

Grundlagen: Informationstechnologie in Bibliotheken

Sven Koesling

ETH-Bibliothek

Herbst 2017

22.12.2017 : Internettechnologien II: von interaktiven Webseiten zu WebApps in der Cloud

- Der Einsatz von JavaScript Frameworks anhand von Primos neuem UI

Beispiele für Elemente im html-Dokument

- Überschriften : `<h1>Überschrift</h1>`
- Absätze : `<p>Absatz</p>`
- Tabellen : `<table><tr><td>Zelle</td></tr></table>`

die Formatierung im alten OPAC



[Benutzungskonto](#) | [Einstellungen](#) | [Geführte Suche](#) | [Hilfe](#) | [Abmelden](#)

[Andere Kataloge](#) [E](#) | [F](#)

[Ergebnisliste](#) | [Suchverlauf](#) | [Liste](#) | [Korb](#)

[Suche](#) | [Erweiterte Suche](#) | [Expertensuche](#) | [Blättern >Zeitschriften >Signaturen >Sachbegriffe >ISBN](#)

Stichwortsuche:

Stichwörter aus allen Feldern

Titel enthält Stichwörter...

Autor/Körperschaft enthält Stichwörter...

Serie/Reihe enthält Stichwörter...

OK

Beispiele:

frisch andorra
frisch and andorra
anthropolog? china
(herz or cardio?) and therap?

Blättern in einer Liste:

Autor/Körperschaft beginnt mit...

Titel beginnt mit...

Zeitschriftentitel beginnt mit...

Sachbegriff NEBIS (D,E,F) beginnt mit...

OK

Beispiele:

frisch m
frisch max
schweizerische afrika gesellschaft
conference on acid rain

Tipp: bei Autoren Nachname zuerst

[Informationen zum Katalog](#)
[Ferien- und Inventurschliessungen](#)

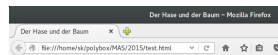
[Feedback/Anregungen](#) - [FAQ](#)



die Formatierung im alten OPAC (Quelltext)

```
<table style="margin-top:0.4em" width="100%" border="0" cellspacing="0"
  cellpadding="0">
  <tr class="topbar">
    <td valign="middle" width="120" rowspan="3">
      <a href="...">
        
      </a>
    </td>
    <td valign="middle" align="left" nowrap>
      <a href="..." class="konto">Benutzungskonto</a> |
      <a href="..." class="black">Einstellungen</a> |
      <a href="..." class="black">Gef&uuml;hrte&nbsp;Suche</a> |
      ...
    </td>
    <td valign="middle" align="right" nowrap>
      <a href="..." class="black">Andere Kataloge</a>
      ...
    </td>
  </tr>
</table>
```

```
<html>
  <head>
    <meta charset="utf-8" />
    <title>Der Hase und der Baum</title>
  </head>
  <body>
    <h1>Der Hase und der Baum</h1>
    <h2>Kapitel 1: Der Hase</h2>
    <p>Meister Lampe hoppelt über ein Feld.</p>
    <h2>Kapitel2: In der Werkstatt</h2>
    <p>Herr K. bestellt eine Knautschzone.</p>
  </body>
</html>
```



Der Hase und der Baum

Kapitel 1: Der Hase

Meister Lampe hoppelt über ein Feld.

Kapitel2: In der Werkstatt

Herr K. bestellt eine Knautschzone.

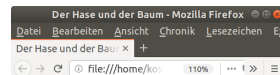
Die Anweisung

style="color: grey;"

lässt das entsprechende Element in grau erscheinen.

erste eigene Formatierung

```
<html>
  <head>
    <meta charset="utf-8" />
    <title>Der Hase und der Baum</title>
  </head>
  <body>
    <h1 style="color: grey;">Der Hase und der
      Baum</h1>
    <h2>Kapitel 1: Der Hase</h2>
    <p>Meister Lampe hoppelt über ein Feld.</p>
    <h2>Kapitel2: In der Werkstatt</h2>
    <p>Herr K. bestellt eine Knautschzone.</p>
  </body>
</html>
```



Der Hase und der Baum

Kapitel 1: Der Hase

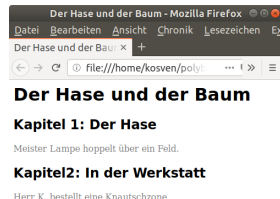
Meister Lampe hoppelt über ein Feld.

Kapitel2: In der Werkstatt

Herr K. bestellt eine Knautschzone.

Formatierung im head

```
<html>
  <head>
    <meta charset="utf-8" />
    <title>Der Hase und der Baum</title>
    <style>
      h1,h2 { font-family: sans-serif; }
      p { color: gray; }
    </style>
  </head>
  <body>
    <h1>Der Hase und der Baum</h1>
    <h2>Kapitel 1: Der Hase</h2>
    <p>Meister Lampe hoppelt über ein Feld.</p>
    <h2>Kapitel2: In der Werkstatt</h2>
    <p>Herr K. bestellt eine Knautschzone.</p>
  </body>
</html>
```



```
<head>  
  <link rel="stylesheet" media="all" href="/assets/application-  
    e77e74d6189491bd233e87f81af0db0315d8ed4a4e7b0070d13b6180200  
    cela7.css">  
</head>
```

Auszug aus dem css von libraries.ch

```
a, label, .btn {  
    font-size: 1em  
}  
  
a {  
    color: #20758c  
}  
...  
body {  
    font-size: 1em  
    overflow-y: scroll  
}  
...  
footer h3 {  
    font-size: 1.2em  
    margin-top: 0  
    margin-bottom: 0  
    height: 1.4em  
}
```

id vergeben

```
<table>
  <tr>
    <th>Kapitel</th><th>Titel</th>
  </tr>
  <tr>
    <td>1</td><td>Der Hase</td>
  </tr>
  <tr id="kapitel2">
    <td>2</td><td>In der Werkstatt</td>
  </tr>
</table>
```

Element mit JavaScript ausblenden

Der JavaScript-Befehl

```
document.getElementById("kapitel2").style.display = 'none';
```

blendet das Element mit der id „kapitel2“ aus.

Inhaltsverzeichnis

Kapitel	Titel
1	Der Hase
2	In der Werkstatt

Inspekt Konso Debugg {} Stilbearbeit Laufzeitana Speich Netzwerkanal Speich Adblock F

+ HTML durchsuchen

```
<html>
  <head>
  </head>
  <body>
    <h1>Inhaltsverzeichnis</h1>
    <table>
    </table>
  </body>
</html>
```

Regeln Berechnet Layout Animationen Schrif

Stile filtern ☐ Browser-Stile

Box-Modell

Außenabstand 8
Rand 0
Innenabstand 0
931x132.433
















931x132.433 static



Keine CSS-Eigenschaften gefunden.

html body

Inhaltsverzeichnis

Kapitel	Titel
1	Der Hase
2	In der Werkstatt

 Inspekt  **Konso**  Debugg  {} Stilbearbeit  @ Laufzeitana  Speich  Netzwerkana  Speich  Adblock P      

  Ausgabe filtern ☐ Nicht leeren

```
>> document.getElementById("kapitel2").style.display = 'none';
```

Element mit script ausblenden

```
<tr>
  <td colspan="2"><a href="#" onclick="verstecken('kapitel2'); return
    false;">Kapitel 2 verstecken</a></td>
</tr>
<tr id="kapitel2" style="display: table-row;">
  <td>2</td><td>In der Werkstatt</td>
</tr>
</table>

<script type="text/javascript">
  function verstecken(id) {
    document.getElementById(id).style.display = 'none';
  }
</script>
```

Und so sieht es im Browser aus:

Element mit script ein- und ausblenden

```
<tr>
  <td colspan="2"><a href="#" onclick="einausblenden('kapitel2');
    return false;">zeige</a></td>
</tr>
<tr id="kapitel2" style="display: none;">
  <td>2</td><td>In der Werkstatt</td>
</tr>
</table>

<script type="text/javascript">
  function einausblenden(id) {
    var meinElement = document.getElementById(id);
    if (meinElement.style.display === "none") {
      meinElement.style.display = "table-row";
    } else {
      meinElement.style.display = "none";
    }
  }
</script>
```

Und so sieht es im Browser aus:

zwei verbreitete JavaScript Libraries

- jquery / jquery UI
- bootstrap

zwei verbreitete JavaScript Frameworks

- AngularJS / Angular 2
- ReactJS

Möglichkeiten, JS einzubinden:

- eine Version direkt auf dem eigenen Server zur Verfügung stellen
- eine bestimmte Version als Link direkt einbinden
- eine Minimalversion einbinden und benötigte Funktionen dynamisch nachladen (problematisch, wenn das Gerät nicht dauerhaft online ist)

die Einbindung von jquery

```
<html>
  <head>
    <meta charset="utf-8" />
    <script src="https://code.jquery.com/jquery-latest.js"></script>
    ...
```

Das Ansteuern der Elemente ist viel einfacher

Der Befehl

```
$("#kapitel2").toggle();
```

blendet das Element mit der id „kapitel2“ ein und aus.

→ Test im Browser

ein- / ausblenden mit jquery

```
...
<tr>
  <td colspan="2"><a href="#" id="schalter">zeige</a></td>
</tr>
<tr id="kapitel2" style="display: none;">
  <td>2</td><td>In der Werkstatt</td>
</tr>
</table>

<script type="text/javascript">
  $(document).ready(function(){
    $("#schalter").click(function(){
      $("#kapitel2").toggle();
    });
  });
</script>
...
```

Userinterface mit jquery UI

Mit der Bibliothek „jquery UI“ lassen sich schnell einheitliche Oberflächen bauen. Hier ein Button:

```
<head>
  <meta charset="utf-8" />
  <script src="https://code.jquery.com/jquery-latest.js"></script>
  <link rel="stylesheet" href="https://ajax.googleapis.com/ajax/libs/
    jqueryui/1.12.1/themes/smoothness/jquery-ui.css">
  <script src="https://ajax.googleapis.com/ajax/libs/jqueryui/1.12.1/
    jquery-ui.min.js"></script>
...
<body>
...
  <button>zeige Kapitel 2</button>
```




















bootstrap statt Tabellenlayout

```
<div class="row">  
  <div class="col-md-3">...</div>  
  <div class="col-md-1">...</div>  
  <div class="col-md-3">...</div>  
  <div class="col-md-1">...</div>  
  <div class="col-md-3">...</div>  
  <div class="col-md-1">...</div>  
</div>
```

auf einem normalen Monitor


Sie sind als Administrator eingeloggt


Suchen


Nachname, Vorname	Geb.	E-Mail	eingeschr.	Kommentar
Test, Lars (Kein Entitlement)	27.06.1978	 dalars27@gmail.com	5.9.2017	
Lüttringhaus, Rahel (Kein Entitlement)	08.03.1982	 rahel.luettringhaus@library.ethz.ch	4.9.2017	
Garkelner, Niemand (Kein Entitlement)	01.01.2000	 mcvsvenster@googlemail.com	10.1.2017	
Müller, Ursulatest (Kein Entitlement)	08.03.1969	 ursula.test@gmail.com	8.12.2016	testkonto - wird nicht freigeschaltet. 
Hännli, Heidi (Kurz)	02.08.1957	 heidi.haenni@library.ethz.ch	7.12.2016	
Spring, Rebekka (Kurz)	03.08.1981	 rebekka.spring@library.ethz.ch	10.11.2016	
testname, testuser (Kein Entitlement)	1.1.2000	 oc2@2consult.ch	26.7.2016	
testuser, oneconsult (Kein Entitlement)	1.1.2000	 oc1@2consult.ch	26.7.2016	
testuser, oneconsult (Kein Entitlement)	1.1.2000	 oc1@2consult.ch	26.7.2016	Ein Oneconsult-Test 

auf einem mobilen Endgerät

libraries.ch


 Sven Koesling




Sie sind als Administrator eingeloggt 

Test, Lars (Kein Entitlement)

27.06.1978


 dalars27@gmail.com

5.9.2017




Lüttringhaus, Rahel (Kein Entitlement)

08.03.1982


 rahel.luettringhaus@library.ethz.ch

4.9.2017



Garkeiner, Niemand (Kein Entitlement)

01.01.2000

 mcvsvenster@googlemail.com

der Einsatz von JavaScript ermöglicht neue Services

- SaaS

der Einsatz von JavaScript ermöglicht neue Services

- SaaS
- IaaS

der Einsatz von JavaScript ermöglicht neue Services

- SaaS
- IaaS
- PaaS

Beispiel für SaaS: iCloud

Software as a Service

The screenshot shows a web browser window with the address bar displaying the URL: https://www.icloud.com/pages/0004dXitlasZaHmfBe0bVcGvg#Fortschrittsbericht_MDM_August_2015. The browser's address bar also shows "Apple Inc. (US)". The page title is "iCloud Pages". The document is titled "Fortschrittsbericht August 2015" and is from the "ETH-Bibliothek Team Metadatenmanagement".

The document content includes a table with the following data:

Sitzungsnummer	16
Datum	27.08.2015, 14:00 - 15:15 Uhr, HG H 59.2
Anwesende	Judith Bissegger, Barbara Wittwer, Marianne Wolff
Entschuldigt	Erica Pfister, Madeleine Völlmin

Below the table, the section "Ergebnisse" (Results) is followed by a list of tasks:

- Nachbearbeitung E-Book Anreicherung: Analyse der angereicherten Daten und notwendige Bereinigungen, sowie Erstellung von SYS-Nr.-Listen für MDM
- Inhalte der Felder 008/906/907 genauer analysiert und Gruppen von fehlerhaften Codes korrigiert
- Prüf-Routinen und Auswahllisten von in Aleph hinterlegten Worten (CtrlP8-Listen) auf EBI05 in Aleph Test eingerichtet und geprüft. Arbeit wird noch fortgesetzt.
- Safari E-Books (2. Einspielung mit 7000 Titeln) analysiert und maschinell aufgearbeitet. Einzelne Aufnahmen wurden manuell angepasst und mit Verknüpfungen zu Serien und Normeinträgen ergänzt. Die Aufnahmen konnten damit auf ein ausreichendes Level gebracht werden, dass sie im Katalog keine weitere Aufarbeitung durch MDM benötigen. Der Code-Zusatz -MDM wurde daher gelöscht.
- E-Book Aufnahmen wurden im Hinblick auf RDA analysiert und ein Kriterium wurde gesucht, anhand dessen eine E-Book Aufnahme nach RDA erkannt werden kann. Dies ist nötig für künftige E-Book Einspielungen und den Abgleich mit Aleph.

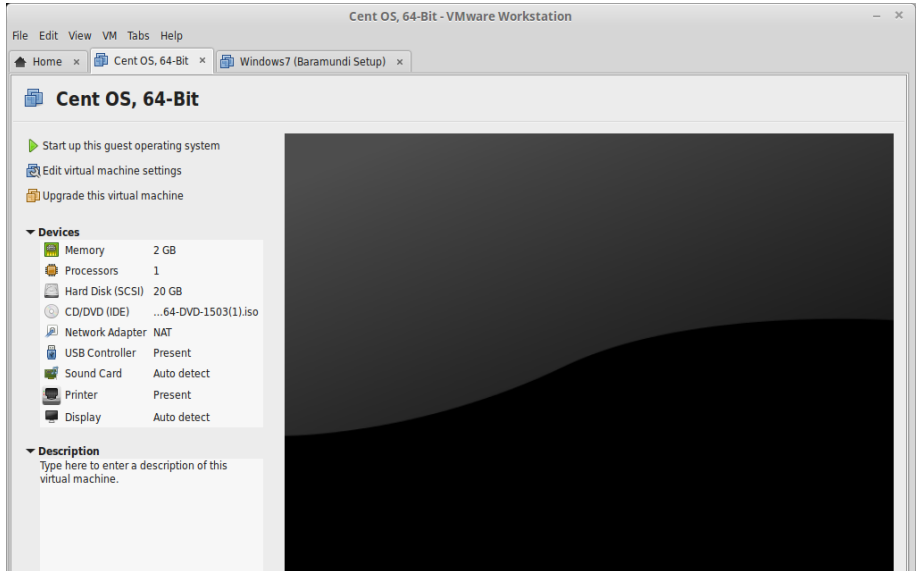
The section "Erkenntnisse" (Conclusions) follows, with a list of tasks:

- Arbeiten, die mit der Einführung von RDA anfallen, eignen sich hervorragend für das MDM-Team. Das bisher erlangte Wissen ist notwendig für bestimmte Arbeiten, wie z.B. Datenanalysen, Angleichungsroutinen, Expertenabfrage, sowie den allg. Umgang mit GUI. Ebenso ist das Datenflussdiagramm auch für die Einführung der RDA ein Gewinn, da es aufzeigt, welche Systeme Daten aus Aleph beziehen und daher auch von der Erstellung auf RDA betroffen sind.

The right sidebar shows formatting options for the text, including "Absatzstil" (Normal), "Schrift" (Helvetica), "Ausrichtung" (Left), "Zellenabstand" (1), "Absatzabstand" (0 pt), "Listen & Zeichen" (Ohne), and "Einrücken" (0).

Beispiel für IaaS

Infrastructure as a Service



Beispiel für Paas

Platform as a Service

Overview

[Refresh](#)

Health

Green

[Causes](#)

Running Version

app-6eed-151222_104543

[Upload and Deploy](#)

Configuration

64bit Amazon Linux 2015.09
v2.0.4 running Ruby 2.2
(Passenger Standalone)

[Change](#)

Recent Events

[Show All](#)

Time	Type	Details
2016-01-03 05:57:41 UTC+0100	INFO	Environment health has transitioned from RED to GREEN
2016-01-03 05:56:50 UTC+0100	WARN	Environment health has transitioned from GREEN to RED

- Wir prüfen nach einem Update, ob sich die Suchalgorithmen verändert haben, indem wir die Trefferzahl zwischen den beiden Systemen mit identischem Datenstand vergleichen.

- Wir prüfen nach einem Update, ob sich die Suchalgorithmen verändert haben, indem wir die Trefferzahl zwischen den beiden Systemen mit identischem Datenstand vergleichen.
- Der Date-Slider hat in der Vergangenheit Probleme gemacht. Wir checken, ob er plausible Ranges liefert.

- Wir prüfen nach einem Update, ob sich die Suchalgorithmen verändert haben, indem wir die Trefferzahl zwischen den beiden Systemen mit identischem Datenstand vergleichen.
- Der Date–Slider hat in der Vergangenheit Probleme gemacht. Wir checken, ob er plausible Ranges liefert.
- Wir testen, ob nicht–lateinische Schriftzeichen gefunden werden.

Anforderung in Gherkin formuliert

```
Szenario: Eine Suche ergibt auf den beiden Prod Systemen eine ähnliche  
Anzahl Treffer  
Wenn ich die Seite "http://terza-prod1-fe41.ethz.ch/primο-explore/  
search?vid=DADS&sortby=rank&lang=de_DE" aufrufe,  
Und ich in den Suchschlitz "Wald" eingebe,  
Und die Anzahl der Treffer nehme  
Und dann die Seite "http://terza-prod2-fe41.ethz.ch/primο-explore/  
search?vid=DADS&sortby=rank&lang=de_DE" aufrufe,  
Wenn ich in den Suchschlitz den Suchbegriff "Wald" eingebe,  
Und dort die Anzahl der Treffer nehme  
Dann sollten die Treffermengen ähnlich, d.h. die Abweichung unter  
1%, sein.
```

Der Suchschlitz im alten UI

ETH-Bibliothek - Wissensportal - Mozilla Firefox

ETH-Bibliothek - Wissens x +

terza-test-fe41.ethz.ch/primo_library/libweb/action/ser...

Gast e-Shelf Mein Konto Anmelden

ExLibris Primo

Neue Suche Tags atoz citationlinker Collection Discovery Hilfe Sprache: Deutsch

Bücher, Zeitschriften und mehr Artikel und mehr

Alle Suchbereiche Suchen Erweiterte Suche

Subscribe to Library News feeds

Alle enthält in allen Feldern

Home Blog RSS FAQ Disclaimer Kontakt

Powered by ExLibris Primo Copyright © 2009 Datenschutz und Cookies Accessibility Statement & Disclaimer

Meine Ansicht automatisch aktualisieren

Inspekt Konsole Debugg {} Stilbearbeitung Laufzeitanalyse Speicher Netzwerkkanal Speicher AdBlock P Layout Animationen Schrift

HTML durchsuchen

```
<input id="trng" name="trng" value="" type="hidden">
<!--search_hidden.jspf end-->
<fieldset>
  <legend class="EXLHiddenCue">Primo Search</legend>
  <div class="EXLSearchTabsContainer">
  <div class="EXLSearchFieldRibbon">
    <div class="EXLSearchFieldRibbonFormFields">
      <div class="EXLSearchFieldRibbonFormSearchFor">
        <label class="EXLHide" form="search_field">Search For:</label>
        <input id="search_field" class="" name="vl(freeText0)" value="" ac...
      </div>
    </div>
  </div>
</div>
</div>
</div>
```

Regeln Berechnet Layout Animationen Schrift

Stile filtern Browser-Stile

Box-Modell

Außenabstand		0	
Rand	1		
Innenabstand	93333		
0	1	8.8	309.95x20
		0	1
		0	0

Quellcode des Suchschlitzes (altes UI)

```
<input name="vl(freeText0)" class="" value="" id="search_field"  
accesskey="s" type="text">
```

Der Suchschlitz im neuen UI

The screenshot shows the Primo von Ex Libris search interface in Mozilla Firefox. The browser's address bar displays the URL `http://terza-prod1-fe41.ethz.ch/primo-explore/search?vid=DA`. The search bar is highlighted with a blue border and contains the text "input#searchBar.ng-pristine.ng-empty.ng-valid.ng-valid-required.flex.ng-touched | 625 x 60". Below the search bar, the text "Alles durchsuchen" is visible. To the right of the search bar, there is a magnifying glass icon and the text "ERWEITERTE SUCHE".

The developer console is open, showing the HTML structure of the search bar. The HTML code is as follows:

```
<input id="searchBar" class="ng-pristine ng-empty ng-valid ng-valid-required flex ng-touched" flex="" name="" ng-if="!floatingLabel" autocomplete="off" ng-required="$mdAutocompleteCtrl.isRequired" ng-disabled="$mdAutocompleteCtrl.isDisabled" ng-readonly="$mdAutocompleteCtrl.isReadOnly" ng-model="$mdAutocompleteCtrl.scope.searchText" ng-
```


Quellcode des Suchschlitzes (neues UI)

```
<input flex="" id="searchBar" name="" ng-if="!floatingLabel"
  autocomplete="off" ng-required="$mdAutocompleteCtrl.isRequired" ng-
  disabled="$mdAutocompleteCtrl.isDisabled" ng-readonly="
  $mdAutocompleteCtrl.isReadOnly" ng-model="$mdAutocompleteCtrl.scope
  .searchText" ng-keydown="$mdAutocompleteCtrl.keydown($event)" ng-
  blur="$mdAutocompleteCtrl.blur()" ng-focus="$mdAutocompleteCtrl.
  focus()" placeholder="Alles durchsuchen" aria-owns="ul-0" aria-
  label="Alles durchsuchen" aria-autocomplete="list" role="combobox"
  aria-haspopup="true" aria-activedescendant="" aria-expanded="false"
  class="ng-pristine ng-empty ng-valid ng-valid-required flex ng-
  touched" aria-invalid="false" style="" type="search">
```

Skript, um „Wald“ in den Suchschlitz zu schreiben

```
Wenn(/^ich in den Suchschlitz "[^"]*" eingebe,$/) do |q|  
  fill_in('#searchBar', with: q)  
  find(".button-confirm").send_keys(:enter)  
end
```

Und der Testablauf

```
cucumber features/suche.feature:50
# language: de
Funktionalität: Suche

Grundlage:                                     # features/suche.feature:3
  # Gegeben sei , dass die Seite "http://terza-test-fe43.ethz.ch" aufgerufen ist
  Gegeben sei , dass die Startseite aufgerufen ist # features/step_definitions/suche_steps.rb:1

  Szenario: Slider bietet eine plausible Zeitangabe zur Einschränkung der Trefferliste # features/suche.feature:48
    Wenn ich in den Suchschlitz den Suchbegriff "cucumber" eingebe, # features/step_definitions/
vollansicht_steps.rb:1
    Dann sollte der Slider plausible Anfangs- und Endwerte haben. # features/step_definitions/
suche_steps.rb:196
    **** alt: 1922--> DS-Startdatum: 1920
    **** jung: 2017--> DS-Enddatum: 2017
    **** Normalisiertes Startdatum: 1920 ***** Normalisiertes Enddatum: 2017

1 scenario (1 passed)
3 steps (3 passed)
0m16.334s
```