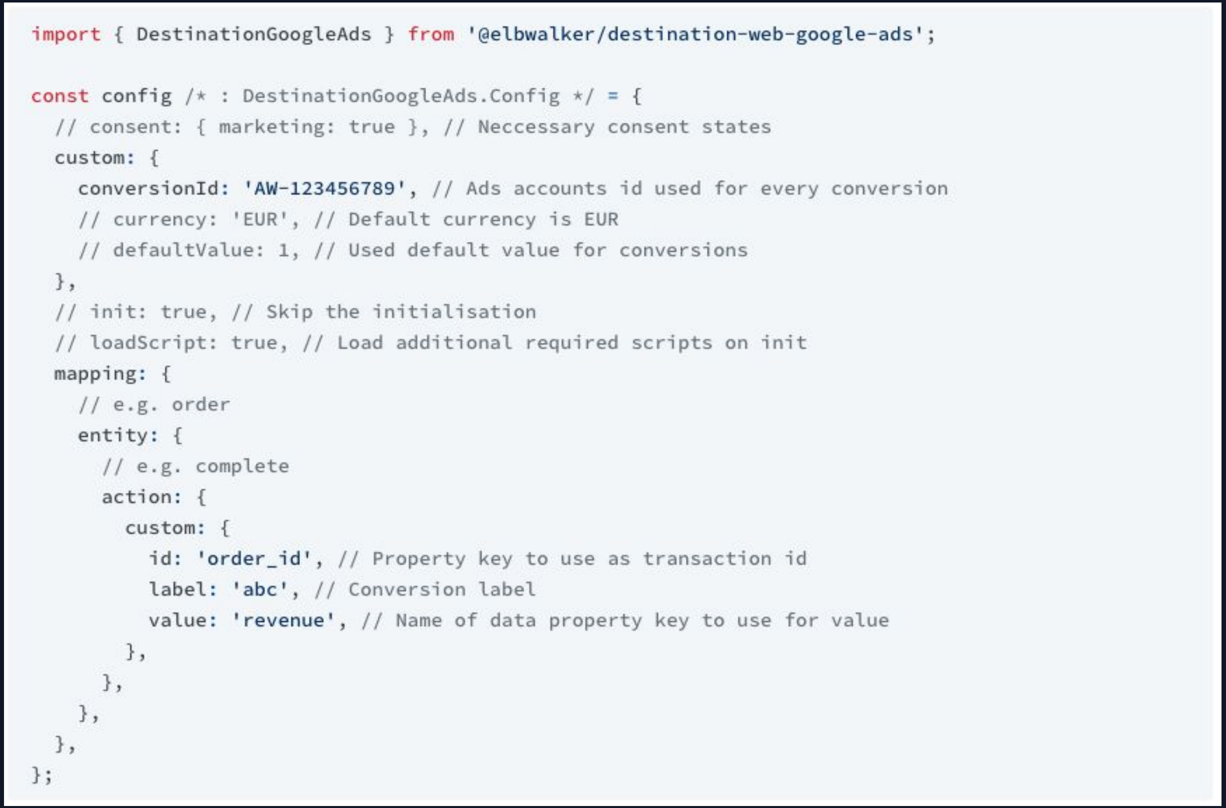


```
import { DestinationGoogleGA4 } from '@elbwalker/destination-web-google-ga4';

const config /* : DestinationGoogleGA4.Config */ = {
  // consent: { marketing: true }, // Neccessary consent states
  custom: {
    measurementId: 'G-XXXXXX-1', // Required
    // transport_url: '', // optional: endpoint where to send data to
  },
  // init: true, // Skip the initialisation
  // loadScript: true, // Load additional required scripts on init
  mapping: {
    '*': { '*': {} }, // Process all events
    // entity: { action: { name: 'custom_name' } },
    page: { view: { ignore: true } }, // Ignore page view events
  },
};
```

Destinations

Data collection with walker.js

Adding destinations

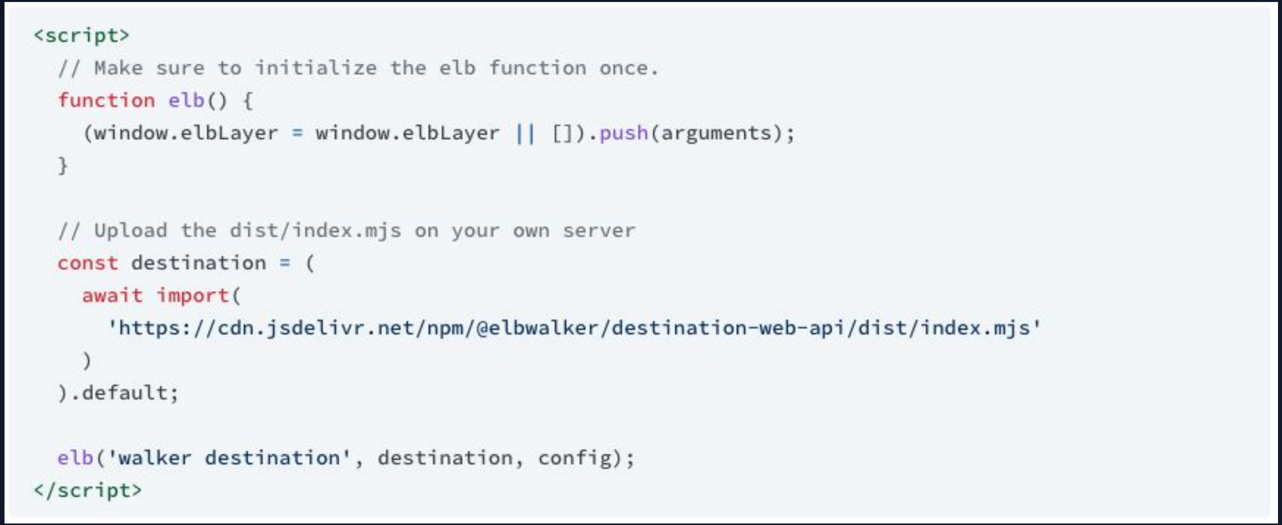


```
import { DestinationGoogleAds } from '@elbwalker/destination-web-google-ads';

const config /* : DestinationGoogleAds.Config */ = {
  // consent: { marketing: true }, // Neccessary consent states
  custom: {
    conversionId: 'AW-123456789', // Ads accounts id used for every conversion
    // currency: 'EUR', // Default currency is EUR
    // defaultValue: 1, // Used default value for conversions
  },
  // init: true, // Skip the initialisation
  // loadScript: true, // Load additional required scripts on init
  mapping: {
    // e.g. order
    entity: {
      // e.g. complete
      action: {
        custom: {
          id: 'order_id', // Property key to use as transaction id
          label: 'abc', // Conversion label
          value: 'revenue', // Name of data property key to use for value
        },
      },
    },
  },
};
```

Destinations

Data collection with walker.js

Adding destinations



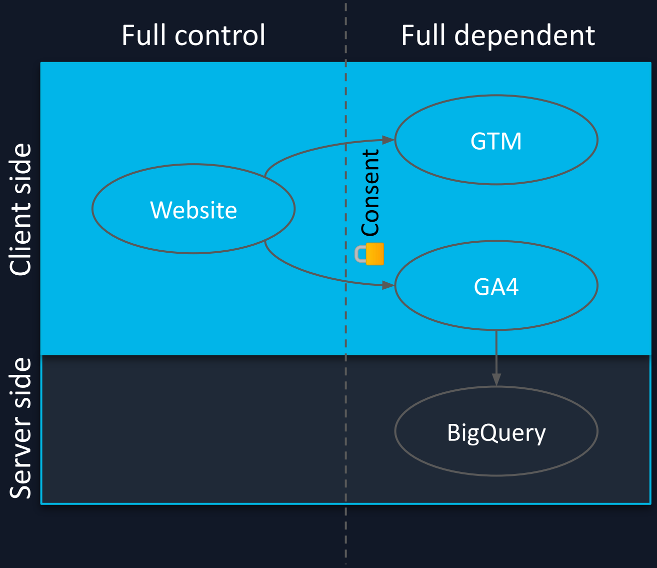
```
<script>
  // Make sure to initialize the elb function once.
  function elb() {
    (window.elbLayer = window.elbLayer || []).push(arguments);
  }

  // Upload the dist/index.mjs on your own server
  const destination = (
    await import(
      'https://cdn.jsdelivr.net/npm/@elbwalker/destination-web-api/dist/index.mjs'
    )
  ).default;

  elb('walker destination', destination, config);
</script>
```

Destinations

Data collection with walker.js

Setup styles

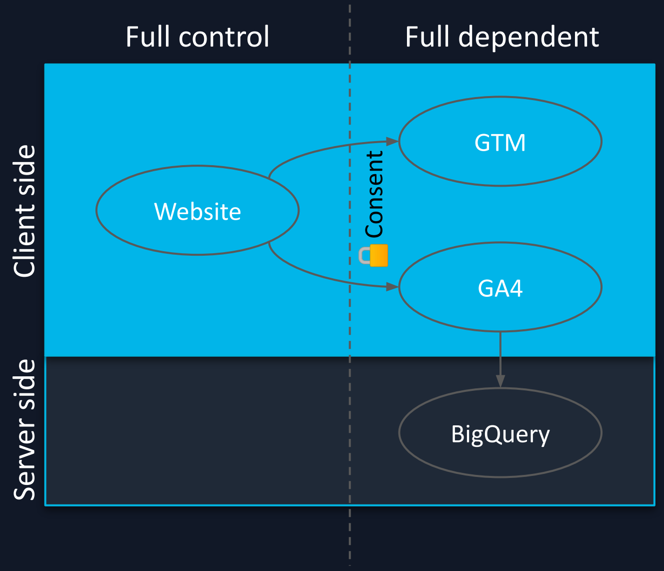
Third Party Setup



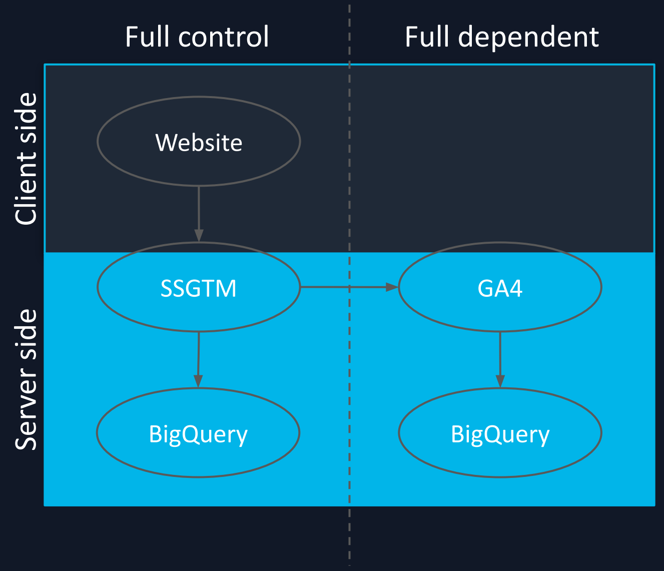
Data collection with walker.js

Setup styles

Third Party Setup



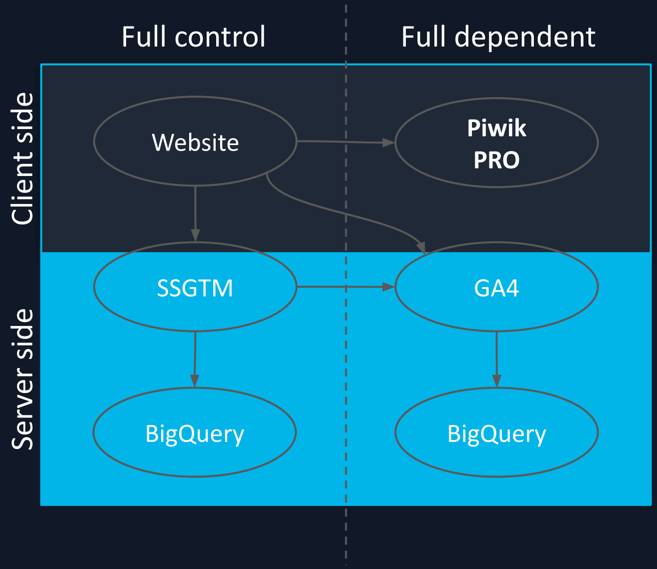
First Party Setup



Data collection with walker.js

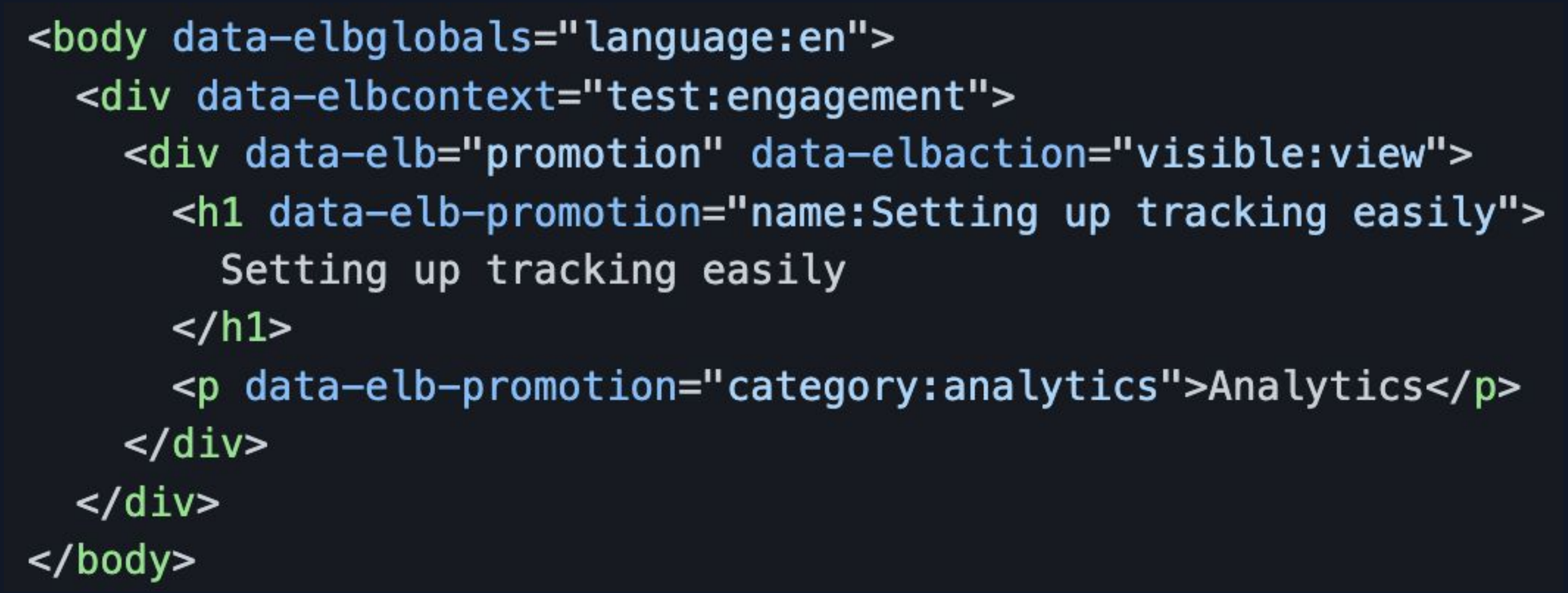
Setup styles

MULTI Party Setup



Data collection with walker.js

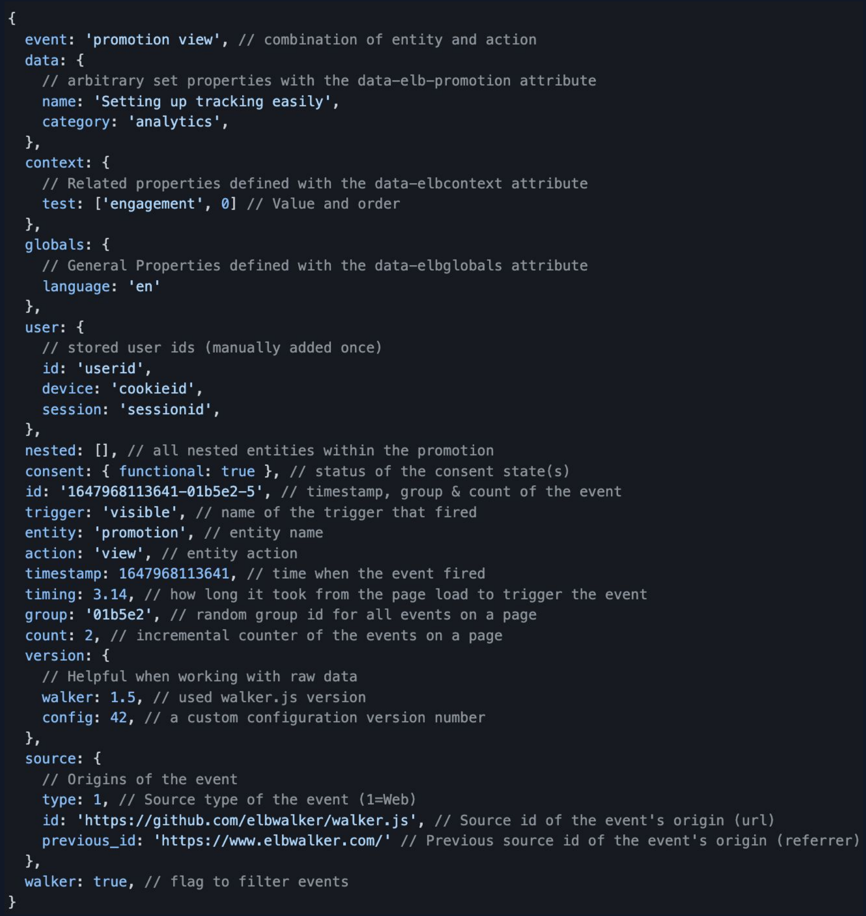
All we need is tagging



```
<body data-elbglobals="language:en">
  <div data-elbcontext="test:engagement">
    <div data-elb="promotion" data-elbaction="visible:view">
      <h1 data-elb-promotion="name:Setting up tracking easily">
        Setting up tracking easily
      </h1>
      <p data-elb-promotion="category:analytics">Analytics</p>
    </div>
  </div>
</body>
```

Data collection with walker.js

All we need is tagging



```
{
  event: 'promotion view', // combination of entity and action
  data: {
    // arbitrary set properties with the data-elb-promotion attribute
    name: 'Setting up tracking easily',
    category: 'analytics',
  },
  context: {
    // Related properties defined with the data-elbcontext attribute
    test: ['engagement', 0] // Value and order
  },
  globals: {
    // General Properties defined with the data-elbglobals attribute
    language: 'en'
  },
  user: {
    // stored user ids (manually added once)
    id: 'userid',
    device: 'cookieid',
    session: 'sessionid',
  },
  nested: [], // all nested entities within the promotion
  consent: { functional: true }, // status of the consent state(s)
  id: '1647968113641-01b5e2-5', // timestamp, group & count of the event
  trigger: 'visible', // name of the trigger that fired
  entity: 'promotion', // entity name
  action: 'view', // entity action
  timestamp: 1647968113641, // time when the event fired
  timing: 3.14, // how long it took from the page load to trigger the event
  group: '01b5e2', // random group id for all events on a page
  count: 2, // incremental counter of the events on a page
  version: {
    // Helpful when working with raw data
    walker: 1.5, // used walker.js version
    config: 42, // a custom configuration version number
  },
  source: {
    // Origins of the event
    type: 1, // Source type of the event (1=Web)
    id: 'https://github.com/elbwalker/walker.js', // Source id of the event's origin (url)
    previous_id: 'https://www.elbwalker.com/' // Previous source id of the event's origin (referrer)
  },
  walker: true, // flag to filter events
}
```

Kiitos paljon

hello@elbwalker.com

elbwalker