

1. Grundlagen

HTML5-Werbemittel bestehen wie Websites aus mehreren Elementen, die nicht analog zu Flash in einem File zusammengeführt und komprimiert werden können. Diese sind:

- HTML-Files
- CSS
- Libraries (jQuery, Greensock etc.)
- Bilder
- Videos

2. Dateigröße & Ladezeiten

Damit das Abspielen des Werbemittels nicht unnötig verzögert wird, sollten die einzelnen Elemente des Werbemittels sowohl hinsichtlich ihrer Anzahl als auch Dateigröße so klein wie möglich gehalten werden, um die Serverrequests zu minimieren.

Um das unnötige Laden externer Libraries zu umgehen, raten wir dazu, auf Quellen in der Cloud zu verlinken. Das hat den Vorteil, dass der User die Scripte mit einer sehr großen Wahrscheinlichkeit bereits in seinem Cache liegen hat und diese daher vom Browser nicht erneut heruntergeladen werden müssen.

```
<script src="https://cdnjs.cloudflare.com/ajax/libs/jquery/1.11.1/jquery.min.js"></script>  
<script src="https://cdnjs.cloudflare.com/ajax/libs/gsap/latest/TweenMax.min.js"></script>  
<script src="https://cdnjs.cloudflare.com/ajax/libs/gsap/latest/TimelineMax.min.js"></script>
```

[Vorlagen Download](#)

3. Auslieferung im iFrame

Da HTML5-Werbemittel ähnlich wie Websites aufgebaut sind, kann es durch die verwendeten CSS-Files und Scripte zu Komplikationen mit der eigentlichen Website kommen. Um dieses Problem zu umgehen, liefert unser Ad-Server jedes Werbemittel automatisch in einem iFrame aus.

Um eine reibungslose Funktion der Clicktags zu gewährleisten, müssen alle angelieferten Werbemittel einem vorgegebenen Aufbau entsprechen, der die Clicktag-Übergabe in den iFrame zulässt. Dieser Aufbau wird dann mit großer Sicherheit auch mit anderen gängigen Ad-Servern kompatibel sein.

4. Clicktag

Folgende Codezeilen sind in das HTML5-Werbemittel zur Übergabe der Clicktags zu integrieren, um eine reibungslose GET-Parameter-Übergabe zu gewährleisten.

```
<script>
  (function() {
    var limit = 10,
        prefix = 'clicktag',
        params, clicktags = {};
    function getUriParams() {
      var query_string = {}
      var query = window.location.search.substring(1);
      var parmsArray = query.split('&');
      if(parmsArray.length <= 0) return query_string;
      for(var i = 0; i < parmsArray.length; i++) {
        var pair = parmsArray[i].split('=');
        var val = decodeURIComponent(pair[1]);
        if (val != '' && pair[0] != '') query_string[pair[0]] = val;
      }
      return query_string;
    }
    function setClickEvent() {
      var id, btns = {}, reset = false;
      for (id in clicktags) {
        if (!clicktags.hasOwnProperty(id)) { continue; }
        btns[id] = document.getElementById(id);
        if (!btns[id]) { reset = true; }
      }
      if (reset) {
        if (--limit>0) { setTimeout(setClickEvent, 100); }
        return;
      }
      for (id in btns) {
        if (!btns.hasOwnProperty(id)) { continue; }
        btns[id].addEventListener("click", clickUrl(clicktags[id]));
      }
    }
    function clickUrl(url) {
      return function(ev) {
        window.open(url, '_blank');
        ev.preventDefault();
      }
    }
    params = getUriParams();
    for (var key in params) {
      if (!params.hasOwnProperty(key)) { continue; }
      if (key.substr(0, prefix.length) === prefix) {
        clicktags[key] = params[key];
      }
    }
    if (Object.keys(clicktags).length) { setClickEvent(); }
  })();
</script>
```

[Vorlagen Download](#)

Die Schreibweise für Clicktags lautet: **clicktag**

Die Schreibweise für Multi-Clicktags lautet: **clicktag, clicktag2, clicktag3**

Im Werbemittel selbst spielt es keine Rolle, ob eine Klickfläche aus einem `<a>` oder beispielsweise aus einem `<div>` besteht. Wichtig ist einzig und allein die richtige **ID**.

```
<div id="clicktag"></div>  
<a id="clicktag2"></a>
```

[Vorlagen Download](#)

5. Fallbackimage

Sollte ein Browser spezielle Features oder verwendete Libraries, die im Werbemittel Anwendung finden, nicht unterstützen, so soll ein definiertes Fallback JPG/GIF ausgespielt werden. Bestenfalls wird dieses Fallback im Werbemittel selbst ausgeliefert, aber auch eine Anlieferung als separate Datei ist möglich.

6. Grafikkomprimierung

Grafiken sind hinsichtlich der Dateigröße zu optimieren. Die Verwendung von Image-Optimierern, wie beispielsweise [JPEGmini](#), und der Einsatz von skalierbaren Vektorgrafiken wird empfohlen.

7. Animation

Bei Animationen ist darauf zu achten, dass diese den Client-CPU nicht unnötig belasten. Mehrere parallel laufende Animationen und überlappende transparente Grafiken sind zu vermeiden. Der Einsatz von CSS3- oder Javascript-Animationen ist mit Rücksicht auf die CPU- und GPU-Auslastung zu wählen.

8. Anlieferung

Die Anlieferung des HTML5-Werbemittels erfolgt als **Zip-Datei**, die alle Elemente des Werbemittels enthält – ausgenommen sind extern geladene Libraries, Videos oder Fonts. **Das Werbemittel muss unbedingt ohne Ordner gezippt werden**, sodass das HTML-File auf der ersten Ebene liegt. Das Werbemittel selbst darf Unterordner, in denen die jeweiligen Files liegen, enthalten.

Die Vorlaufzeit für die Einbindung und das Testen beträgt 5 Werktage.