

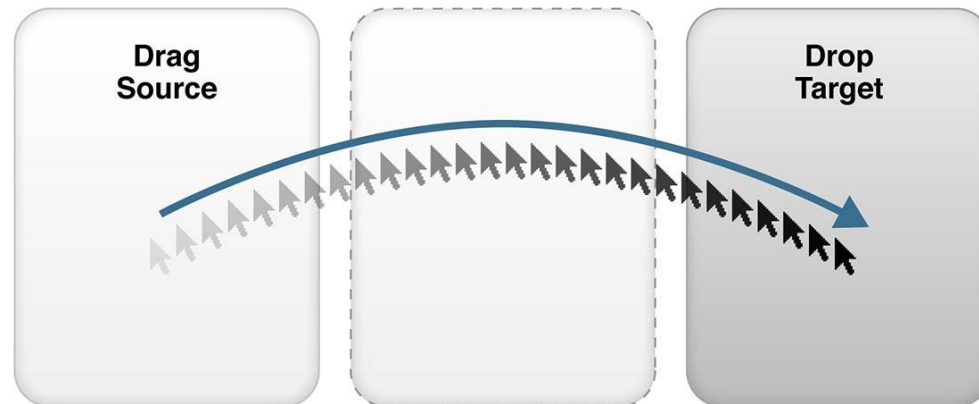
JavaScript – Drag and Drop

Ingo Köster

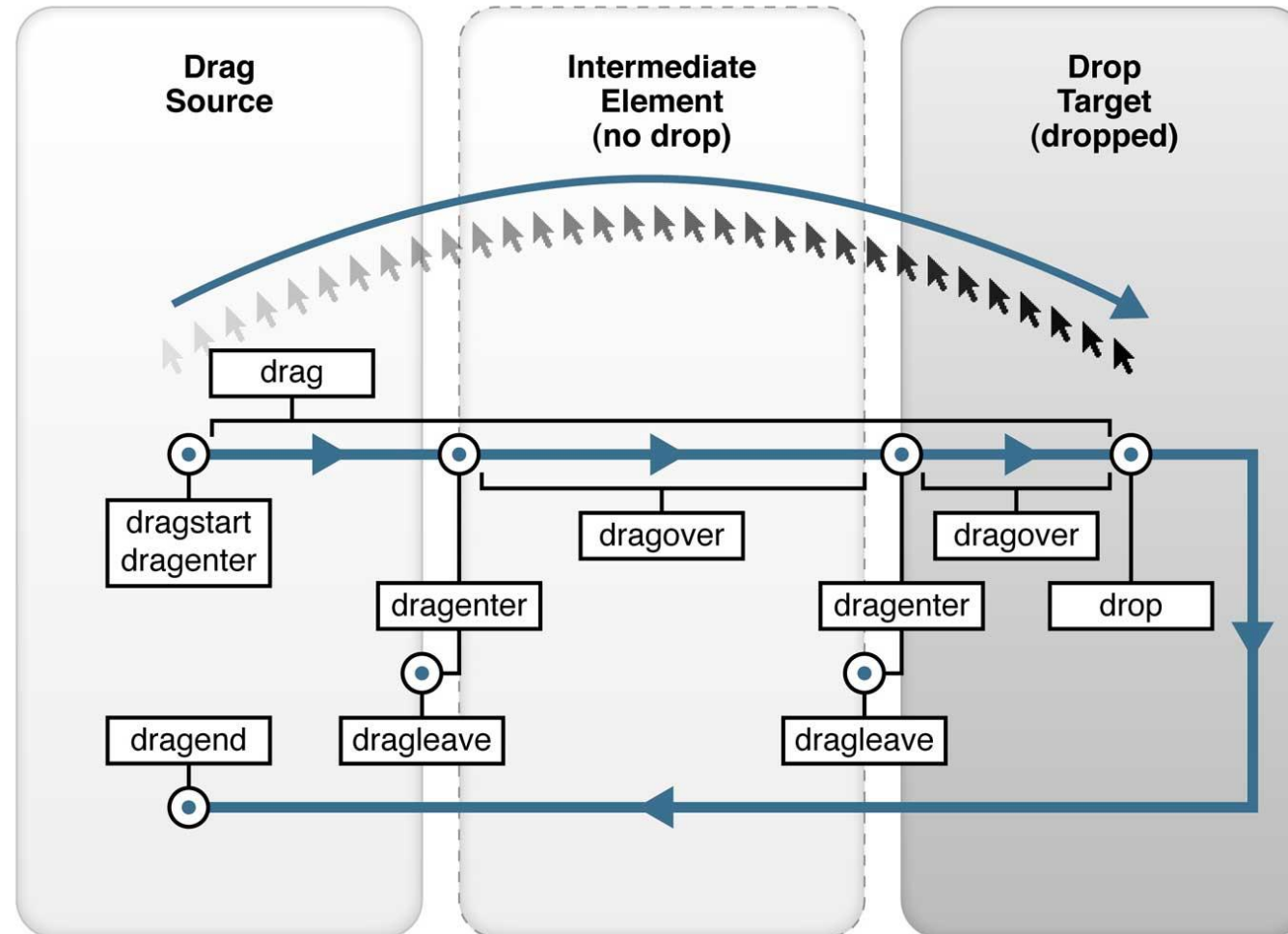
Diplom Informatiker (FH)

Übersicht

- › Mit HTML5 kann mit der Hilfe von Ereignissen (Events) und Attributen festgestellt werden, ob ein Anwender ein Element per Drag-and-Drop bewegen möchte



Drag-and-Drop Event-Modell



Drag-and-Drop Event-Modell

› draggable

Attribut, muss auf true/auto stehen, wenn DnD möglich sein soll

› dragstart

Event, feuert, sobald der Anwender das Objekt zu bewegen beginnt

› drag

› dragenter

› dragleave

Event, feuert permanent, solange der Anwender das Objekt bewegt

› dragover

› drop

Event, wird beim „Fallenlassen“ des Objekts ausgelöst

› dragend

Beispiel – Teil 1

```
<div id="inhalt" draggable="true">Inhalt</div>
```

```
<div id="ziel">Zielbereich für Drag ‘n Drop</div>
```

```
document.getElementById("inhalt").ondragstart = dragStart;
```

```
document.getElementById("ziel").ondragover = handleDrag;
```

```
function dragStart(ev)
{
    ev.dataTransfer.setData("text/plain", ev.target.id);
}
```

```
function handleDrag(ev)
{
    ev.preventDefault();
}
```

Beispiel – Teil 2

```
<div id="inhalt" draggable="true">Inhalt</div>
```

```
<div id="ziel">Zielbereich für Drag ‘n Drop</div>
```

```
document.getElementById("ziel").ondrop = handleDrop;
```

```
function handleDrop(ev)
{
    ev.preventDefault();
    let inhaltId = ev.dataTransfer.getData("text/plain");
    event.target.appendChild(document.getElementById(inhaltId));
}
```

Funktionen

- › `dataTransfer.setData()`
 - › Schreibt Information darüber, welches Objekt vom Anwender mittels Drag and Drop bewegt wird, in das Event-Objekt. Im Beispiel wird die Id des Elements übertragen
- › `dataTransfer.getData()`
 - › Lädt Daten aus dem Drag-Event. Im Beispiel die Id des Elements
- › `event.preventDefault()`
 - › Sorgt dafür, dass der Browser nicht sein "normales" Verhalten zeigt, wenn ein entsprechendes Objekt, z.B. ein Bild, per Drag and Drop übertragen wird

Das DataTransfer-Objekt

Properties	Type
<code>dropEffect</code>	<code>String</code>
<code>effectAllowed</code>	<code>String</code>
<code>files</code>	<code>FileList</code>
<code>mozCursor</code> ⚠	<code>String</code>
<code>mozItemCount</code> ⚠	<code>unsigned long</code>
<code>mozSourceNode</code> ⚠	<code>Node</code>
<code>mozUserCancelled</code>	<code>Boolean</code>
<code>types</code>	<code>StringList</code>

Eigenschaften

Methoden

```
void addElement(in Element element)
void clearData([in String type])
String getData(in String type)
void setData(in String type, in String data)
void setDragImage(in nsIDOMElement image, in long x, in long y)
void mozClearDataAt([in String type, in unsigned long index])
nsIVariant mozGetDataAt(in String type, in unsigned long index)
void mozSetDataAt(in String type, in nsIVariant data, in unsigned long index)
StringList mozTypesAt([in unsigned long index])
```


Typen für den Datentransfer

MIME Type	Wirkung
text/plain	A standard MIME type for unformatted text. We can use it as the most common representation, such as the person's name.
image/png	A standard MIME type for PNG images. Here, it could represent the person's picture in PNG format.
image/jpeg	The standard MIME type for JPEG images. It could be used to transfer the person's picture in that format.
text/x-age	A non-standard MIME type (as indicated by the x- prefix). We could use this format to transfer our own types of information, such as the person's age.

Drag-Effekt festlegen

Value	Description
<code>none</code>	No drag operations are allowed on the element.
<code>copy</code>	The contents of the element should be copied to the destination only.
<code>link</code>	The contents of the element should be shared with the drop destination using a link back to the original.
<code>move</code>	The element should be moved to the destination only.
<code>copyLink</code>	The element can be copied or linked.
<code>copyMove</code>	The element can be copied or moved. This is the default value.
<code>linkMove</code>	The element can be linked or moved.
<code>all</code>	The element can be copied, moved, or linked.

Hinweis jQuery

- › Bei der Verwendung von jQuery ist das Event-Objekt z.B. eines Drag-n-Drop Events ein jQuery-Objekt
- › Mittels der Eigenschaft `originalEvent` auf den Ursprungsevent zugreifen

```
$('myid').on('dragstart', function(e) {  
  e.originalEvent.dataTransfer.setData('text/plain', '42');  
}, false);
```