

# Internet Explorer 8/9: Leitfaden für Kompatibilitätstests

## Entwurf und Test Ihrer Website für Internet Explorer 8 und Internet Explorer 9

Hans Krijger  
Microsoft Corporation

## Einleitung

Durch das Fehlen von formalen Standards und dem daraus folgenden nicht standardkonformen Verhalten von bestimmten Browsern, müssen die Entwickler von Websites oft verschiedene Browser berücksichtigen. In vielen Fällen werden im HTML-Code einer Webseite Name und Version des Browsers abgefragt. Dann wird die Webseite entsprechend angepasst. Die Entwicklung für verschiedene Browser, die übergreifende Kompatibilität und die parallele Entwicklung mehrerer Websites machen die Arbeit der Entwickler oft sehr mühsam. Mit Internet Explorer 8 zielen wir darauf ab, den Zeit- und Arbeitsaufwand für diese Aufgaben zu verringern und die Unterstützung echter plattformübergreifender Standards für mehr Browserkompatibilität und Compliance zu verbessern.

Wenn die Entwickler den entsprechenden Best-Practices folgen, dann sollte sich eine Website in Internet Explorer 8 genauso verhalten und genauso angezeigt werden wie in Internet Explorer 7. Normalerweise nutzt Internet Explorer 8/9 einen Standardmodus, der sich strenger an Vorgaben der aktuellsten Funktionen der CSS 2.1- und HTML4, sowie CSS3 und HTML5-Spezifikationen hält. Eine Entwicklung, die dies berücksichtigt, durch eine bessere Einhaltung der Standards und somit für eine einheitliche Funktion und Anzeige in verschiedenen Browsern für eine einheitliche Codebasis und schnellere und einfachere Entwicklungsprozesse. Internet Explorer 8/9 bietet außerdem einen Mechanismus zur Emulation des Standardverhaltens von Internet Explorer 7.

In diesem Dokument geben wir Ihnen einen kurzen Überblick der für das Testen von Websites wichtigen Änderungen in Internet Explorer 8 und 9. Wir stellen diese Änderungen hierbei in Beziehung zu tatsächlich in freier Wildbahn auftretenden Problemen und zeigen so die besten Ansätze für das Debuggen von Code und zur Reduzierung von Problemen auf. Des Weiteren erhalten Sie Informationen zum Testen des Dokumentmodus Internet Explorer 8 bzw. 9-Standards.

## Inhalt

Die Entwickler-Trickkiste .....	2
Kommt Ihnen dieser Code bekannt vor? .....	4
Graben Sie tiefer .....	5
Einkreisen des Problems.....	8

## Die Entwickler-Trickkiste

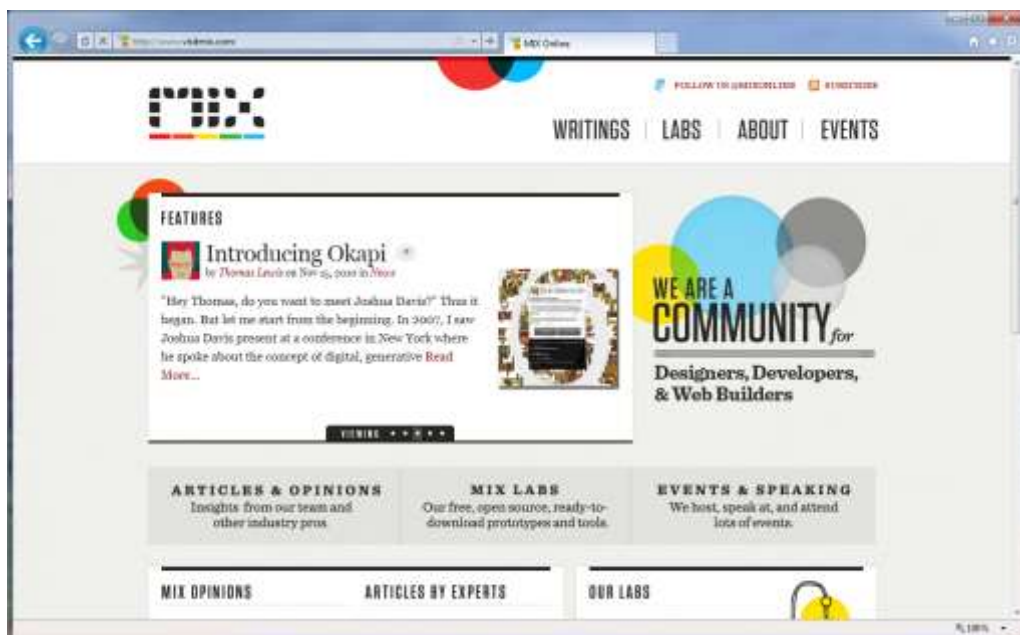
In diesem Abschnitt erfahren Sie mehr zu den für das Testen und Prüfen der Kompatibilität relevanten Funktionen von Internet Explorer 8 und 9:

- Konfigurierbarer Dokumentmodus: Quirksmodus, Internet Explorer 7-Standards, Internet Explorer 8 und Internet Explorer 9-Standards
- Leistungsstarke Debugging-Tools für HTML, CSS, Skripte und seit Internet Explorer 9 auch für Netzwerkprobleme
- Opt-In-Möglichkeit des Internet Explorer 8/9-Modus zur Entwicklung für unterschiedliche Zielgruppen
- Auswahlmöglichkeit zwischen standardmäßigem Internet Explorer 9, Internet Explorer 8-Modus und Internet Explorer 7-**Kompatibilitätsansicht**

Der Dokumentmodus kann über die Internet Explorer 8/9-Entwicklertools konfiguriert werden. Drücken Sie F12, und wechseln Sie über das Menü „Dokumentmodus:“ zum gewünschten Modus. Auf diese Art und Weise können Sie ganz einfach testen, ob sich Ihre Website unter anderen Browserversionen korrekt verhält. Bisher im Modus Internet Explorer 7-Standards dargestellte Websites werden nun standardmäßig im Modus Internet Explorer 8 bzw. 9-Standards angezeigt (der Zusatz „Seitenstandard“ macht dies deutlich). Ist die Website korrekt entworfen worden, so wird sie exakt so wie in Internet Explorer 7 dargestellt.

Statt nur den Quellcode anzuzeigen, verschaffen Sie sich mit den Entwicklertools mehr Einblick in die interne Darstellung einer Website. Sie können so leichter feststellen, warum eine Website nicht wie erwartet dargestellt wird oder sich anders verhält, als sie sollte. Im HTML-Bereich können Sie im DOM-Baum navigieren und diesen anzeigen. Sie können Attribute direkt verändert und so schnell mögliche Lösungen für Probleme finden und testen. Außerdem können Sie CSS und Skripte separat anzeigen. Eine der hilfreichsten Funktionen der Skriptansicht ist die Möglichkeit, Haltepunkte in den Code einzufügen und diesen so schrittweise auszuführen und Variablen sowie den Call-Stack (Aufrufliste) zu überwachen und zu verändern.

Trotz erheblicher Bemühungen, Internet Explorer 8 und 9 rückwärtskompatibel zu gestalten werden bestimmte Entwicklungspraktiken zur Überprüfung des User-Agent-Headers eingesetzt. Um eine möglichst kompatible Browserumgebung zu gewährleisten, bietet Ihnen Internet Explorer 8 und 9 daher die Möglichkeit, den User-Agent-String für Internet Explorer 7 zurückzugeben. Wechseln Sie hierzu in den Modus Internet Explorer 7-Standards. Um gleichzeitig Internet Explorer 7 zu emulieren, wechseln Sie über die den entsprechenden Schalter in der Befehlsleiste oder über das Menü Extras zur Kompatibilitätsansicht.



Beispiel für einen User-Agent-String für Internet Explorer 9:

```
User-Agent: Mozilla/5.0 (compatible; MSIE 9.0; Windows NT 6.1; WOW64; Trident/5.0)
```

Beispiel für einen User-Agent-String für Internet Explorer 8:

```
User-Agent: Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 6.0; SLCC1; .NET CLR 2.0.50727; .NET CLR 1.1.4322)
```

Beispiel für einen User-Agent-String für Internet Explorer 7:

```
User-Agent: Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.0; SLCC1; .NET CLR 2.0.50727; .NET CLR 1.1.4322)
```

Der normalerweise verwendete Modus ist Internet Explorer 8 bzw. 9-Standards. In diesem Modus wird die erweiterte Unterstützung der CSS 2.1 bzw. 3-Spezifikation genutzt. Um standardmäßig den Modus Internet Explorer 7-Standards zu nutzen, fügen Sie dem Head-Tag Ihrer Webseite das folgende Metatag hinzu:

```
<meta http-equiv="X-UA-Compatible" content="IE=7" />
```

## Kommt Ihnen dieser Code bekannt vor?

Ein häufig genutztes Verfahren bei der Entwicklung für verschiedene Browser sieht folgendermaßen aus:

```
<script type = "text/javascript">
function detectBrowser()
{
    var browser = navigator.appName;
    var b_version = navigator.appVersion;
    var version = parseFloat(b_version);
    if ((browser == "Netscape" || browser == "Microsoft Internet Explorer")
    && (version >= 5))
    {
        alert("Ihr Browser ist Ok!");
    }
    else
    {
        alert("Es wird Zeit, ein Upgrade Ihres Browsers durchzuführen!");
    }
}
</script>
```

Ein recht ungefährliches Beispiel. Ein paar kleine Veränderungen können jedoch erheblichen Ärger verursachen:

```
if (version >= 5 && version <= 7)
    alert("Ihr Browser hat den Test bestanden!");
else
    alert("Da ich vergessen habe, Internet Explorer 8 zu berücksichtigen,
ist Ihr Browser leider durchgefallen.");

if (is_nav4 || is_ie6) { /* Was nur mit diesen Versionen funktioniert */ }
if (ie_version == 7) { /* Was nur mit Internet Explorer 7 funktioniert */ }
```

Mit dem folgenden CSS-Workaround werden zwei unterschiedliche Eigenschaften genutzt – eine für Internet Explorer und eine für alle anderen Browser.

```
#header {margin-bottom: 3em;}
html>body #header {margin-bottom: 1em;}
```

Die folgenden Objekte werden oft inkorrekt zur Ermittlung eines bestimmten Browsers genutzt:

window.Iterator	Erkennt Firefox 2 und höher.
document.compatMode && document.all	Erkennt Internet Explorer 6 und höher.
window.XMLHttpRequest	Erkennt Internet Explorer 7,
Firefox1+ und Opera 8 und höher.	

## Graben Sie tiefer



Wenn Sie den Best-Practices folgen und die im vorherigen Abschnitt gezeigten Probleme vermeiden, dann ist es trotzdem möglich, dass Internet Explorer 8 und 9 Ihre Website nicht wie erwartet darstellt. Es kann sein, dass die Maßnahmen zur Verbesserung der Darstellung in Internet Explorer 8 und 9 eine fehlerfreie Rückwärtskompatibilität verhindern. In diesem Abschnitt finden Sie daher allgemeine Informationen zum Debuggen und Beheben von Kompatibilitätsproblemen mit Internet Explorer 8 und 9.

### Der User-Agent-String

Der Wechsel des User-Agent-Strings zum Internet Explorer 7-Modus ist eine einfache Möglichkeit, eine Webseite zu testen und erste Schritte zum Verständnis des Problems zu unternehmen. Wir haben zwar einen erheblichen Aufwand darin betrieben, Internet Explorer 8 und 9 rückwärtskompatibel zu gestalten, gehen jedoch trotzdem davon aus, dass einige Websites nicht mit dem neuen User-Agent-String umgehen können. Wir gehen jedoch auch davon aus, dass einfache Probleme mithilfe des Internet Explorer 7-Modus gelöst werden können. Diese so gewonnen Informationen geben dann den Website-Verantwortlichen die Möglichkeit die Probleme zu beseitigen.

### Die Entwicklertools

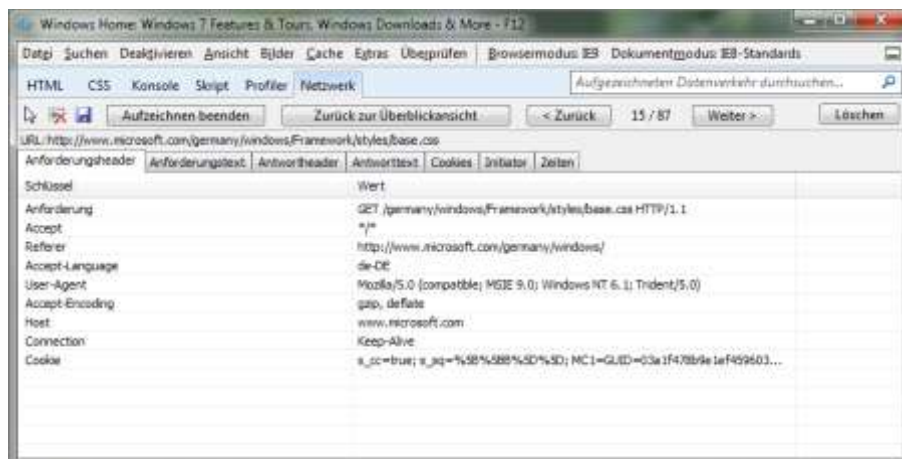
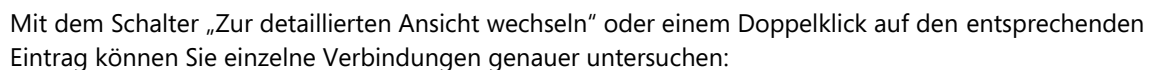
Detaillierte Informationen zu den Entwicklertools finden Sie im [Developer Tools Hands-On-Lab](#) (engl.) In diesem Abschnitt zeigen wir nur die grundlegenden Möglichkeiten auf. Bei Problemen mit tiefergehenden Ursachen – besonders in den Bereichen DOM, CSS, Skripting und Networking (nur Internet Explorer 9) – stehen Ihnen über die Entwicklertools zusätzliche Möglichkeiten zur Verfügung. Die Entwicklertools sind Teil von Internet Explorer 8 und 9. Sie können über die Werkzeugleiste oder über die Taste F12 aufgerufen werden. In den Entwicklertools finden Sie drei Registerkarten – entsprechend der drei oben genannten Bereiche.

Öffnen Sie Internet Explorer 8 oder 9, navigieren Sie zu einer beliebigen Website, und öffnen Sie die Entwicklertools. Klicken Sie in der Werkzeugleiste auf **Element durch Klicken auswählen**, und klicken Sie dann auf ein Element der Website. Sie bekommen nun alle Attribute des entsprechenden Elements und seine Position in der DOM-Hierarchie angezeigt. Mit diesem Verfahren können Sie sehr einfach einzelne Knoten in komplexen Webseiten herausgreifen (vor allem in dynamisch aufgebauten Seiten). Navigieren Sie zu einer Webseite mit einem Skript, und klicken Sie auf die Registerkarte **Skript**. Das Debuggen von Skripten starten Sie mit einem Klick auf **Debuggen starten**. Jetzt haben Sie mit einem Klick auf die entsprechende

Mit dem beschriebenen Verfahren steht Ihnen eine einfache Möglichkeit bereit, mit der Sie Probleme im Rahmen der Prüfung der Browseridentität über Objekte beheben können. Änderungen im Rahmen der Einhaltung von Standards können dazu führen, dass einige dieser Objekte nicht mehr wie erwartet reagieren. Möglicherweise geben Sie nur den Wert NULL zurück und führen so bei bestehendem Code zu Fehlern.

Mit Internet Explorer 9 wurde ein zusätzliches Analysewerkzeug eingefügt, mit dem sich das Verhalten auf Netzwerkebene direkt untersuchen lässt. Nachdem eine Aufzeichnung gestartet wurde, wird jeder Anforderung und die darauffolgende Antwort aufgezeichnet. Dabei wird auch das Timing berücksichtigt und dargestellt.

In der folgenden Abbildung sehen Sie beispielsweise, dass Internet Explorer 9 standardmäßig bis zu sechs gleichzeitige Verbindungen pro Server zulässt.



Hier können neben Request- und Response Header auch die Cookies, den „Initiator“ (zum Beispiel ein script-Tag) und die Zeiten genauer betrachtet werden. Über die Zeiten können Sie mithilfe der Antwortzeiten beispielsweise Lastprobleme bei Servern erkennen und so eigene Webanwendungen gezielt optimieren.

### **Fiddler**

Bei Problemen, die durch Änderungen an Requests, Responses, Headern, Statuscodes, Zeichensätzen und durch die Verarbeitung von Daten vor der Weitergabe an den Browser entstehen, können Sie das Tool [Fiddler](#) (engl.) nutzen. Mit diesem Tool haben Sie die Möglichkeit, Requests und Responses vor deren Übertragung zu überprüfen. **Fiddler** ermöglicht außerdem eine Nachverfolgung des Netzwerkverkehrs (beispielsweise Connects, Posts und Webservice-Requests), die über andere Wege nur schwer zu realisieren ist.

## Einkreisen des Problems

Mit den folgenden Schritten können Sie die Ursache für das Problem Schritt für Schritt nachstellen und einkreisen.

### Schritt 1: Überprüfung des Dokumentmodus

1. Haben Sie Änderungen an der Registrierungsdatenbank durchgeführt?
2. Ist die **Kompatibilitätsansicht** aktiviert? Überprüfen Sie den Schalter in der Befehlsleiste, die Menüoption und die Domänenliste.
3. Gibt es im Markup-Code oder http-Header der Website ein Metatag, das eine Version definiert?
4. In welchen Modus versetzt die DDT den Browser?
5. Nutzen Sie die Entwicklertools, um den Browsermodus und den Dokumentmodus zu überprüfen oder festzulegen?
6. Führen Sie das folgende Skript über die Skript- oder Adressleiste aus:  
`javascript:alert(document.documentMode).`
7. Versuchen Sie den Mauszeiger über auswählbarem Text zu platzieren. Wenn sich der Mauszeiger nicht in das Textwahlsymbol ändert, dann wird die Webseite im Modus Internet Explorer 8-Standards angezeigt.
8. Navigieren Sie zu einer Marquee-Testseite. Marquee-Elemente wurden noch nicht in Internet Explorer 8 implementiert.

### Schritt 2: Lokales Speichern der Webseite

Da sich Webseiten sehr schnell verändern können, ist dieser Schritt besonders wichtig.

1. Speichern Sie die Webseite lokal. Benennen Sie sie nach ihrer Domäne (beispielsweise microsoft.htm). Wenn während des Speicherns keine Fehlermeldungen auftreten, fahren Sie mit dem nächsten Schritt fort.
2. Der gemeinsam mit der htm-Datei gespeicherte Ordner enthält die Ressourcendateien (beispielsweise .css, .js und .jpg).
3. Die html-Datei ist mit den Ressourcendateien verknüpft. Wenn Sie die htm-Datei löschen, dann werden auch die Ressourcendateien gelöscht.
4. Wenn Sie die Option **Webseite, nur HTML** nutzen, dann werden relative Pfade wie beispielsweise `src=../support/example.css` nicht angepasst.

### Schritt 3: Reproduzieren des Problems

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3c.org/TR/1999/REC-html401
<!-- saved from url=(0020)http://www.live.com/ -->
<HTML lang=en class="liveApp la_en lo_us" xml:lang="en"
xmlns="http://www.w3.org/1999/xhtml" xmlns:web><HEAD><TITLE>Live Search</TITLE>
<META content="text/html; charset=utf-8" http-equiv=Content-Type>
<META name=description
content="Find exactly what you are looking for ! FAST! With Live Search.">
<META name=ROBOTS content=NOODP><LINK title="Live Search" rel=search
type=application/opensearchdescription+xml
href="http://www.live.com/search/search.xml">
<SCRIPT language=JavaScript>
    if(location.href.indexOf("#")!==-1&&location.href.indexOf("#")<location.href.lastIndexOf("?")){
    else if(location.hash!=""&&location.hash!="#"){var url=location.href.substring(0,location.href
    var Timers = new function() {this.next = function(){};}}
</SCRIPT>
<LINK rel=stylesheet type=text/css href="http://www.live.com/Live%20Search_files/searchlanding2.css"
media="screen, print, projection, tv">
<META name=GENERATOR content="MSHTML 6.00.6001.17181"></HEAD>
<BODY><!--First Run -->
<DIV id=sv_page>
<DIV id=sv_width><NOSCRIPT>
<META content="0: URL=downlevel2.aspx?js=0&ap;" http-equiv=Refresh></NOSCRIPT>
</BODY>
```

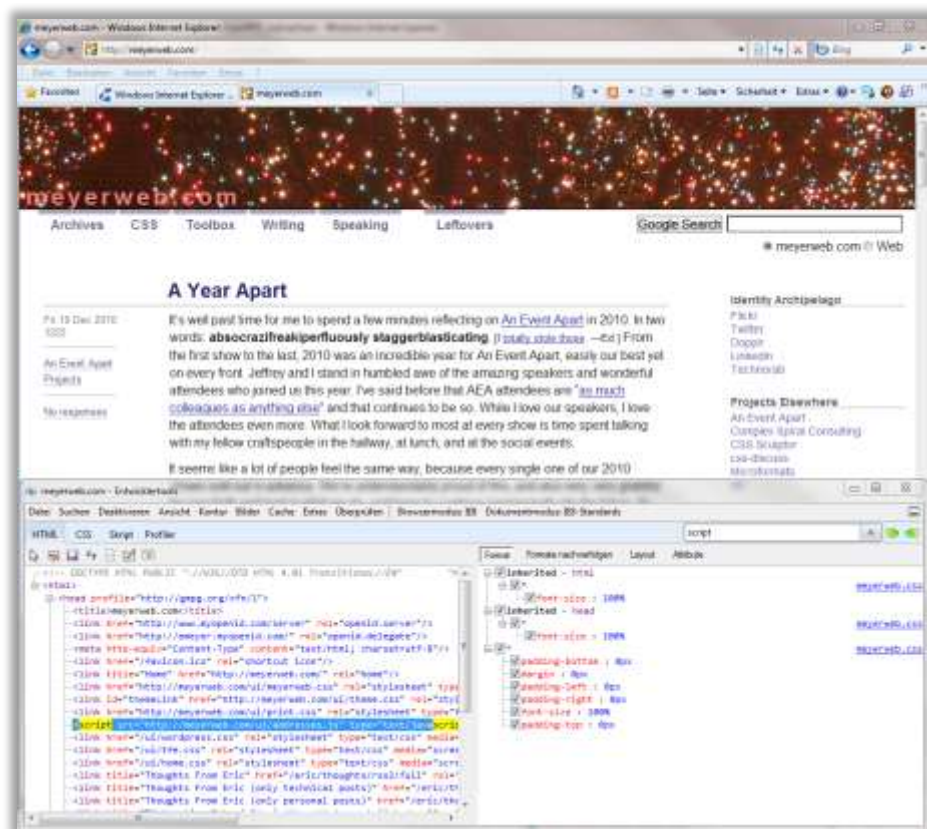
Relativer Pfad passt in lokaler Kopie nicht mehr

Lokaler Pfad passt

1. Der erste Lösungsschritt ist das Nachstellen des Problems mithilfe der lokalen Kopie.
2. Wenn das Problem mit der lokalen Kopie direkt nachstellbar ist, dann können Sie mit Schritt 4 fortfahren.
3. Tritt das Problem bei der lokalen Kopie jedoch nicht auf, dann sollten Sie als Erstes alle externen Ressourcenpfade suchen und bearbeiten. In dem in der Abbildung zu sehenden Beispiel würden Sie den relativen Pfad `../../search/search.xml` durch den vollständigen Pfad `http://www.live.com/live/search/search.xml` ersetzen.
4. Bearbeiten Sie immer nur eine Codezeile, und aktualisieren Sie dann die Seite.
5. Sobald das Problem, wie gewünscht, auch in der lokalen Kopie auftritt, fahren Sie mit dem nächsten Schritt fort.

#### Schritt 4: Grobe Einkreisung des Problembereichs

1. Löschen Sie als Erstes die Inhalte im Header. Hier finden sich die meisten externen Referenzen.
2. Entfernen Sie alle Script-Blöcke, Metatags, den Title-Tag und alles Andere, das nicht von grundlegender Bedeutung für das Layout der Webseite ist.
3. Entfernen Sie so viele Style-Sheet-Verweise wie möglich. Lassen Sie nur die erforderlichen Verweise bestehen.
4. Aktualisieren Sie die Seite, und prüfen Sie, ob das Problem weiterhin auftritt.
5. Suchen Sie im Rest des Dokumentes nach Script- oder Style-Blöcken. Entfernen Sie die einzelnen Blöcke, und prüfen Sie, ob das Problem weiterhin auftritt.



#### Schritt 5: Genaue Eingrenzung des Problembereichs

Mit diesem Schritt stellen Sie fest, wo genau in der DOM-Hierarchie das Problem zu finden ist. Beginnen Sie mit dem Knoten auf der obersten Hierarchieebene, in der Sie das Problem lokalisiert haben. In diesem Beispiel ist dies das Element mit der ID „extra“. Alle Elemente außerhalb dieses Knotens haben für gewöhnlich nichts mit dem Problem zu tun und können gelöscht werden.

```

<HTML>
<HEAD>
<meta http-equiv="X-UA-Compatible" content="IE=8" />
<LINK rel="stylesheet" type="text/css" href="meyerweb_files/meyerweb" />
</HEAD>
<BODY>
<DIV id="extra">
<DIV class="panel">
<UL>
<LI>
<A href="asdf">Flickr</A>
</LI>
</UL>
</DIV>
</DIV>
</BODY>
</HTML>

```

Indem Sie Codezeile für Codezeile, Styles, IDs und Klassen beseitigt haben, haben Sie die Webseite nun so weit wie möglich bereinigt.

### Schritt 6: Bereinigung des Style-Sheet

In diesem Schritt bereinigen Sie die CSS-Datei. Öffnen Sie die Datei in einem Editor, und kopieren Sie alle Styles in einen <STYLE>-Block in die lokale Kopie der Webseite. In den meisten Fällen ist die Liste der Styles recht lang. Gibt es mehr als eine CSS-Datei, so wiederholen Sie diesen Schritt für alle Dateien. Stellen Sie sicher, dass Sie den Verweis auf die CSS-Datei löschen.

Entfernen Sie alle Styles, die nicht zur Anzeige der verbliebenen Webseitenelemente erforderlich sind. Gehen Sie mit den verbliebenen Styles nun genau wie oben schon beschrieben vor: Löschen Sie die einzelnen Styles, und prüfen Sie nach jedem Löschen, ob das Problem noch besteht.

```

<HTML>
<HEAD>
<meta http-equiv="X-UA-Compatible" content="IE=8" />
<style>
#extra .panel LI
{
TEXT-INDENT: -1em;
MARGIN-LEFT: 1em;
}
</style>
</HEAD>
<BODY>
<DIV id="extra">
<DIV class="panel">
<UL>
<LI>
<A href="asdf">Flickr</A>
</LI>
</UL>
</DIV>
</DIV>
</BODY>
</HTML>

```

Das Ergebnis Ihrer Bemühungen sollte eine relativ kleine HTML-Datei sein, mit der Sie das Problem sehr einfach nachstellen und testen können.




---

**Hans Krijger** ist Mitglied des Internet Explorer-Teams und befasst sich mit der Website-Kompatibilität.