How is the web written (and how to read it)

SICSS

CREST

INSTITUT POLYTECHNIQUE DE PARIS

Outline

1. The WWW is not the Internet

- A Brief History of the Internet
- Birth of the WWW
- Internet, HTML and Browsers

2. The WWW, Back & Front

- How is the web written?
- The structure of a simple webpage
- The www *millefeuille*

3. How-to in R

4. A note on APIs

5. Headless browsers

The WWW is not the Internet

1. A Brief History of the Internet

The WWW is not the Internet

1. A Brief History of the Internet

- *Competing Histories*
  - A “Cold War Technology”
    - DARPA and the race for the future of technology
    - DARPA & ARPA-NET (1958-1969)

Безопасное путешествие лайка

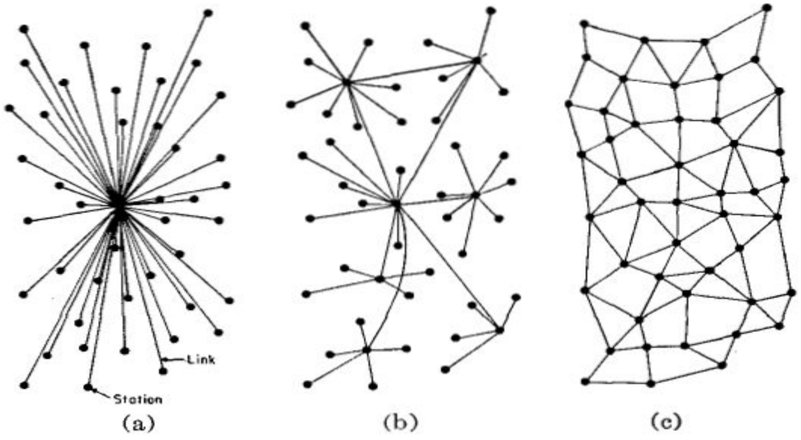


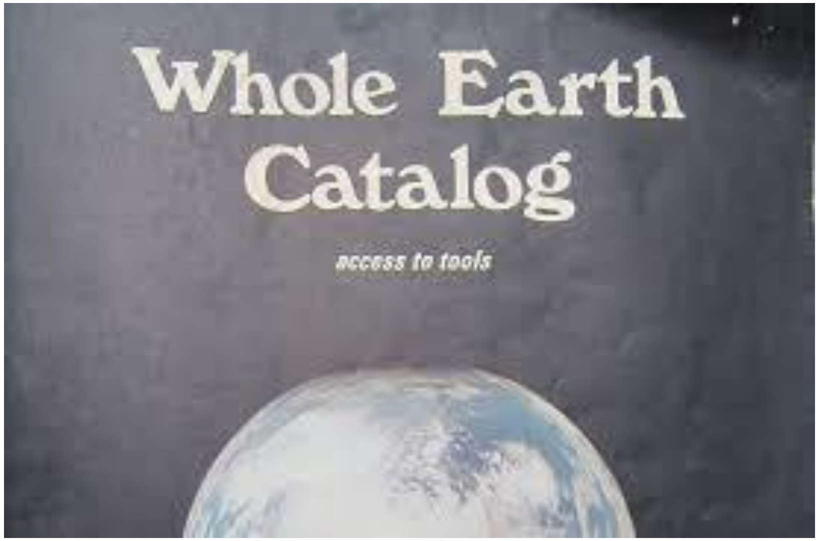
Fig. 1—(a) Centralized. (b) Decentralized. (c) Distributed networks.

Paul Baran, 1962

The WWW is not the Internet

1. A Brief History of the Internet

- *Competing Histories*
  - “The Hippies did it”: Libertarian origins
    - Augmenting individuals: D. Engelbart against technoscience
    - Collective collaboration and the rise of hackers: science (and connected computers) for the people.



The WWW is not the Internet

1. A Brief History of the Internet

- *Technological & Political Struggles*
  - From circuit switching to packet switching
  - A flurry of networks → TCP/IP (1978-1983)
  - Whose Technology?
  - 1980s: NSF grant to connect US universities

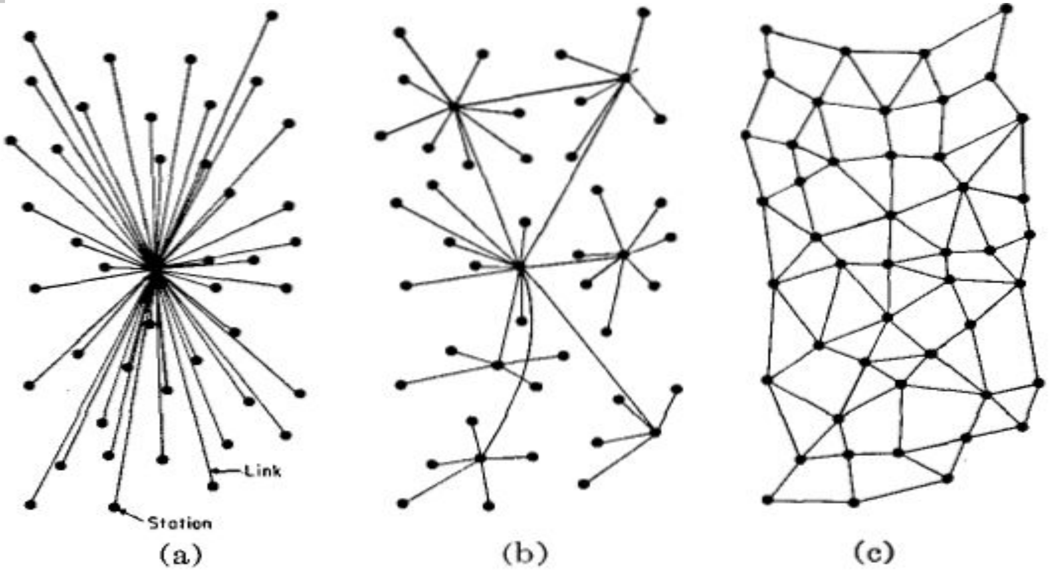


Fig. 1—(a) Centralized. (b) Decentralized. (c) Distributed networks.

The French Contender (the Minitel)

The WWW is not the Internet

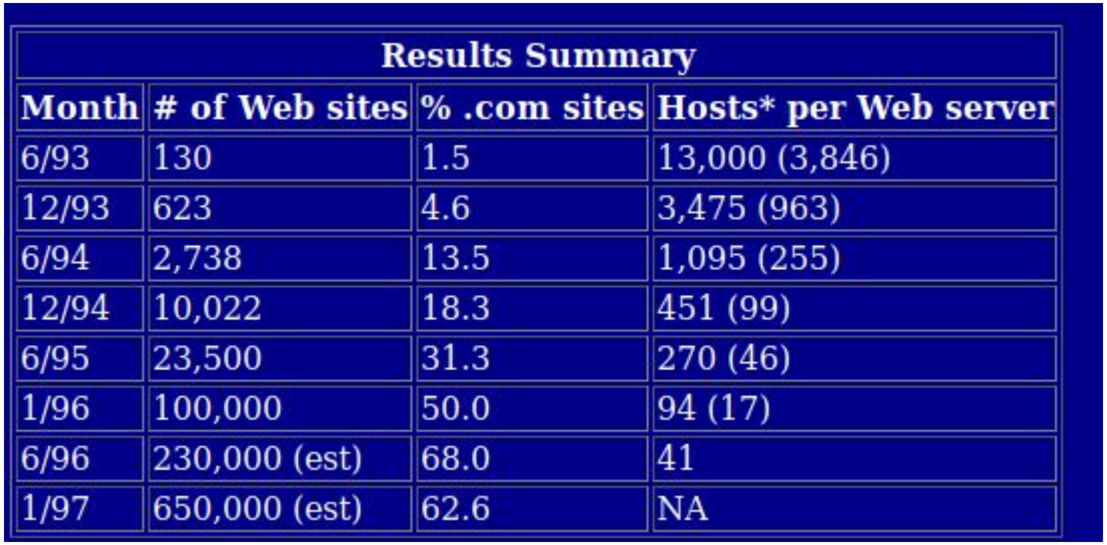
2. Birth of the WWW

- *A network that uses the internet*
  - Tim Berners-Lee (1989)
- *A decentralized sorting system*
  - Child of the previous evolution
  - The hyperlink at its core
- *HTML & HTTP*
  - HTML: Hyper Text Markup Language
  - HTTP: Hyper Text Transfer Protocol

The WWW is not the Internet

2. Birth of the WWW

- *The Internet as a Product*
  - IMDB 1990; Amazon, Ebay, Craiglist 1995;
  - Hotmail 1996; Yahoo, Google, Paypal 1998; 2001 Wiki;etc



Results Summary			
Month	# of Web sites	% .com sites	Hosts* per Web server
6/93	130	1.5	13,000 (3,846)
12/93	623	4.6	3,475 (963)
6/94	2,738	13.5	1,095 (255)
12/94	10,022	18.3	451 (99)
6/95	23,500	31.3	270 (46)
1/96	100,000	50.0	94 (17)
6/96	230,000 (est)	68.0	41
1/97	650,000 (est)	62.6	NA

Avenue Q: The internet is for...

The WWW is not the Internet

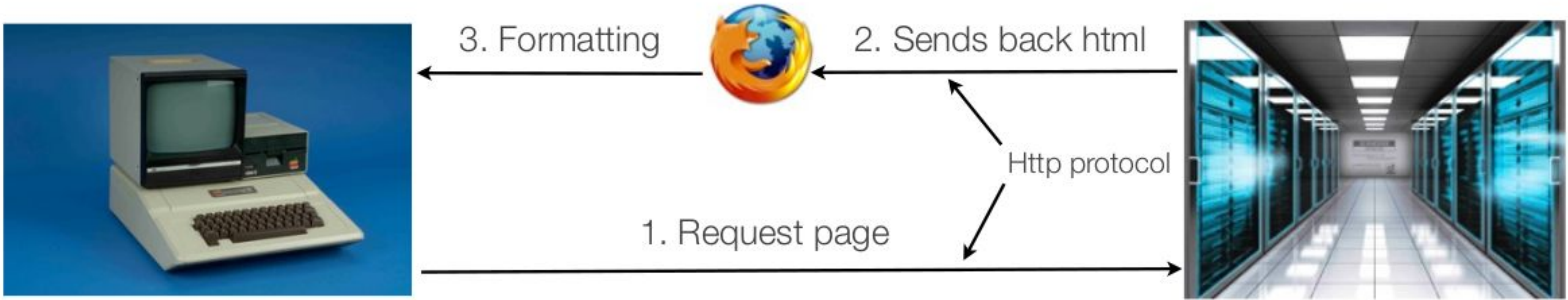
3. The Internet, the WWW and Browsers

- *Browsers were essential to popularizing the WWW*

The WWW is not the Internet

3. The Internet, the WWW and Browsers

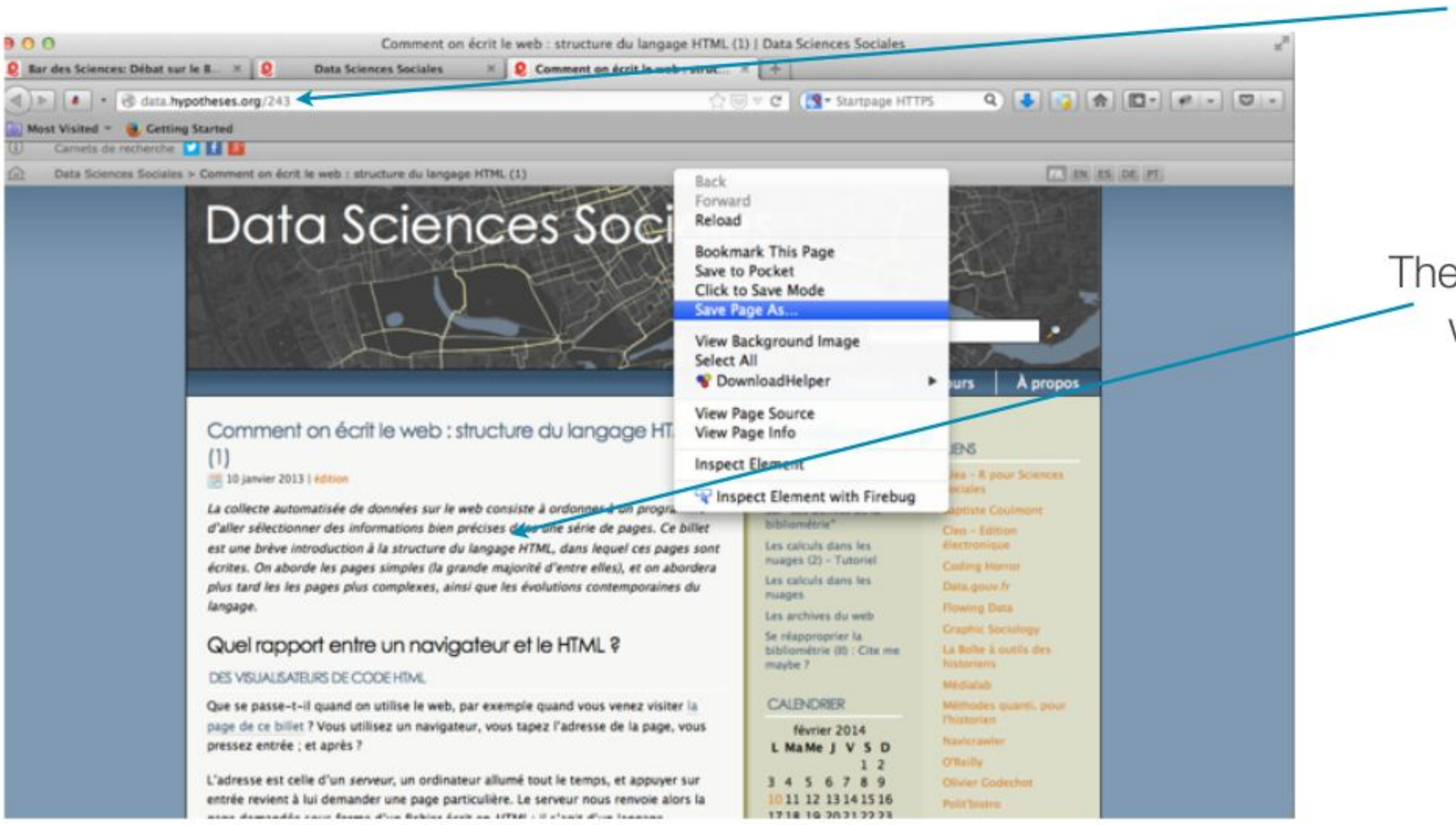
- *Browsers send, receive and interpret code*
- The web relies on the circulation of text in HTML
  - The **www** is based on communication between computers via a protocol called **HTTP**
  - Computers & pages are identified by their address, called a **URL** (Uniform Resource Locator)
  - **HTML** files are transferred and subsequently formatted into a legible format.



The WWW is not the Internet

3. The Internet, the WWW and Browsers

- *Browsers visualize code*



Comment on écrit le web : structure du langage HT (1)

10 janvier 2013 | édition

La collecte automatisée de données sur le web consiste à ordonner à un progra d'aller sélectionner des informations bien précises dans une série de pages. Ce billet est une brève introduction à la structure du langage HTML, dans lequel ces pages sont écrites. On aborde les pages simples (la grande majorité d'entre elles), et on abordera plus tard les les pages plus complexes, ainsi que les évolutions contemporaines du langage.

Quel rapport entre un navigateur et le HTML ?

DES VISUALISATEURS DE CODE HTML

Que se passe-t-il quand on utilise le web, par exemple quand vous venez visiter la page de ce billet ? Vous utilisez un navigateur, vous tapez l'adresse de la page, vous pressez entrée ; et après ?

L'adresse est celle d'un serveur, un ordinateur allumé tout le temps, et appuyer sur entrée revient à lui demander une page particulière. Le serveur nous renvoie alors la

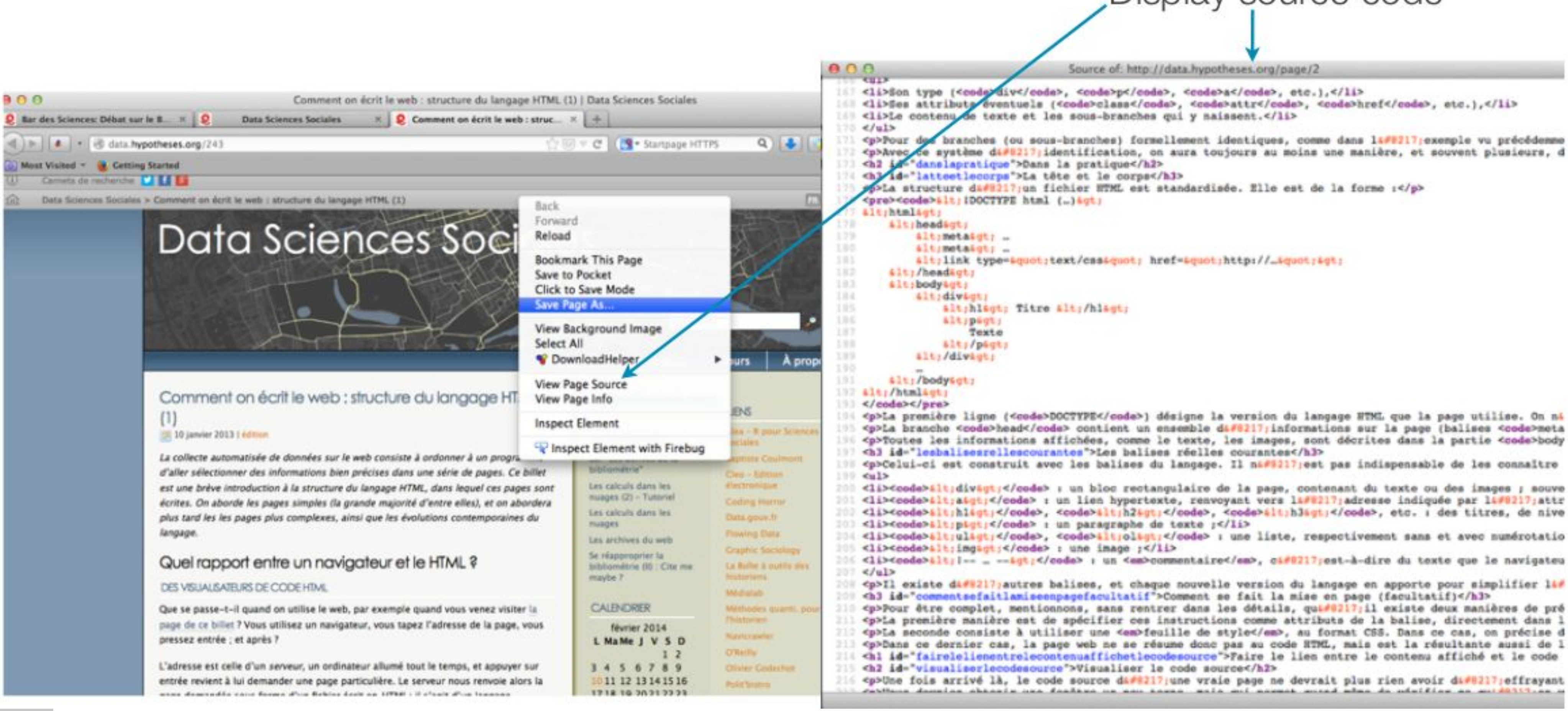
URL

The "body" of a webpage

The WWW is not the Internet

3. The Internet, the WWW and Browsers

- *Browsers visualize code*



Display source code

Comment on écrit le web : structure du langage HT (1)

10 janvier 2013 | édition

La collecte automatisée de données sur le web consiste à ordonner à un progra d'aller sélectionner des informations bien précises dans une série de pages. Ce billet est une brève introduction à la structure du langage HTML, dans lequel ces pages sont écrites. On aborde les pages simples (la grande majorité d'entre elles), et on abordera plus tard les les pages plus complexes, ainsi que les évolutions contemporaines du langage.

Quel rapport entre un navigateur et le HTML ?

DES VISUALISATEURS DE CODE HTML

Que se passe-t-il quand on utilise le web, par exemple quand vous venez visiter la page de ce billet ? Vous utilisez un navigateur, vous tapez l'adresse de la page, vous pressez entrée ; et après ?

L'adresse est celle d'un serveur, un ordinateur allumé tout le temps, et appuyer sur entrée revient à lui demander une page particulière. Le serveur nous renvoie alors la

```
<ul>
<li>Son type (<code>div</code>, <code>p</code>, <code>a</code>, etc.),</li>
<li>Ses attributs éventuels (<code>class</code>, <code>attr</code>, <code>href</code>, etc.),</li>
<li>Le contenu de texte et les sous-branches qui y naissent.</li>
</ul>
<p>Pour des branches (ou sous-branches) formellement identiques, comme dans l&#8217;exemple vu précédemme
<p>Avec ce système d&#8217;identification, on aura toujours au moins une manière, et souvent plusieurs, d
<h2 id="danslapratique">Dans la pratique</h2>
<h3 id="latteetlecorps">La tête et le corps</h3>
<p>La structure d&#8217;un fichier HTML est standardisée. Elle est de la forme :</p>
<pre><code>&lt;!DOCTYPE html (…)&gt;
&lt;html&gt;
    &lt;head&gt;
        &lt;meta&gt; …
        &lt;meta&gt; …
        &lt;link type=&quot;text/css&quot; href=&quot;http://…&quot;&gt;
    &lt;/head&gt;
    &lt;body&gt;
        &lt;div&gt;
            &lt;h1&gt; Titre &lt;/h1&gt;
            &lt;p&gt;
                Texte
            &lt;/p&gt;
        &lt;/div&gt;
        …
    &lt;/body&gt;
&lt;/html&gt;
</code></pre>
<p>La première ligne (<code>DOCTYPE</code>) désigne la version du langage HTML que la page utilise. On n&
<p>La branche <code>head</code> contient un ensemble d&#8217;informations sur la page (balises <code>meta
<p>Toutes les informations affichées, comme le texte, les images, sont décrites dans la partie <code>body
<h3 id="lesbalisesrellescourantes">Les balises réelles courantes</h3>
<p>Celui-ci est construit avec les balises du langage. Il n&#8217;est pas indispensable de les connaître
<ul>
<li><code>&lt;div&gt;</code> : un bloc rectangulaire de la page, contenant du texte ou des images ; souve
<li><code>&lt;a&gt;</code> : un lien hypertexte, renvoyant vers l&#8217;adresse indiquée par l&#8217;attr
<li><code>&lt;h1&gt;</code>, <code>&lt;h2&gt;</code>, <code>&lt;h3&gt;</code>, etc. : des titres, de nive
<li><code>&lt;p&gt;</code> : un paragraphe de texte ;</li>
<li><code>&lt;ul&gt;</code>, <code>&lt;ol&gt;</code> : une liste, respectivement sans et avec numérotatio
<li><code>&lt;img&gt;</code> : une image ;</li>
<li><code>&lt;!-- … --&gt;</code> : un <em>commentaire</em>, c&#8217;est-à-dire du texte que le navigateu
</ul>
<p>Il existe d&#8217;autres balises, et chaque nouvelle version du langage en apporte pour simplifier l&#
<h3 id="commentsefaitlamiseenpagefacultatif">Comment se fait la mise en page (facultatif)</h3>
<p>Pour être complet, mentionnons, sans rentrer dans les détails, qu&#8217;il existe deux manières de pré
<p>La première manière est de spécifier ces instructions comme attributs de la balise, directement dans l
<p>La seconde consiste à utiliser une <em>feuille de style</em>, au format CSS. Dans ce cas, on précise d
<p>Dans ce dernier cas, la page web ne se résume donc pas au code HTML, mais est la résultante aussi de l
<h1 id="fairelelienentrelecontenuaffichetlecodesource">Faire le lien entre le contenu affiché et le code
<h2 id="visualiserlecodesource">Visualiser le code source</h2>
<p>Une fois arrivé là, le code source d&#8217;une vraie page ne devrait plus rien avoir d&#8217;effrayant
```

The WWW, Back & Front

The WWW, Back & Front

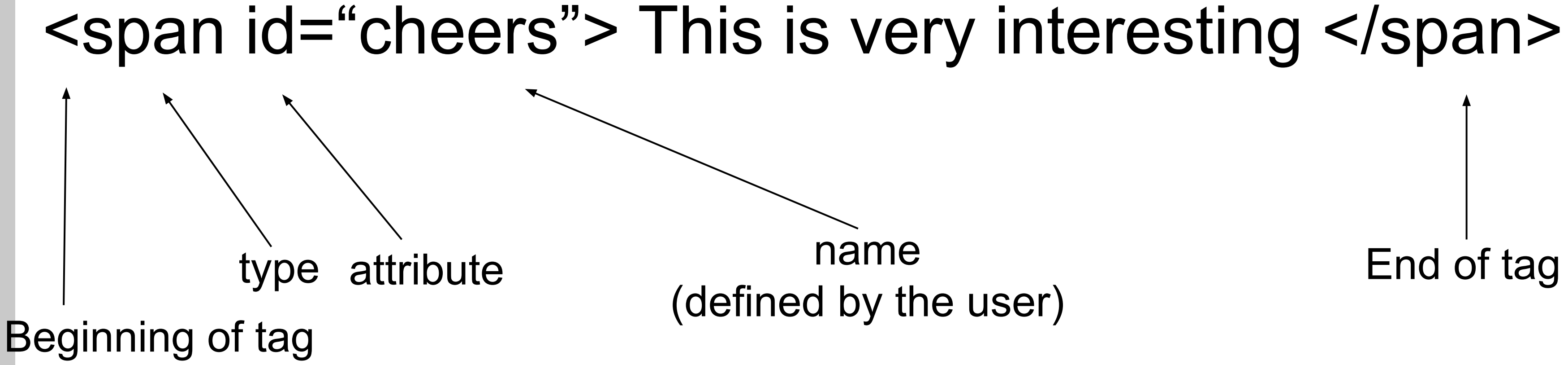
1. How is the Web Written?

- *The Web is Premised on HTML*
  - A “Rich Language”
  - A Structuring Principle: tags
    - HTML works with tags
      - The text displayed is surrounded by extra information, contained in these containers.
    - Ex. <TAG> This is very interesting </TAG>

The WWW, Back & Front

1. How is the Web Written?

- *The Web is Premised on HTML*
  - A “Rich Language” (more than meets the eye)
  - A Structuring Principle: **tags**
  - Tags have a **type** and an **attribute**
    - *Types* are fix (a, span, li, div). They have a limited number of attributes.
    - For more, see this page.



<span id=“cheers”> This is very interesting </span>

Beginning of tag

type

attribute

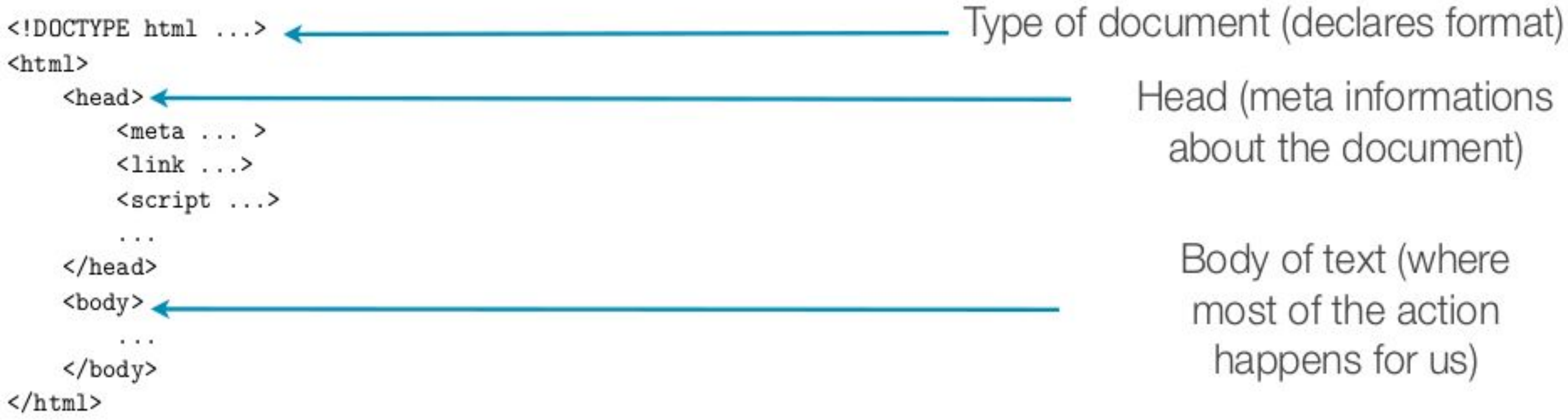
name
(defined by the user)

End of tag

The WWW, Back & Front

1. How is the Web Written?

- *Head & Body*



```
<!DOCTYPE html ...>
<html>
    <head>
        <meta ... >
        <link ...>
        <script ...>
        ...
    </head>
    <body>
        ...
    </body>
</html>
```

Type of document (declares format)

Head (meta informations about the document)

Body of text (where most of the action happens for us)

The WWW, Back & Front

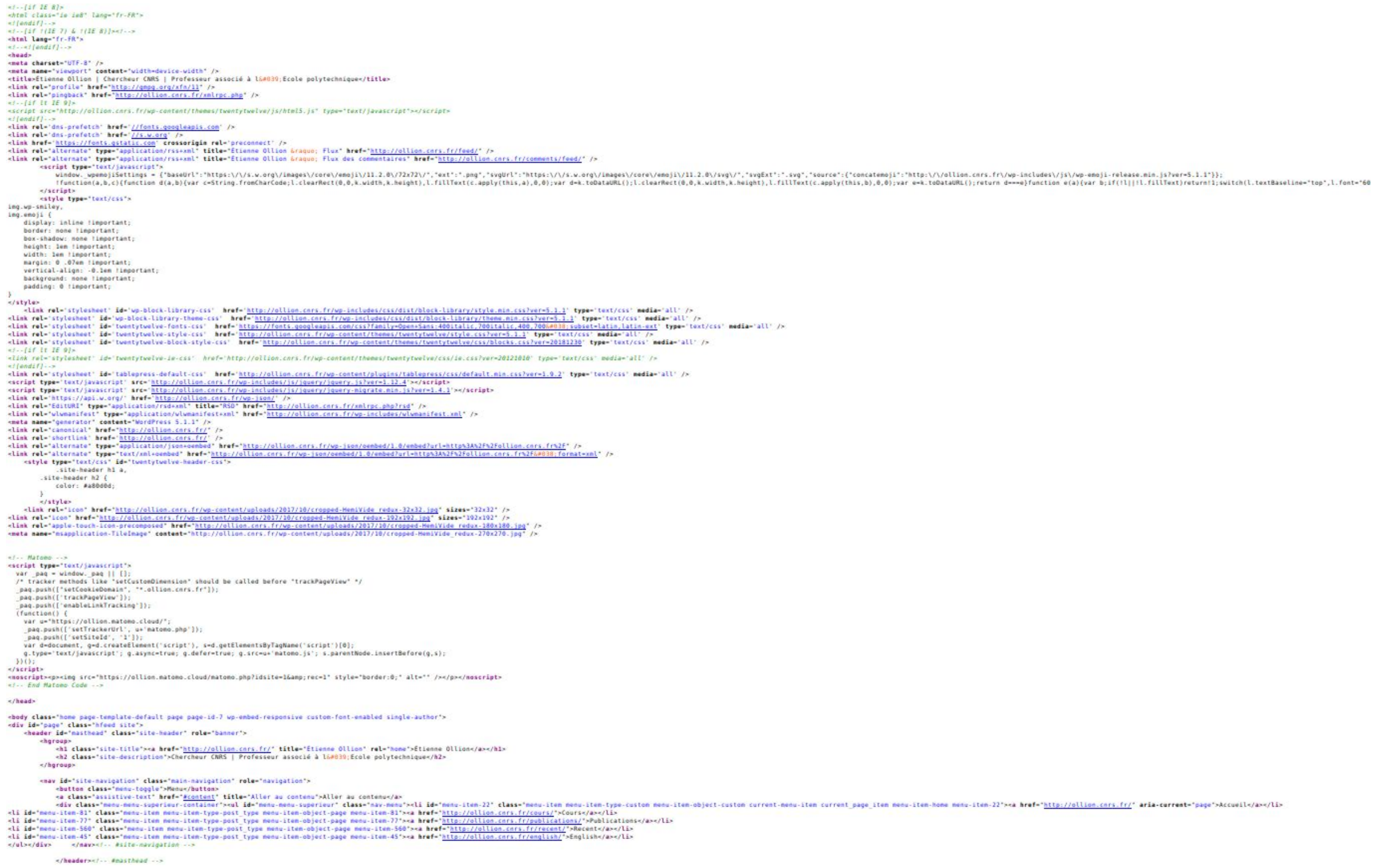
1. How is the Web Written?

- *A few common tags*
  - <div> : block of text
  - <p> : paragraph
  - <a> : hypertext link
  - <h1> : (resp. h2, h3, h4, h5) titles
  - <!...> : Comment

The WWW, Back & Front

2. The Structure of a Simple Webpage

- *An HTML File has a Tree Structure*

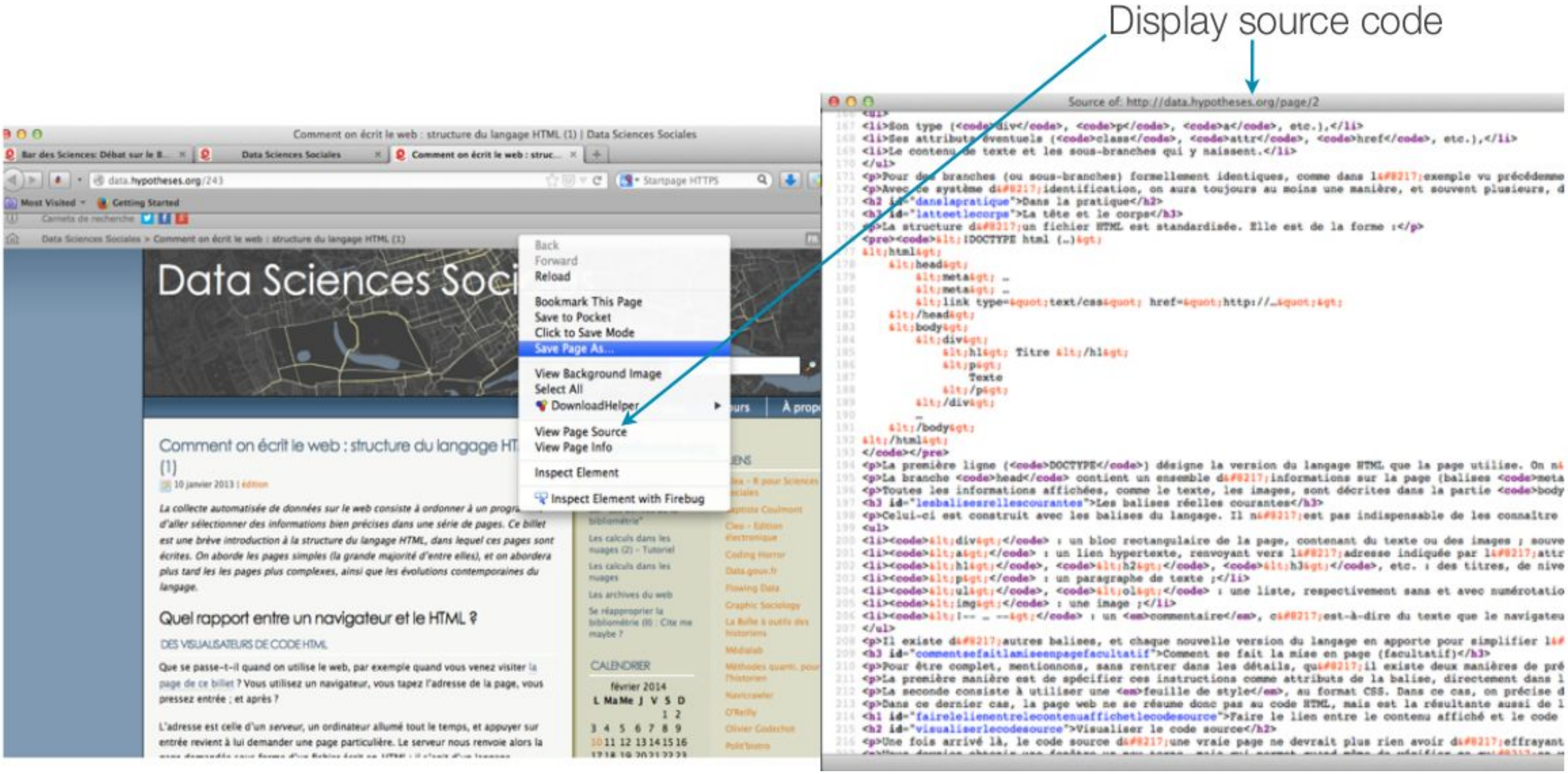


```
<!--[if IE 8]>
<html class="ie ie8" lang="fr-FR">
<![endif]-->
<!--[if !(IE 7) & !(IE 8)]><!-->
<html lang="fr-FR">
<!--<![endif]-->
<head>
<meta charset="UTF-8" />
<meta name="viewport" content="width=device-width" />
<title>Étienne Ollion | Chercheur CNRS | Professeur associé à l&#039;Ecole polytechnique</title>
<link rel="profile" href="http://gmpg.org/xfn/11" />
<link rel="pingback" href="http://ollion.cnrs.fr/xmlrpc.php" />
<!--[if lt IE 9]>
<script src="http://ollion.cnrs.fr/wp-content/themes/twentytwelve/js/html5.js" type="text/javascript"></script>
<![endif]-->
<link rel='dns-prefetch' href='//fonts.googleapis.com' />
<link rel='dns-prefetch' href='//s.w.org' />
<link href='https://fonts.gstatic.com' crossorigin rel='preconnect' />
<link rel="alternate" type="application/rss+xml" title="Étienne Ollion &raquo; Flux" href="http://ollion.cnrs.fr/feed/" />
<link rel="alternate" type="application/rss+xml" title="Étienne Ollion &raquo; Flux des commentaires" href="http://ollion.cnrs.fr/comments/feed/" />
        <script type="text/javascript">
            window._wpemojiSettings = {"baseUrl":"https:\/\/s.w.org\/images\/core\/emoji\/11.2.0\/72x72\/","ext":".png","svgUrl":"https:\/\/s.w.org\/images\/core\/emoji\/11.2.0\/svg\/","svgExt":".svg","source":{"concatemoji":"http:\/\/ollion.cnrs.fr\/wp-includes\/js\/wp-emoji-release.min.js?ver=5.1.1"}};
            !function(a,b,c){function d(a,b){var c=String.fromCharCode;l.clearRect(0,0,k.width,k.height),l.fillText(c.apply(this,a),0,0);var d=k.toDataURL();l.clearRect(0,0,k.width,k.height),l.fillText(c.apply(this,b),0,0);var e=k.toDataURL();return d===e}function e(a){var b;if(!l||!l.fillText)return!1;switch(l.textBaseline="top",l.font="60
        </script>
        <style type="text/css">
img.wp-smiley,
img.emoji {
    display: inline !important;
    border: none !important;
    box-shadow: none !important;
    height: 1em !important;
    width: 1em !important;
    margin: 0 .07em !important;
    vertical-align: -0.1em !important;
    background: none !important;
    padding: 0 !important;
}
</style>
    <link rel='stylesheet' id='wp-block-library-css'  href='http://ollion.cnrs.fr/wp-includes/css/dist/block-library/style.min.css?ver=5.1.1' type='text/css' media='all' />
<link rel='stylesheet' id='wp-block-library-theme-css'  href='http://ollion.cnrs.fr/wp-includes/css/dist/block-library/theme.min.css?ver=5.1.1' type='text/css' media='all' />
<link rel='stylesheet' id='twentytwelve-fonts-css'  href='https://fonts.googleapis.com/css?family=Open+Sans:400italic,700italic,400,700&#038;subset=latin,latin-ext' type='text/css' media='all' />
<link rel='stylesheet' id='twentytwelve-style-css'  href='http://ollion.cnrs.fr/wp-content/themes/twentytwelve/style.css?ver=5.1.1' type='text/css' media='all' />
<link rel='stylesheet' id='twentytwelve-block-style-css'  href='http://ollion.cnrs.fr/wp-content/themes/twentytwelve/css/blocks.css?ver=20181230' type='text/css' media='all' />
<!--[if lt IE 9]>
<link rel='stylesheet' id='twentytwelve-ie-css'  href='http://ollion.cnrs.fr/wp-content/themes/twentytwelve/css/ie.css?ver=20121010' type='text/css' media='all' />
<![endif]-->
<link rel='stylesheet' id='tablepress-default-css'  href='http://ollion.cnrs.fr/wp-content/plugins/tablepress/css/default.min.css?ver=1.9.2' type='text/css' media='all' />
<script type='text/javascript' src='http://ollion.cnrs.fr/wp-includes/js/jquery/jquery.js?ver=1.12.4'></script>
<script type='text/javascript' src='http://ollion.cnrs.fr/wp-includes/js/jquery/jquery-migrate.min.js?ver=1.4.1'></script>
<link rel='https://api.w.org/' href='http://ollion.cnrs.fr/wp-json/' />
<link rel="EditURI" type="application/rsd+xml" title="RSD" href="http://ollion.cnrs.fr/xmlrpc.php?rsd" />
<link rel="wlwmanifest" type="application/wlwmanifest+xml" href="http://ollion.cnrs.fr/wp-includes/wlwmanifest.xml" />
<meta name="generator" content="WordPress 5.1.1" />
<link rel="canonical" href="http://ollion.cnrs.fr/" />
<link rel='shortlink' href='http://ollion.cnrs.fr/' />
<link rel="alternate" type="application/json+oembed" href="http://ollion.cnrs.fr/wp-json/oembed/1.0/embed?url=http%3A%2F%2Follion.cnrs.fr%2F" />
<link rel="alternate" type="text/xml+oembed" href="http://ollion.cnrs.fr/wp-json/oembed/1.0/embed?url=http%3A%2F%2Follion.cnrs.fr%2F&#038;format=xml" />
    <style type="text/css" id="twentytwelve-header-css">
            .site-header h1 a,
        .site-header h2 {
            color: #a80d0d;
        }
        </style>
    <link rel="icon" href="http://ollion.cnrs.fr/wp-content/uploads/2017/10/cropped-HemiVide_redux-32x32.jpg" sizes="32x32" />
<link rel="icon" href="http://ollion.cnrs.fr/wp-content/uploads/2017/10/cropped-HemiVide_redux-192x192.jpg" sizes="192x192" />
<link rel="apple-touch-icon-precomposed" href="http://ollion.cnrs.fr/wp-content/uploads/2017/10/cropped-HemiVide_redux-180x180.jpg" />
<meta name="msapplication-TileImage" content="http://ollion.cnrs.fr/wp-content/uploads/2017/10/cropped-HemiVide_redux-270x270.jpg" />


<!-- Matomo -->
<script type="text/javascript">
  var _paq = window._paq || [];
  /* tracker methods like "setCustomDimension" should be called before "trackPageView" */
  _paq.push(["setCookieDomain", "*.ollion.cnrs.fr"]);
  _paq.push(['trackPageView']);
  _paq.push(['enableLinkTracking']);
  (function() {
    var u="https://ollion.matomo.cloud/";
    _paq.push(['setTrackerUrl', u+'matomo.php']);
    _paq.push(['setSiteId', '1']);
    var d=document, g=d.createElement('script'), s=d.getElementsByTagName('script')[0];
    g.type='text/javascript'; g.async=true; g.defer=true; g.src=u+'matomo.js'; s.parentNode.insertBefore(g,s);
  })();
</script>
<noscript><p><img src="https://ollion.matomo.cloud/matomo.php?idsite=1&amp;rec=1" style="border:0;" alt="" /></p></noscript>
<!-- End Matomo Code -->

</head>

<body class="home page-template-default page page-id-7 wp-embed-responsive custom-font-enabled single-author">
<div id="page" class="hfeed site">
    <header id="masthead" class="site-header" role="banner">
        <hgroup>
            <h1 class="site-title"><a href="http://ollion.cnrs.fr/" title="Étienne Ollion" rel="home">Étienne Ollion</a></h1>
            <h2 class="site-description">Chercheur CNRS | Professeur associé à l&#039;Ecole polytechnique</h2>
        </hgroup>

        <nav id="site-navigation" class="main-navigation" role="navigation">
            <button class="menu-toggle">Menu</button>
            <a class="assistive-text" href="#content" title="Aller au contenu">Aller au contenu</a>
            <div class="menu-menu-superieur-container"><ul id="menu-menu-superieur" class="nav-menu"><li id="menu-item-22" class="menu-item menu-item-type-custom menu-item-object-custom current-menu-item current_page_item menu-item-home menu-item-22"><a href="http://ollion.cnrs.fr/" aria-current="page">Accueil</a></li>
<li id="menu-item-81" class="menu-item menu-item-type-post_type menu-item-object-page menu-item-81"><a href="http://ollion.cnrs.fr/cours/">Cours</a></li>
<li id="menu-item-77" class="menu-item menu-item-type-post_type menu-item-object-page menu-item-77"><a href="http://ollion.cnrs.fr/publications/">Publications</a></li>
<li id="menu-item-560" class="menu-item menu-item-type-post_type menu-item-object-page menu-item-560"><a href="http://ollion.cnrs.fr/recent/">Recent</a></li>
<li id="menu-item-45" class="menu-item menu-item-type-post_type menu-item-object-page menu-item-45"><a href="http://ollion.cnrs.fr/english/">English</a></li>
</ul></div>     </nav><!-- #site-navigation -->

            </header><!-- #masthead -->
```

The WWW, Back & Front

2. The Structure of a Simple Webpage

- *What happens in the code is visible on the page*



Display source code

Comment on écrit le web : structure du langage HT (1)

10 janvier 2013 | édition

La collecte automatisée de données sur le web consiste à ordonner à un progra d'aller sélectionner des informations bien précises dans une série de pages. Ce billet est une brève introduction à la structure du langage HTML, dans lequel ces pages sont écrites. On aborde les pages simples (la grande majorité d'entre elles), et on abordera plus tard les les pages plus complexes, ainsi que les évolutions contemporaines du langage.

Quel rapport entre un navigateur et le HTML ?

DES VISUALISATEURS DE CODE HTML

Que se passe-t-il quand on utilise le web, par exemple quand vous venez visiter la page de ce billet ? Vous utilisez un navigateur, vous tapez l'adresse de la page, vous pressez entrée ; et après ?

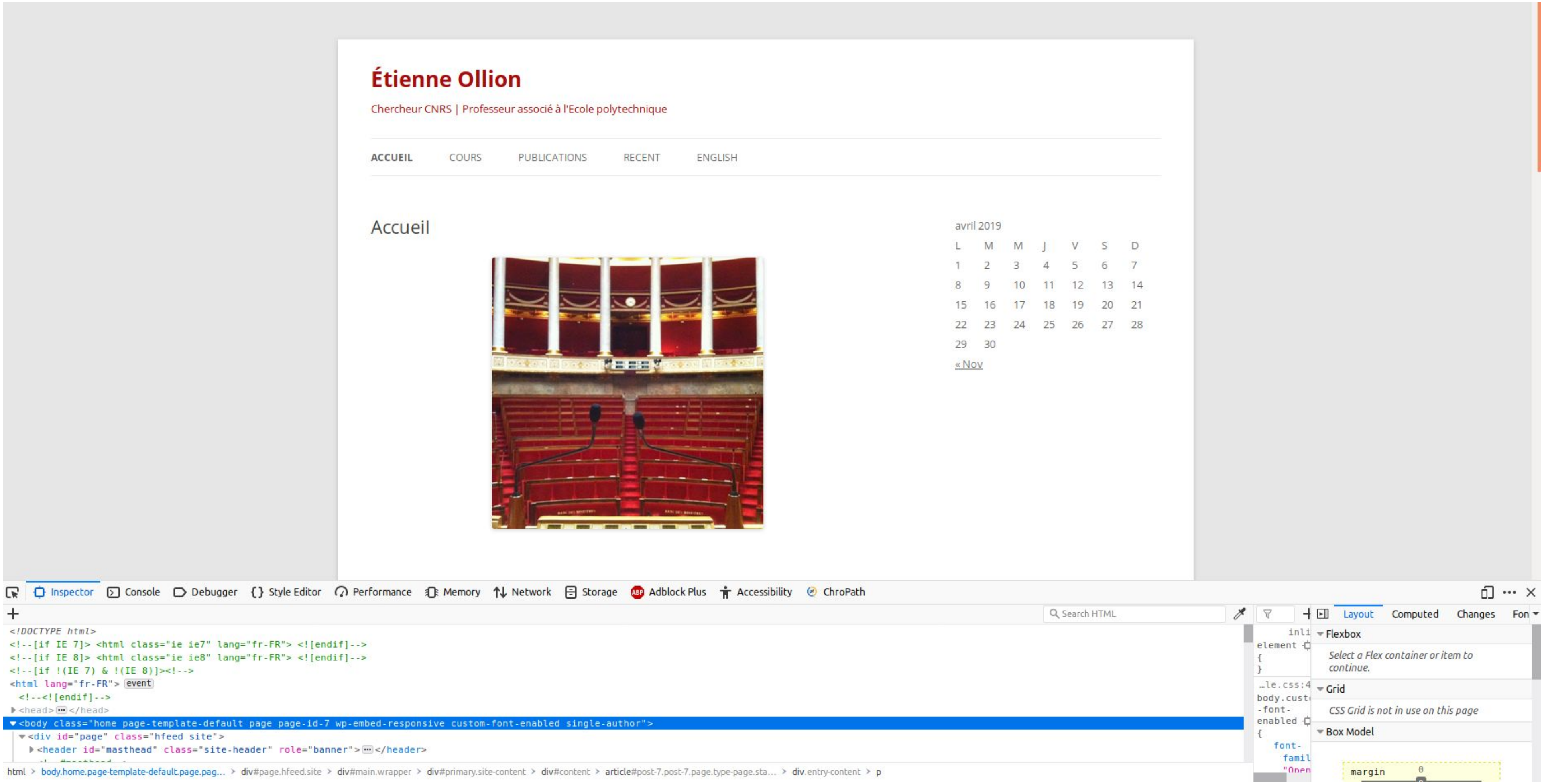
L'adresse est celle d'un serveur, un ordinateur allumé tout le temps, et appuyer sur entrée revient à lui demander une page particulière. Le serveur nous renvoie alors la

```
<li>Son type (<code>div</code>, <code>p</code>, <code>a</code>, etc.),</li>
<li>Ses attributs éventuels (<code>class</code>, <code>attr</code>, <code>href</code>, etc.),</li>
<li>Le contenu de texte et les sous-branches qui y naissent.</li>
</ul>
<p>Pour des branches (ou sous-branches) formellement identiques, comme dans l&#8217;exemple vu précédemme
<p>Avec ce système d&#8217;identification, on aura toujours au moins une manière, et souvent plusieurs, d
<h2 id="danslapratique">Dans la pratique</h2>
<h3 id="latteetlecorps">La tête et le corps</h3>
<p>La structure d&#8217;un fichier HTML est standardisée. Elle est de la forme :</p>
<pre><code>&lt;!DOCTYPE html (…)&gt;
&lt;html&gt;
    &lt;head&gt;
        &lt;meta&gt; …
        &lt;meta&gt; …
        &lt;link type=&quot;text/css&quot; href=&quot;http://…&quot;&gt;
    &lt;/head&gt;
    &lt;body&gt;
        &lt;div&gt;
            &lt;h1&gt; Titre &lt;/h1&gt;
            &lt;p&gt;
                Texte
            &lt;/p&gt;
        &lt;/div&gt;
        …
    &lt;/body&gt;
&lt;/html&gt;
</code></pre>
<p>La première ligne (<code>DOCTYPE</code>) désigne la version du langage HTML que la page utilise. On n&
<p>La branche <code>head</code> contient un ensemble d&#8217;informations sur la page (balises <code>meta
<p>Toutes les informations affichées, comme le texte, les images, sont décrites dans la partie <code>body
<h3 id="lesbalisesrellescourantes">Les balises réelles courantes</h3>
<p>Celui-ci est construit avec les balises du langage. Il n&#8217;est pas indispensable de les connaître
<ul>
<li><code>&lt;div&gt;</code> : un bloc rectangulaire de la page, contenant du texte ou des images ; souve
<li><code>&lt;a&gt;</code> : un lien hypertexte, renvoyant vers l&#8217;adresse indiquée par l&#8217;attr
<li><code>&lt;h1&gt;</code>, <code>&lt;h2&gt;</code>, <code>&lt;h3&gt;</code>, etc. : des titres, de nive
<li><code>&lt;p&gt;</code> : un paragraphe de texte ;</li>
<li><code>&lt;ul&gt;</code>, <code>&lt;ol&gt;</code> : une liste, respectivement sans et avec numérotatio
<li><code>&lt;img&gt;</code> : une image ;</li>
<li><code>&lt;!-- … --&gt;</code> : un <em>commentaire</em>, c&#8217;est-à-dire du texte que le navigateu
</ul>
<p>Il existe d&#8217;autres balises, et chaque nouvelle version du langage en apporte pour simplifier l&#
<h3 id="commentsefaitlamiseenpagefacultatif">Comment se fait la mise en page (facultatif)</h3>
<p>Pour être complet, mentionnons, sans rentrer dans les détails, qu&#8217;il existe deux manières de pré
<p>La première manière est de spécifier ces instructions comme attributs de la balise, directement dans l
<p>La seconde consiste à utiliser une <em>feuille de style</em>, au format CSS. Dans ce cas, on précise d
<p>Dans ce dernier cas, la page web ne se résume donc pas au code HTML, mais est la résultante aussi de l
<h1 id="fairelelienentrelecontenuaffichetlecodesource">Faire le lien entre le contenu affiché et le code
<h2 id="visualiserlecodesource">Visualiser le code source</h2>
<p>Une fois arrivé là, le code source d&#8217;une vraie page ne devrait plus rien avoir d&#8217;effrayant
```

The WWW, Back & Front

2. The Structure of a Simple Webpage

- *Tools in your browsers help you “inspect elements”*



Étienne Ollion

Chercheur CNRS | Professeur associé à l'Ecole polytechnique

Accueil

avril 2019

L	M	M	J	V	S	D
1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28
29	30					

« Nov

```
<!DOCTYPE html>
<!--[if IE 7]> <html class="ie ie7" lang="fr-FR"> <![endif]-->
<!--[if IE 8]> <html class="ie ie8" lang="fr-FR"> <![endif]-->
<!--[if !(IE 7) & !(IE 8)]><!-->
<html lang="fr-FR"> event
  <!--<![endif]-->
 <head>…</head>
 <body class="home page-template-default page page-id-7 wp-embed-responsive custom-font-enabled single-author">
  <div id="page" class="hfeed site">
   <header id="masthead" class="site-header" role="banner">…</header>
```

html > body.home.page-template-default.page.pag… > div#page.hfeed.site > div#main.wrapper > div#primary.site-content > div#content > article#post-7.post-7.page.type-page.sta… > div.entry-content > p

```
inli
element {
}
…le.css:4
body.cust
-font-
enabled {
  font-
   famil
   "Open
```

Flexbox

Select a Flex container or item to continue.

Grid

CSS Grid is not in use on this page

Box Model

The WWW, Back & Front

3. The www *millefeuille*

- A webpage is built on HTML

- And it includes other types of files
  - Pictures
  - Content Style Sheet (CSS)
  - Javascript

> Increasingly, the web has become a *millefeuille*

The WWW, Back & Front

3. The www *millefeuille*

> Increasingly, the web has become a *millefeuille*

This has consequences for scraping. But keep in mind that a regularity on the screen means regularity in the code. We are going to use this

How-to in R

1. Finding your way around

There is a wealth of dedicated libraries

This page maintains a list of all that there is at moment (and it’s plenty)

Scraping: *httr or* ***rvest***

Selecting in HTML (or XML): *XML or* ***rvest***
Selection in json: *rjson*, *rjsonio, jsonlite...*

How-to in R

2. Basic instructions

read_html() will read the page and transform it into an XML document.

Thus,

read_html("https://sicss.io/2022/paris/schedule") will output the source code of the schedule page for the SICSS-Paris program.

How-to in R

2. Basic instructions

How-to in R

2. Basic instructions

Yes, in ~70% of the cases, all you need to do to scrape a page is to do

read_html("*PAGE*")

How-to in R

Sometimes that won’t be enough, **and the website will see the evil crawler in you**.

You will need to dress-up like an honest browser

This goes through the user_agent command (*httr*)

```
user_agent("Mozilla/5.0 (Macinstosh;U;
Intel MacOS X 10.6; en-US")
```

```
user_agent("roger.rabbit@gmail.com")
```

“On the Internet, nobody knows you're a dog.”

How-to in R

Sometimes that won’t be enough, **because you’ll need cookies**.

With rvest, you will have to create a **session**, which stores the said cookies and allows you to navigate from there.

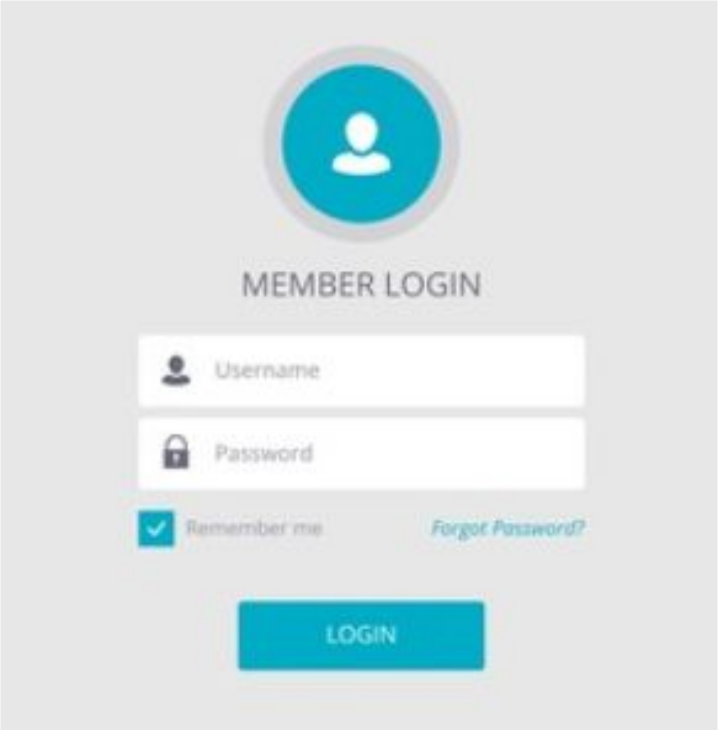
And then you’ll use **read_html()**

How-to in R

Sometimes that won’t be enough, **because you’ll need to log in**.

You’ll use a function called **html_form(),**

And then you’ll use **read_html()**



How-to in R

Once you have done that:

Great news: we are back to square 1!

(You could have copied and pasted the source code in your console, couldn’t you?)

Except that now, it is in R, and you can use it

A quick note on APIs

APIs

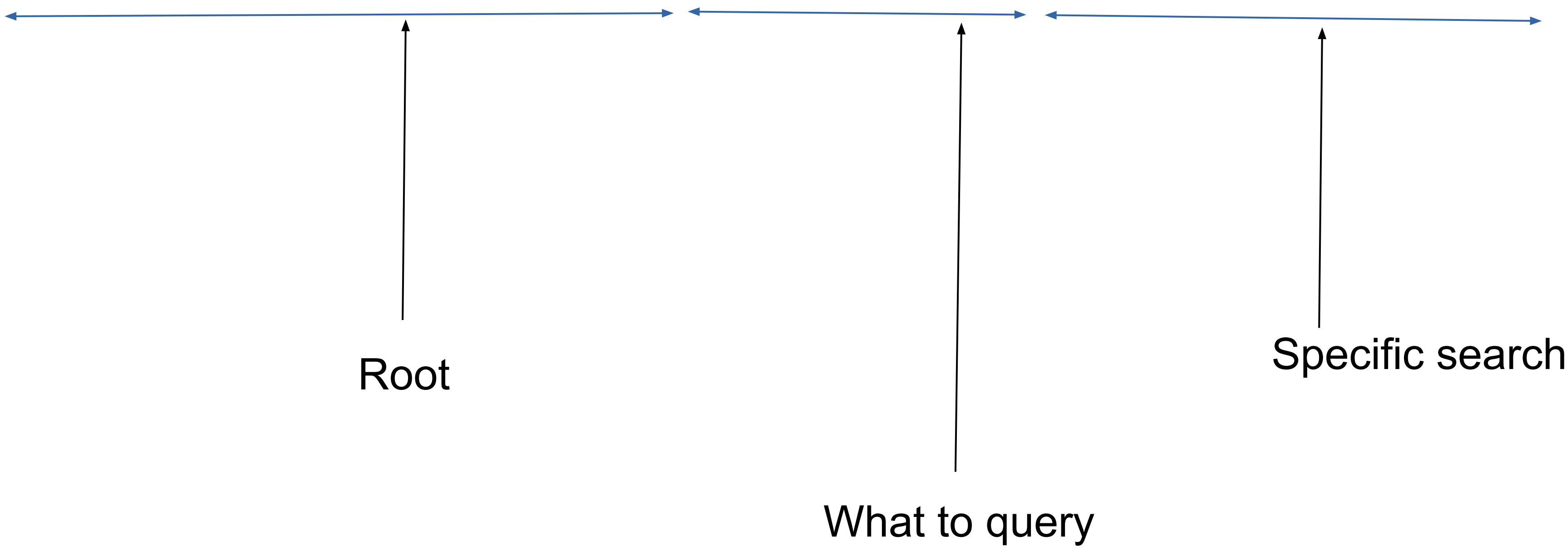
Application Programmers Interface

- A feature of the web 2.0
- Legal, and often easier
- Different forms: login, with rate limits, or just open

APIs

A simple yet elaborate API

- https://www.openstreetmap.org/search?query=ecole%20polytechnique%20paris



Root

What to query

Specific search

APIs

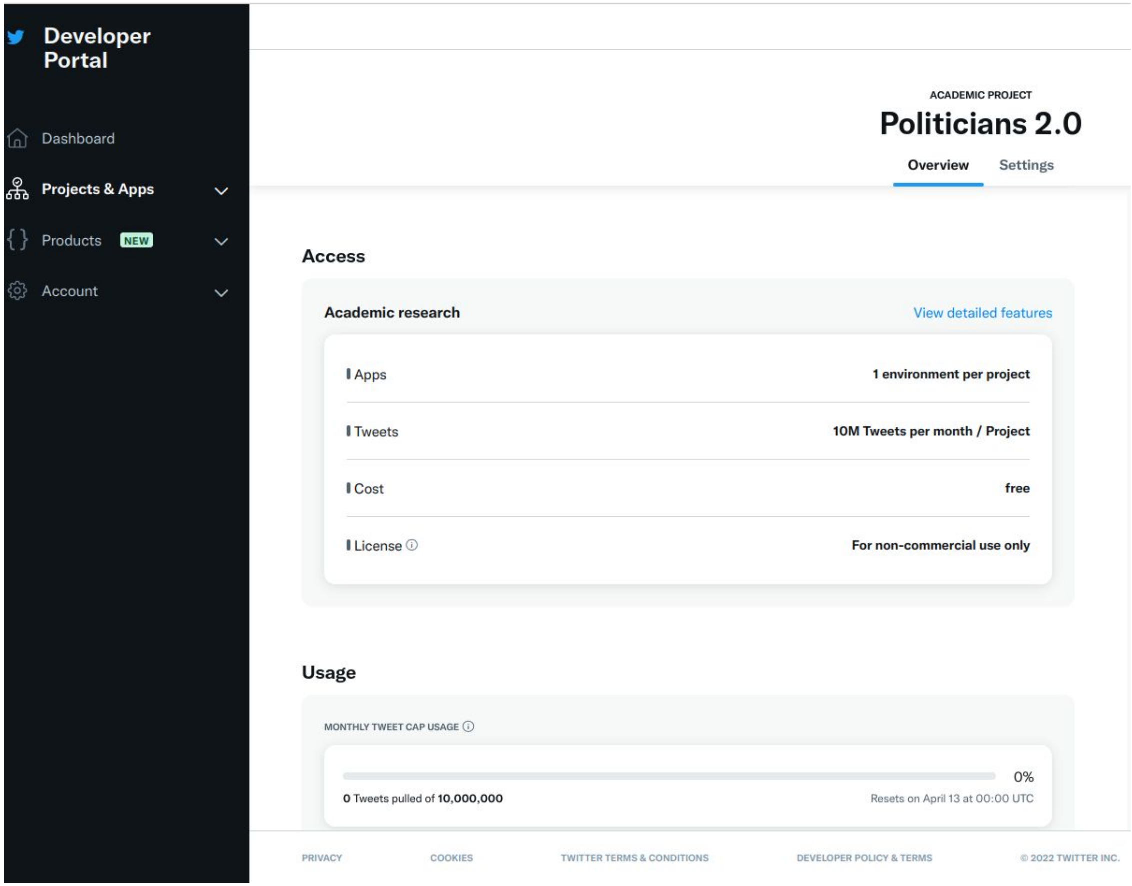
Most APIs require registration

See the now ~~over~~often-used Twitter API for academics

1. Register for a project
https://apps.twitter.com
2. Wait for approval
3. Get your credentials
4. Make requests

See Chris Bail's detailed page on APIs

See academicTwitteR webpage



ACADEMIC PROJECT
Politicians 2.0

Access

Academic research

Apps	1 environment per project
Tweets	10M Tweets per month / Project
Cost	free
License ⓘ	For non-commercial use only

Usage

MONTHLY TWEET CAP USAGE ⓘ

0%

0 Tweets pulled of 10,000,000

Resets on April 13 at 00:00 UTC

APIs

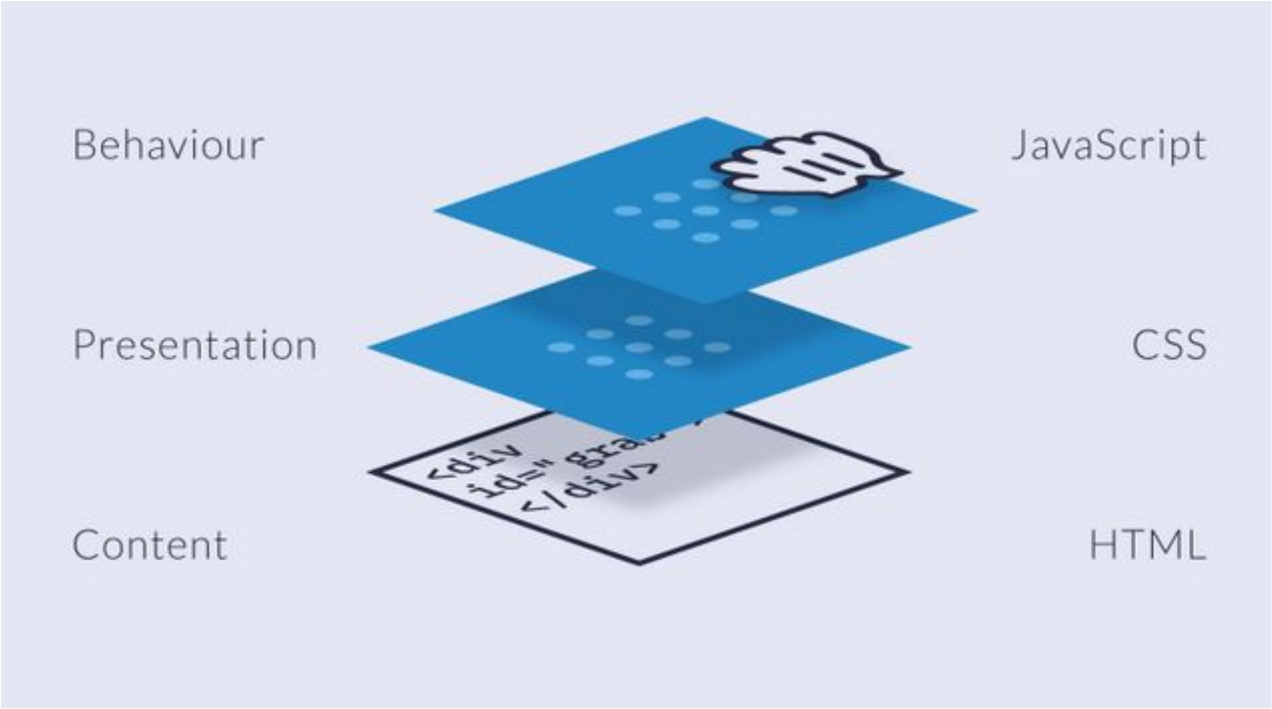
Back to legal and deontological matters

- If there is an API, use it
- If there is no API, ask yourself: are you doing something illegal?

- Sure, scraping Twitter is legal, and its content is public. But what do you learn from individuals? And how should you protect them?

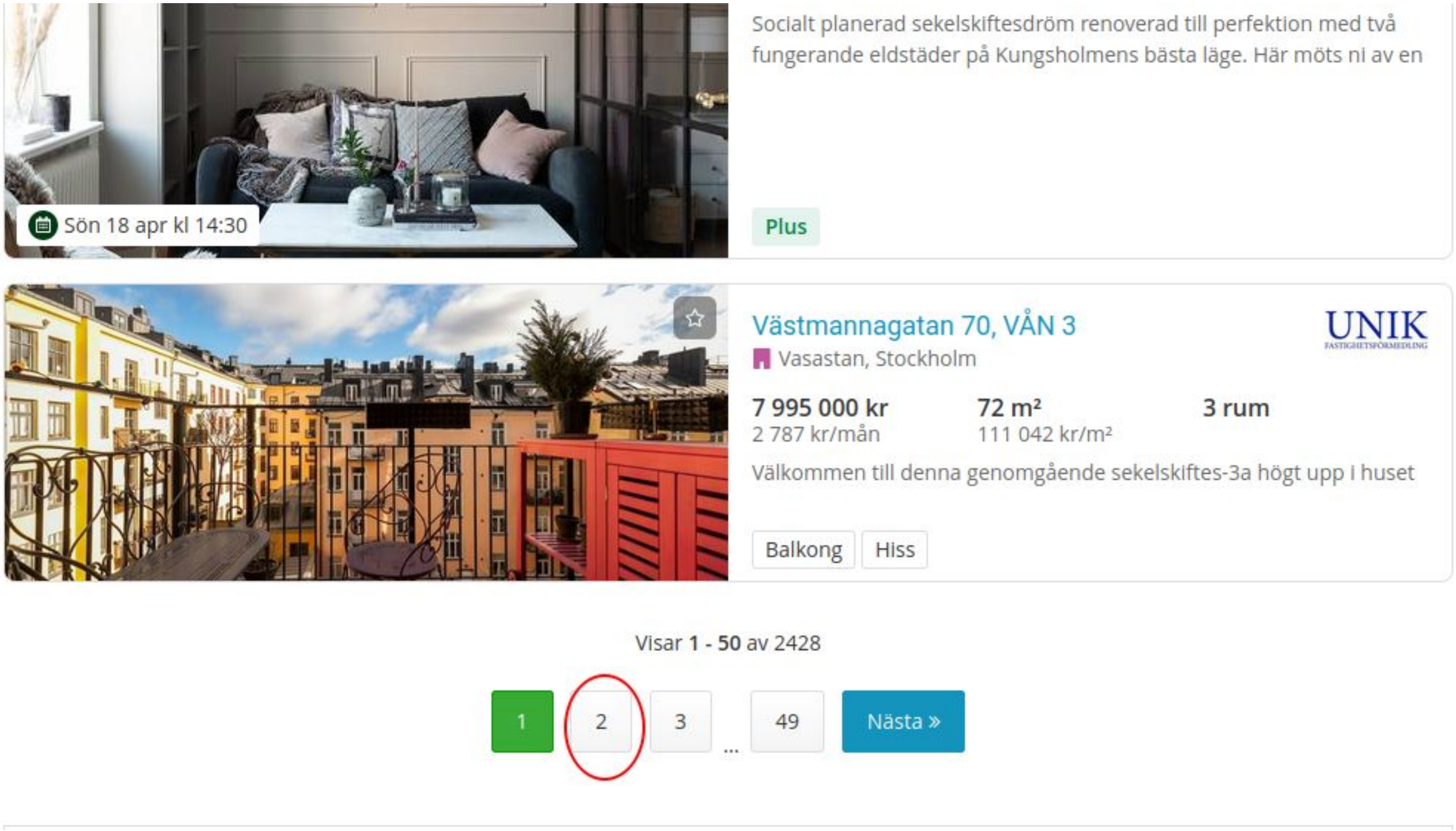
A Quick Note on Headless Browsers

A growing trend in the web industry is to have websites that respond to your behavior (scrolling, clicking, etc).



A Quick Note on Headless Browsers

For behavior-based websites



Socialt planerad sekelskiftesdröm renoverad till perfektion med två fungerande eldstäder på Kungsholmens bästa läge. Här möts ni av en

Sön 18 apr kl 14:30

Plus

Västmannagatan 70, VÅN 3

Vasastan, Stockholm

7 995 000 kr
2 787 kr/mån

72 m²
111 042 kr/m²

3 rum

Välkommen till denna genomgående sekelskiftes-3a högt upp i huset

Balkong Hiss

Visar **1 - 50** av 2428

A Quick Note on Headless Browsers

To do this, you will need to use Javascript in order to create a “**headless browser**”, i.e. a browser piloted from your command line.

A Quick Note on Headless Browsers

To do this, you will need to use Javascript in order to create a “**headless browser**”, i.e. a browser piloted from your command line.

This is slightly more complex as you need to install other software, but we’ll see an example later.

This is also an easy way to avoid some classic headaches.

A Quick Note on Headless Browsers

In R, this is often done using “Selenium”.
To do so, install “RSelenium”

For me, it worked better installing Docker too (…)

And you will need to type, in the command line, a few lines of code. See this explanation by Chris Bail.

Conclusion

But all you need to know is **read_html()**
(and where to look for)