leipzig school of media

Clientseitige Skriptprogrammierung

Norman Heino / Sebastian Tramp

Überblick

leipzig school of media

- JavaScript
- Document Object Model (DOM)
- Ajax
- jQuery

Vorteile

leipzig school of media

- Entlastung des Servers
- Formularfehler können vor dem Senden bemerkt werden
- Einfacher Dokumentzugriff über DOM

Nachteile

leipzig school of media

- Client potentiell unsicher
- Moglichkeit von XSS-Attacken
- Benutzer kann Skripte kompromittieren

leipzig school of media

JavaScript

Geschichte von JavaScript

leipzig school of media

- 1995 – Netscape Navigator 2.0 mit
- Mocha/LiveScript
- 1995 – Umbenennung in JavaScript
- 1998 – Standardisierung (ECMA-262)
- 2000 – ECMA-262 3rd Edition

JavaScript – Einsatzgebiete

leipzig school of media

- DOM-Skripte, Browserskripte
- Ajax
- Datenbanken (MongoDB)
- Serverprogrammierung (Node.js)
- *Lingua franca des Webs*

JavaScript – Eigenschaften

leipzig school of media

- dynamisch typisiert
- objektbasiert
- keine Klassen
- funktional
- prototypische Vererbung

JavaScript & HTML

leipzig school of media

- Skripte müssen in HTML eingebettet oder eingebunden werden

```
<script type="text/javascript">
    /* JavaScript-Code */
</script>
```

```
<script type="text/javascript" src="/script.js"></script>
```

JavaScript – Kommentare

leipzig school of media

- einzeilig

```
// Dies ist ein einzeiliger Kommentar
```

- mehrzeilig

```
/* Dies ist ein Kommentar, der über
   mehrere Zeilen geht */
```

JavaScript – Beispiele

leipzig school of media

```
// Variablen
var anzahl = 5, text = 'Hallo Welt';

// Funktion
function hallo(name) {
    alert('Hallo, ' + name);
}

// Array
var array = [1, 2, 3, 4, 5, 6];

// Objekt
var objekt = {
    name: 'Norman Heino',
    email: 'heino@informatik.uni-leipzig.de'
}

// Objektzugriff
alert(objekt.name);
alert(objekt['email']);
```

JavaScript – Beispiele (2)

leipzig school of media

```
// Schleife für Arrays
for (var i = 0; i < array.length; i++) {
    alert(Math.pow(2, array[i]));
}

// Schleife für Objekte
for (var key in object) {
    alert(key + ': ' + object[key]);
}

// Schleifen für Boole'sche Bedingung
while (/* Bedingung */) {
    /* Anweisungen */
};
do {
    /* Anweisungen */
} while (/* Bedingung */);
```

JavaScript – Eigenschaften (2)

leipzig school of media

- Variablen sind nicht typisiert (nur Werte)
- Funktionen können in Variablen gespeichert und als Parameter übergeben werden
- Objektattribute können beliebige Werte besitzen (auch Funktionen)

JavaScript + HTML (2)

leipzig school of media

- Verbindung zwischen HTML und JavaScript über *Ereignisse* (events)
- HTML-Elemente können mit Ereignishandlern assoziiert werden

```
<script type="text/javascript">
function sagHallo() {
    alert('Hallo');
}
</script>

<button onclick="sagHallo()">Hallo</button>
```

JavaScript – Sprachobjekte

leipzig school of media

- String – Zeichenketten
- Number – Zahlen (Ganz- und Dezimal-)
- RegExp – Reguläre Ausrücke
- Array – Felder
- Math – Mathematikfunktionen
- Date – Datumsfunktionen

http://de.selfhtml.org/javascript/objekte/index.htm

JavaScript – Beispiele (String)

leipzig school of media

```
var text = 'Dies ist ein Text, den wir gleich durchsuchen werden.';

var position1  = text.search('gleich');
var position2 = text.search('später');

alert(position1); // 27
alert(position2); // -1

alert(text.substr(0, 8));      // 'Dies ist'
alert(text.substring(13, 23)); // 'Text, den'

// Text an Zeichen auftrennen
alert('Text enthält ' + (split(',').length - 1) + ' Kommata.');
```

http://de.selfhtml.org/javascript/objekte/string.htm

JavaScript – Beispiele (Array)

leipzig school of media

```
var array = ['ist', 'ein'];

array.push('Array');   // ['ist', 'ein', 'Array']
array.unshift('Dies'); // ['Dies', 'ist', 'ein', 'Array'];

var join = array.join(' ');
alert(join); // 'Dies ist ein Array'
```

http://de.selfhtml.org/javascript/objekte/array.htm

JavaScript – Beispiele (Math)

leipzig school of media

```
var zahl1 = 1.23456;
var zahl2 = 5;

alert(Math.max(zahl1, zahl2)); // 5
alert(Math.min(zahl1, zahl2)); // 1.23456
alert(Math.round(zahl1));      // 1

alert(Math.random()); // "zufällige" Zahl zwischen 0 und 1
```

http://de.selfhtml.org/javascript/objekte/math.htm

JavaScript – Beispiele (Date)

leipzig school of media

```
var jetzt = new Date();

alert(jetzt.getFullYear()); // 2010
alert(jetzt.getMonth());    // 12
alert(jetzt.getDate());     // 10
alert(jetzt.getDay());      // 5
```

http://de.selfhtml.org/javascript/objekte/date.htm

JavaScript – Browserobjekte

leipzig school of media

- window – Fensterobjekt (globales Objekt)
- document – Document (HTML-Seite)
- forms – Formulare
- plugins – installierte Plugins (z. B. Flash)

leipzig school of media

DOM – Document Object Model

DOM – Geschichte

leipzig school of media

- 1997 – DHTML erlaubt Änderungen an HTML-Dokument (browserspezifisch!)
- 1998 – DOM Level 1
- 2000 – DOM Level 2 (getElementById, events)
- 2004 – DOM Level 3 (XPath)

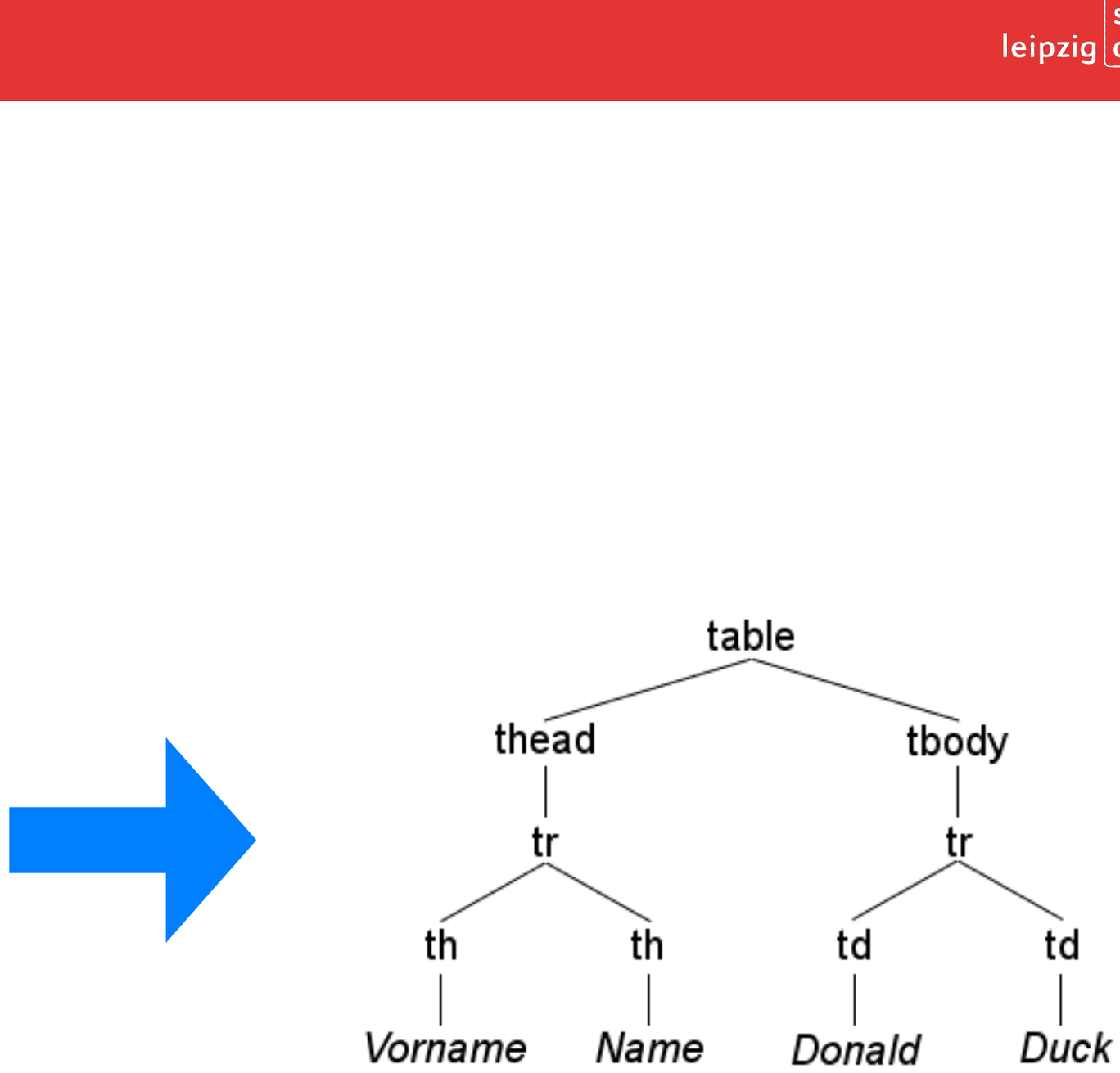
DOM – Einführung

leipzig school of media

- API (application programming interface)
- Zugriff auf Elemente von HTML- und XML-Dokumenten über Baumstruktur
- Definiert Methoden für Elementzugriff

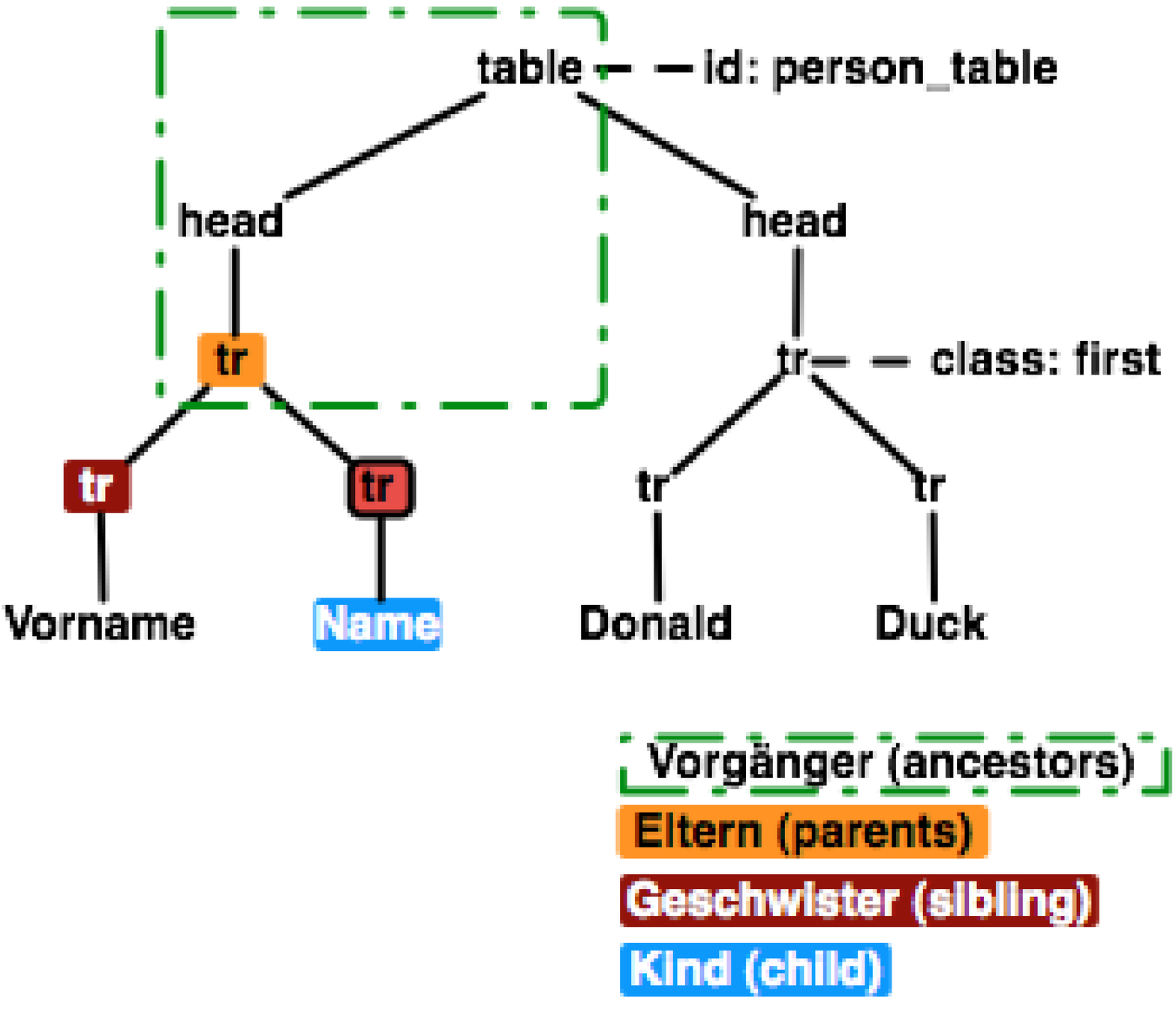
DOM – Beispiel

```
<table id="person_table">
 <thead>
  <tr>
   <th>Vorname</th>
    <th>Name</th>
  </tr>
 </thead>
 <tbody>
  <tr class="first">
   <td>Donald</td>
   <td>Duck</td>
  </tr>
 </tbody>
</table>
```



http://de.wikipedia.org/wiki/Document_Object_Model

DOM – Traversierung



DOM – Knotenarten

leipzig school of media

- Dokument
- Dokumentfragment
- Elementknoten
- Attributknoten
- Textknoten (Inhalt v. Element- oder Attributknoten)

DOM – Nachteile

leipzig school of media

- gesamtes Dokument muss im Speicher gehalten werden
- Zugriff über CSS-Klasse nicht möglich
- Durch Frameworks (jQuery) verbessert

leipzig school of media

Ajax

Ajax – Begriff

leipzig school of media

- ursprünglich: AJAX – Asynchronous JavaScript and XML
- heute: Ajax – asynchrones Nachladen von Daten in versch. Formaten (JSON, HTML, Text, XML, ...)
- Verändern der Webseite (DOM) mit neuen Daten vom Webserver ohne diese neu zu laden

leipzig school of media - Go

www.google.de/#hl=de&expIds=17259,18168,27744,27868,28021&xhr=t&q=leipzig+school+of+media&cp=19&pf=p&sclient=psy&aq=0&a...

Web Bilder Videos Maps News Shopping E-Mail Mehr ▾ norman.heino@googlemail.com | Webprotokoll | Einstellungen ▾ | Abmelden

Google

leipzig school of media Suche

Google Instant ist aktiviert. ▾

leipzig school of **media**
leipzig school of **management**
leipzig school of **media gmbh**
leipzig school of m

Weitere Informationen

Alles
Bilder
Videos
Mehr

Leipzig
Standort ändern

Das Web
Seiten auf Deutsch
Seiten aus Deutschland
Übersetzte fremdsprachige Seiten

Mehr Optionen

Ungefähr 2.060.000 Ergebnisse (0,08 Sekunden) Erweiterte Suche

LEIPZIG SCHOOL OF MEDIA: Über uns - Weiterbildung, Studium ...
Die **Leipzig School of Media** zählt zu den führenden Weiterbildungseinrichtungen im crossmedialen Bereich - mit berufsbegleitenden Masterstudiengängen und ...
www.**leipzigschoolofmedia**.de/ - Im Cache - Ähnliche Seiten

Stellenangebote
Team
Dozenten
Kurse und Schulungen
Kontakt
News und Presse
Master Corporate Publishing

Weitere Ergebnisse von leipzigschoolofmedia.de »

LEIPZIG SCHOOL OF **MEDIA** gGmbH - Crossmedia Weiterbildungen ...
Google Places-Profil

Poetenweg 28
04155 Leipzig
0341 56296701
Tram: G.-Schumann-/Lindenthaler Straße
Wegbeschreibung abrufen - Stimmen diese Angaben?

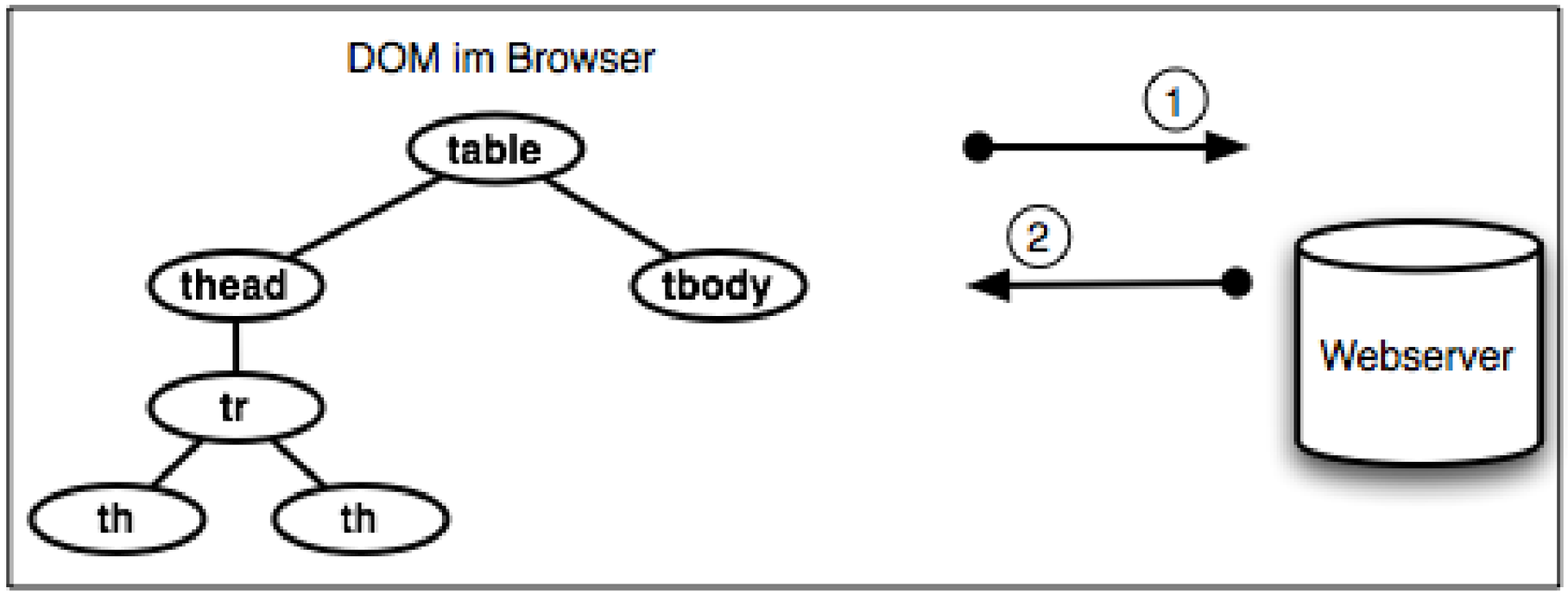
1 Bewertung - Beurteilung schreiben

LEIPZIG SCHOOL OF MEDIA: Stellenangebote - Weiterbildung, Medien ...
Zur Unterstützung unseres Teams suchen wir Studentinnen und Studenten.
www.**leipzigschoolofmedia**.de/.../stellenangebote/ - Im Cache - Ähnliche Seiten

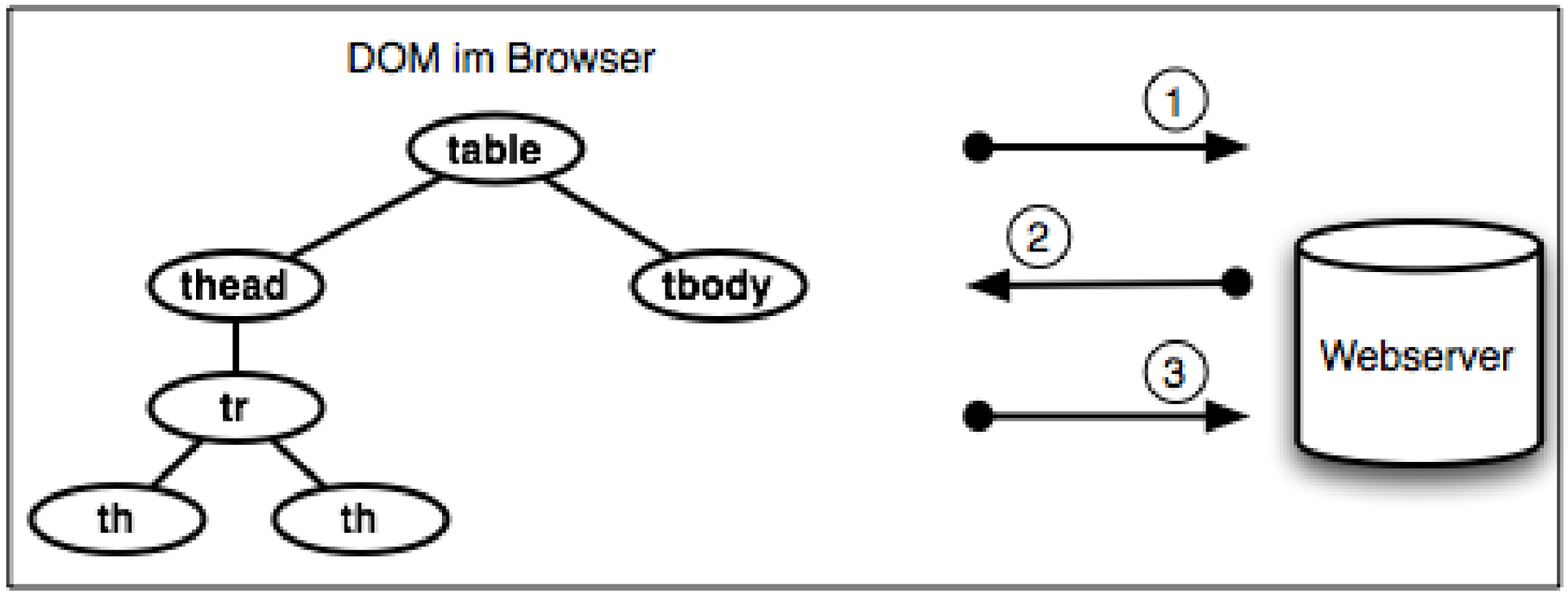
LEIPZIG SCHOOL OF MEDIA: Master Crossmedia Publishing - Crossmedia ...
Das viersemestrige, berufsbegleitende Masterstudium Crossmedia Publishing (M ...
www.**leipzigschoolofmedia**.de/master-cross**media**-publishing/ - Im Cache - Ähnliche Seiten

LEIPZIG SCHOOL OF MEDIA: Master Content and Media Engineering ...
Mit dem viersemestrigen Masterstudium Content and **Media** Engineering wird ...

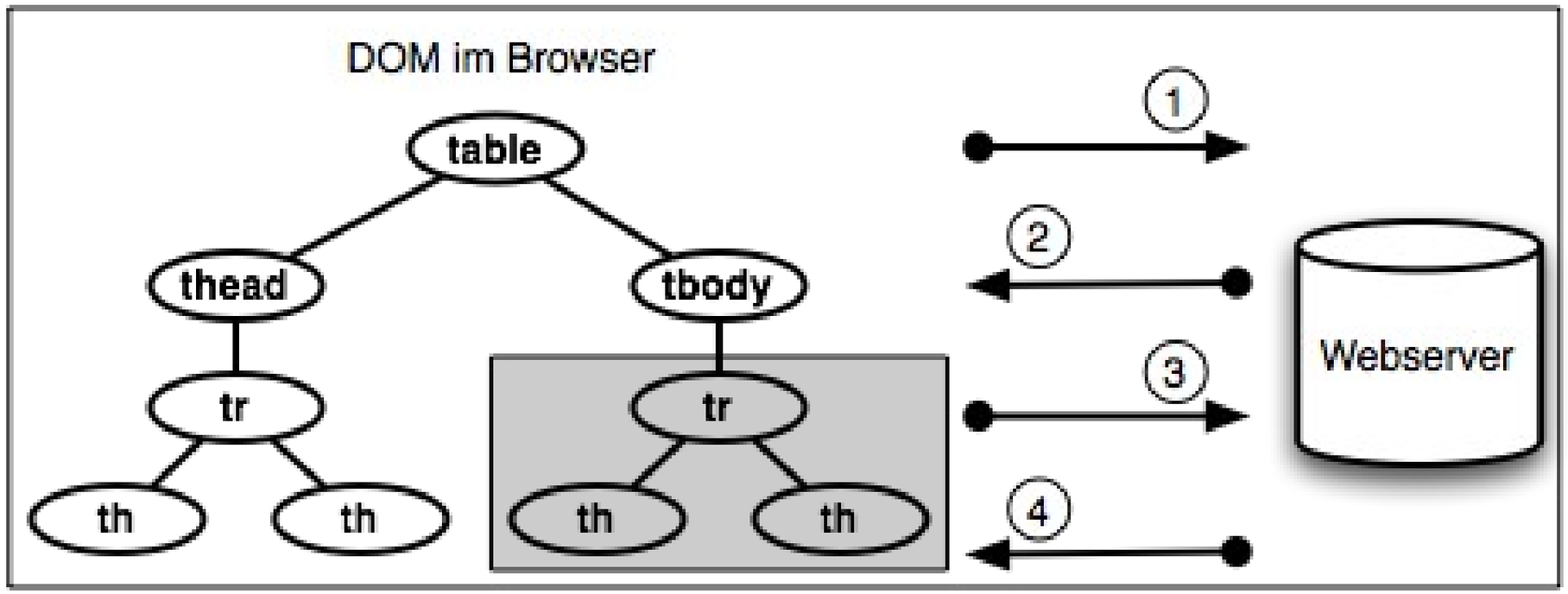
Ajax – Architektur



Ajax – Architektur



Ajax – Architektur



Ajax – Technologien

leipzig school of media

- JavaScript
  - eventgesteuerte Anfragegenerierung
- DOM
  - dynamisches Verändern der Seite
- XMLHTTPRequest (XHR)
  - Browser-Objekt für asynchrone Anfragen

Ajax – Vorteile

leipzig school of media

- Benutzer
  - Webanwendungen nähern sich Desktop an
  - plattformunabhängig
  - schnellere Antwortzeiten
- Entwickler
  - geringerer Ressourcenverbrauch
  - einfachere Modularisierung

Ajax – Nachteile

leipzig school of media

- Benutzer
  - Browser-History funktioniert nicht
  - keine Deep-Links (wird z.Z. Durch fragment identifier umgangen, was aber dreckig ist)
  - Internetverbindung muss bestehen
- Entwickler
  - Unterschiede zwischen Browsern
  - unterschiedliche Programmiersprachen zwischen Client und Server

Ajax – Frameworks

leipzig school of media

- Probleme bei Ajax-Entwicklung:
  - browserspezifische Unterschiede
  - viele Komponenten
- Frameworks:
  - abstrahieren Browserunterschiede
  - vereinfachen Zugriff auf XHR

leipzig school of media

jQuery

jQuery – Funktionalität

leipzig school of media

- Selektor-API (CSS3)
- DOM-Manipulation
- Normalisiertes Event-System
- Hilfsfunktionen
- Ajax
- Effekte, Animationen
- Erweiterbar über Plug-in-System

jQuery – Paradigmata

leipzig school of media

- ein jQuery-Objekt (jQuery oder $)
- Selektoren geben wieder jQuery-Objekt zurück, das alle gefundenen DOM-Elemente "enthält" – *Chaining*
- Kann mit weiteren Selektoren verfeinert werden oder mittels Aktion verändert werden

jQuery – Selektoren

leipzig school of media

- Alles – $('*')
- Tag – $('p')
- ID – $('#eine-id')
- Klasse – $('.eine-klasse')
- Attribute – $('[name]'),
  $('[name="wert"]'),
  $('[name!="wert"]')

jQuery – Traversierung

leipzig school of media

- .parent() – Elternknoten
- .parents() – Vorgänger
- .closest() – erster Vorgänger der auf Selektor passt
- .children() – Kinder
- .next(), .nextAll(), .prev(), .prevAll(), .siblings() – Geschwister

http://api.jquery.com/category/traversing/

jQuery – Manipulation

leipzig school of media

- .addClass(), .removeClass() – CSS-Klassen hinzufügen/entfernen
- .append(), .prepend() – Elemente/HTML davor/danach einfügen
- .height(), .width() – Höhe/Breite auslesen/setzen
- .html(), .text(), .val() – Elemente, Textinhalt, value-Attribut auslesen/setzen

jQuery – Vorgehen

leipzig school of media

- Event: $(document).ready(function () {/* … */}) – ausgeführt sobald Dokument vollständig geladen
- Innerhalb dann:
  - Elemente selektieren
    - für Events registrieren
    - Plug-ins benutzen
    - Events starten

jQuery – Beispiele

leipzig school of media

```
// Selektoren
$('#person_table').parent('div').hide();

// Manipulation
$('#person_table').find('tr.first').append(
    '<a class="my_button">Button</a>');

// CSS-Eigenschaften ändern
$('#my_button').css('font-color', 'red');

// Events
$('#my_button').click(function () {
    // $(this) ist der geklickte Button
    alert('Schaltfläche wurde geklickt');
});

// Inhalt per Ajax laden
$('#update-element').load('http://example.com/getData');
```

```
// Ajax mit Callback
$.get(url, parameter, function (data) {
    /* data verarbeiten */
});

// Ajax mit Callback und JSON-Daten
$.getJSON(url, parameter, function (data) {
    /* data verarbeiten */
});
```

http://api.jquery.com/jQuery.get/