

Technologies for Digital Editions in the WWW: JavaScript

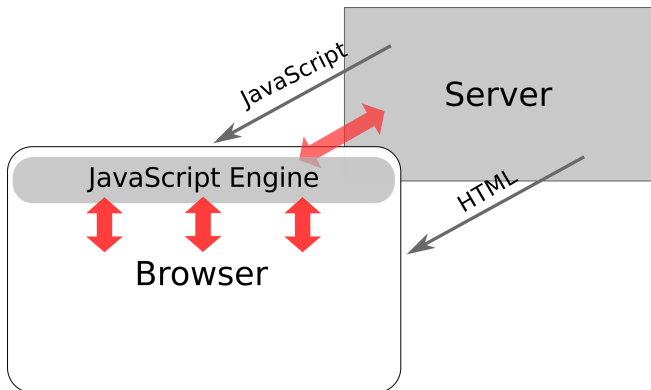
Gunter Vasold
gunter.vasold@uni-graz.at
@gunv

SpringSchool
Advanced XML/TEI technologies
Center for Informations Modelling, Graz University

April 13-17 2015

What is JavaScript?

- Programming language to be executed in web browsers



- Increasing importance for other purposes (server side programming, macro language, ...)
- Do not confuse with Java!

What can be done with JavaScript?

- A lot! E.g.
 - Make web pages dynamic
 - Send / fetch data in the background
- A few examples from Graz
 - Interconnect text with images:
`http://gams.uni-graz.at/fedora/get/o:me.1r/bdef:TEI/get/`
 - Interactive visualization `http://ginko.uni-graz.at/konrad4/rgraph`

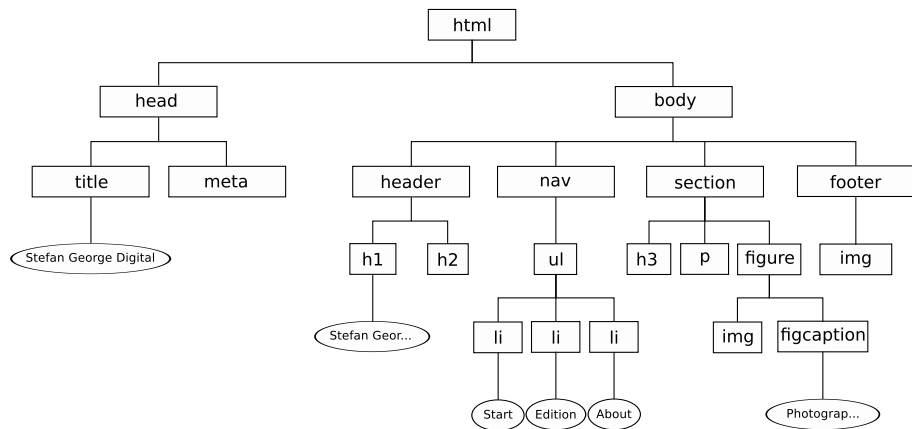
HTML as string

```

<meta name="keywords" content="George, Poetry, Symbolism, 1900, Typography" />
<meta charset="UTF-8" />
</head>
<body>
  <header>
    <h1>Stefan George Digital</h1>
    <h2>The edition of the lyrical works</h2>
  </header>
  <nav>
    <ul>
      <li><a href="index.html">Start</a></li>
      <li><a href="edition.html">Edition</a></li>
      <li><a href="about.html">About</a></li>
    </ul>
  </nav>
  <section>
    <h3>Welcome to Stefan George Digital</h3>
    <p>At the moment three works of Stefan George are
      transcribed, encoded, and enhanced with corresponding facsimiles. Further works will
      follow in Summer 2015.</p>
    <figure>
      
      <figcaption>Photography of Stefan George (1910)</figcaption>
    </figure>
  </section>
  <footer>
    
  </footer>
</body>
</html>--

```

HTML as Document Object Model (DOM)



Selecting a DOM Node with JavaScript

- `document.getElementsByTagName('tagname')`
- `document.getElementById('idvalue')`
- `document.querySelector('css-selector')`
- `document.querySelectorAll('css-selector')`

Code example 1 : Make first h2 red on page load

```
<script>  
    document.getElementsByTagName('h2')[0].style.color='red';  
</script>
```

Start JavaScript on event

Code example 2 : Make first h2 red on click

```
<button onclick="document.getElementsByTagName('h2')[0].style.color='red  
'">Make h2 red</button>
```

Using functions

Functions organize code in reusable sub-programs

Code example 3 : Make first headings red in function

```
1 <a href="#" onclick="makeHeadingsRed()">Make red</a>
2
3 <script>
4   function makeHeadingsRed() {
5       document.getElementsByTagName('h1')[0].style.color='red';
6       document.getElementsByTagName('h2')[0].style.color='red';
7       document.getElementsByTagName('h3')[0].style.color='red';
8   }
9 </script>
```


JavaScript files

- We can put all JavaScript Code in a separate file: js/edition.js
- Like external CSS files, an external JS files can be used from various html pages

Code example 4 : Content of edition.js

```
1  function makeHeadingsRed() {
2      document.getElementsByTagName('h1')[0].style.color='red';
3      document.getElementsByTagName('h2')[0].style.color='red';
4      document.getElementsByTagName('h3')[0].style.color='red';
5  }
```

- To use this external JavaScript, we have to reference it from our HTML

Code example 5 : Loading the JavaScript file

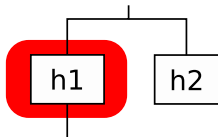
```
1  <script src="js/edition.js"></script>
```

Summary

- JavaScripts are (small) programs which are loaded from a server but run inside the browser
- JavaScript „sees“ the content of an HTML page as tree of node objects, where HTML tags, their attributes and texts are represented by node objects
- By selecting one or more such node objects, we are able to modify the object or the DOM tree by
 - changing its properties (css styles, text, . . .)
 - adding or removing node objects or sub trees
 - moving nodes objects or subtrees to other positions
- JavaScript code can be placed within HTML or in separate js files

jQuery

- A programming library for JavaScript
- Wraps JavaScript DOM nodes into jQuery objects



- Makes manipulating the DOM tree easier
- Many ready-to-use plugins
- www.jquery.com

Code example 6 : Including jQuery

```
<script src="js/jquery.min.js"></script>
```

Wrapping nodes into jQuery objects

Code example 7 : Wrapping DOM nodes

```
1 // The jQuery() funktion wraps a node into a jQuery object
2 var usagenode = jQuery('#usage');
3
4 // If the selector matches more than one node,
5 // a list of nodes is returned
6 var persons = jQuery('.person');
7
8 // Normaly we do not use the function name jQuery()
9 // but the shorter alias $()
10 var usagenode = $('#usage');
11 var persons = $('.person');
```

Using jQuery

Code example 8 : The pure JS version of makeHeadingsRed() we used before

```
1 function makeHeadingsRed() {
2     document.getElementsByTagName('h1')[0].style.color='red';
3     ...
```

Code example 9 : jQuery version of makeHeadingsRed()

```
8 // Make (all) headings red (jQuery version)
9 function makeHeadingsRed2() {
10     $('h1').css('color', 'red');
11     $('h2').css('color', 'red');
12     $('h3').css('color', 'red');
13 }
```

Code example 10 : Or even shorter

```
15 // Make (all) headings red: jQuery + pseudo-class
16 function makeHeadingsRed3() {
17     $(':header').css('color', 'red');
18 }
```

Example: highlighting nodes

Code example 11 : Highlighting nodes: HTML

```
1 <aside>
2   <form id="highlighters">
3     <div>
4       <input type="checkbox" id="highlightPlaces"></input>
5       <label for="highlightPlaces">Highlight places</label>
6     </div>
7     ...
8   </form>
9 </aside>
```

Code example 12 : Highlighting nodes: JavaScript*

```
1 $(document).ready(function() {
2   $('#highlightPlaces').on('change', function(evt) {
3     if($('#highlightPlaces').is(':checked')) {
4       $('.place').css('background-color', 'yellow');
5     } else {
6       $('.place').css('background-color', '');
7     }
8   });
9 });
```

Example: hiding/showing nodes

Adding a button to index.html

```
<button id="imgToggler" onclick="toggleImage()">Hide image</button>
```

Code example 13 : toggleImage()

```
1 function toggleImage() {
2     if($('#imgToggler').text() == 'Hide image') {
3         $('section figure').hide();
4         $('#imgToggler').text('Show image');
5     } else {
6         $('section figure').show();
7         $('#imgToggler').text('Hide image');
8     }
9 }
```

Using animations

```
1 ...
2 $('section figure').hide(500);
3 ...
4 $('section figure').show(2500);
5 ...
```

jQuery Plugins

- There are many ready-to-use plugins, which can be used without programming skills
- <https://plugins.jquery.com>

A first plugin: sticky-kit

- 1 Search <http://plugins.jquery.com> for 'sticky'
- 2 Download (or save) `jquery.sticky-kit.min.js` locally (<http://leafo.net/sticky-kit/>)
- 3 Include `jquery.sticky-kit.min.js` `<script src='js/jquery.sticky-kit.min.js'></script>`
- 4 Select node(s) and make them sticky. Make sure that `stick_in_parent()` is called as soon as the DOM tree has been constructed

Code example 14 : Make aside sticky

```
1 $(document).ready(function() {  
2     $('aside').stick_in_parent();  
3 });
```

Make images zoomable

- Search <https://plugins.jquery.com> for 'image' and 'zoom'
- Download plugin, e.g. `elevatezoom.js`
(<http://www.elevateweb.co.uk/image-zoom>)
- Include plugin into `edition.html`

```
<script src="js/jquery.elevateZoom.min.js"></script>
```
- Make images zoomable

Code example 15 : Make images zoomable

```
1 $(document).ready(function() {  
2     $('img').elevateZoom();  
3 });
```

Most plugins can be configured by a configuration object

Code example 16 : Make images inline zoomable

```
1 $(document).ready(function() {  
2     $('img').elevateZoom({  
3         zoomType: "inner",  
4         cursor: "crosshair"  
5     });  
6 });
```

Make images zoomable (2)

elevatezoom also has a magnifying lens option

Code example 17 : Magnifying lens

```
1 $(document).ready(function() {
2     $('img').elevateZoom({
3         zoomType: "lens",
4         lensShape: "round",
5         lensSize: 150
6     });
7 });
```

Use a second, higher resolution image for zooming

Code example 18 : Magnifying lens

```
1 
```

Make tables sortable with tablesorter

- Download tablesorter
- Include the js file `<script src="tablesorter/js/jquery.tablesorter.min.js"></script>`
- Attention: we also need to include a css file!
`<link rel="stylesheet" href="tablesorter/css/theme.default.css"/>`
- Activate tablesorter

Code example 19 : Make table columns sortable

```
1 $(document).ready(function() {  
2     $('#literature').tablesorter();  
3 });
```

Summary

- The jQuery library simplifies dealing with the DOM tree
- Provides powerful CSS selectors for nodes
- Extends JavaScript node objects with additional properties and methods

- There are many jQuery plugins which can be used with no or minimal programming knowledge
- <https://plugins.jquery.com> as the central plugin registry and the best way to find and evaluate plugins