



Form og dens underlige box model

Denne artikel handler om hvilke problemer man som webudvikler kan komme ud for, og hvordan man løser dem. Jeg kommer dog ikke ind på problemetikerne omkring brug af ældre browser som IE7.

Skrevet den **15. Jun 2010** af **andreas13_fam** | kategorien **(D)HTML / CSS** | ★★☆☆☆

Lige først

Dette er min første artikel, så jeg vil sætte pris på at i er konstruktive i jeres kritik.

Problemet

Problemet beskrives enlig bedst med et eksempel:

Bemærk: at man aldrig bør brug `br {clear:both;}` jeg har kun gjort det for at gøre eksemplet simpelt

```
<!DOCTYPE html>
<html lang="da"><head>
  <title>Form problemmet</title>
  <meta charset="utf-8">
  <style>
    input, textarea, button {
      width: 300px;
      height: 20px;
      border: 1px solid #000;
      margin: 3px;
      padding: 3px;
    }

    div {border: 1px solid #F00;float: left;}
    br {clear: both;}
  </style>
</head><body>
<form action="#" method="get">
  <p>
    <div><input type="text"></div><br>
    <div><input type="password"></div><br>
    <div><input type="file"></div><br>
    <div><input type="image"
src="http://www.google.dk/images/srpr/nav_logo13.png"></div><br>
    <div><input type="radio"></div><br>
    <div><input type="checkbox"></div><br>
    <div><textarea></textarea></div><br>
    <div><button type="submit">test</button></div><br>
  </p>
</form>
</body></html>
```

Her bruges der de klassiske input type der kendes fra Web Form 1.0, som det fremgår i alle browser, ser radio, checkbox, file og button anderledes ud en resten. Dette skyldes at de benytter det der kaldes IE box

modellen.

Ganske kort er der 2 box modeller

Content-box

Content-box er bedst kendt som standart box modellen og fremkommer altid når man benytter en DOCTYPE, det søger nemlig for at Internet Explorer ikke går i quirksmode (en bagud kompatibel tilstand). Content-box beregnes således at width og height er selve den del hvor der kan være indhold, det kan være tekst eller andre elementer. Herefter ligges en padding på som fungerer som en indre magen, dernæst kommer border som er kanten af boksen. Tilsidst er kommer så margin som er den ydre magen.

En box som

```
<div style="width: 10px; padding: 3px; border: 2px; margin: 1px">
```

vil således få den totale brede på $10+3*2+2*2+1*2 = 22px$

Border-box (IE-modellen)

Border-box moden er oprindeligt en Microsoft model, som siger at width og height beskriver hvor stor den synlige den af boksen er, padding og border har derfor ikke nogen indflydelse på størrelse, det har width, height og margin alene.

En box som

```
<div style="width: 10px; padding: 3px; border: 2px; margin: 1px">
```

vil således få den totale brede på $10+1*2 = 12px$

CSS3 box modellen

Med CSS3 er der ikke længere DEN rigtige model, men derimod 2. Content-box modellen er stadig standart modellen, men border-box modellen er også en mulighed.

Det angives i CSS således

```
box-sizing: content-box | Content-box;
```

Denne egenskab er understøttet af IE8+ og Opera, vil man være lidt mere kompatibel skal man tilføje -webkit og -moz, så det bliver

```
box-sizing: content-box | Content-box;
-moz-box-sizing: content-box | Content-box;
-webkit-box-sizing: content-box | Content-box;
```

Mozilla understøtter en 3. Mulighed men den vil jeg ikke komme mere ind på.

Løsningen

Med denne viden kan vi løse de problemer der var i det første eksempel:

```
<!DOCTYPE html>
<html lang="da"><head>
  <title>Form problemmet</title>
  <meta charset="utf-8">
  <style>
    input, textarea, button {
      -moz-box-sizing: content-box;
      -webkit-box-sizing: content-box;
      box-sizing: content-box;
```

```

        width: 300px;
        height: 20px;
        border: 1px solid #000;
        margin: 3px;
        padding: 3px;
        background: none;
    }

    div {border: 1px solid #F00;float: left;}
    br {clear: both;}
</style>
</head><body>
<form action="#" method="get">
    <p>
        <div><input type="text"></div><br>
        <div><input type="password"></div><br>
        <div><input type="file"></div><br>
        <div><input type="image"
src="http://www.google.dk/images/srpr/nav_logo13.png"></div><br>
        <div><input type="radio"></div><br>
        <div><input type="checkbox"></div><br>
        <div><textarea></textarea></div><br>
        <div><button type="submit">test</button></div><br>
    </p>
</form>
</body></html>

```

Og dog

Efter flere test med 2. eksempel kan man konstatere:

Oprea : Alt fungere

IE : file-input fungere ikke

Webkit : checkbox-input og radio-input fungere ikke

Firefox: file-input, checkbox-input og radio-input fungere ikke

JavaScript løsningen

Dette er selvfølgelig stadig ikke optimalt, men for at få løst de sidste problemer er man desværre nødt til at skulle ud i en JavaScript løsning der først kan afvikles må DOM træet er indlæst, placer derfor koden lige før </body> :

En sådan løsning ser vil se sådan ud:

```

<!DOCTYPE html>
<html lang="da"><head>
    <title>Form problemmet</title>
    <meta charset="utf-8">
    <style type="text/css">
        input, textarea, button {
            -moz-box-sizing: content-box;
            -webkit-box-sizing: content-box;
            box-sizing: content-box;

```

```

        width: 300px;
        height: 20px;
        border: 1px solid #000;
        margin: 3px;
        padding: 3px;
        background: none;
    }

    div[debug] {border: 1px solid #F00;float: left;}
    br {clear: both;}
</style>
</head><body>
<form action="#" method="get">
    <div debug><input type="text" id="test"></div><br>
    <div debug><input type="password"></div><br>
    <div debug><input type="file"></div><br>
    <div debug><input type="image"
src="http://www.google.dk/images/srpr/nav\_logo13.png"></div><br>
    <div debug><input type="radio"></div><br>
    <div debug><input type="checkbox"></div><br>
    <div debug><textarea></textarea></div><br>
    <div debug><button type="submit">test</button></div><br>
</form>
<script type="text/javascript">
(function () {
    //Funktion der finder den udregnede style
    var getStyle = function (el,styleCom,stylePro) {
        if (el.style[stylePro] && el.style[stylePro] != '') {
            return el.style[stylePro];
        }

        if (el.currentStyle) {
            return el.currentStyle[stylePro];
        } else if (window.getComputedStyle) {
            return
document.defaultView.getComputedStyle(el,null).getPropertyValue(styleCom);
        }
    };

    //Finder de elementer der stadig ikke virker
    var nodes =
document.querySelectorAll('input[type=file],input[type=radio],input[type=check
box]'),i;
    //Køre alle elementer igennem
    for (i=0;i<nodes.length;i++) {
        //Kloner input elementet
        var clone = nodes[i].cloneNode(true);

        //Opretter div elementet som bekendt undersætter content-box
        var div = document.createElement('div');
        div.style.overflow = 'hidden';
        //Kopier og nulstiller alle box relevante værdier
        div.style.marginTop = getStyle(nodes[i],'margin-top'
,'marginTop' );

```

```

    div.style.marginRight = getStyle(nodes[i], 'margin-right'
, 'marginRight' );
    div.style.marginBottom = getStyle(nodes[i], 'margin-bottom'
, 'marginBottom' );
    div.style.marginLeft = getStyle(nodes[i], 'margin-left'
, 'marginLeft' );
    clone.style.margin = '0';

    div.style.paddingTop = getStyle(nodes[i], 'padding-top'
, 'paddingTop' );
    div.style.paddingRight = getStyle(nodes[i], 'padding-right'
, 'paddingRight' );
    div.style.paddingBottom = getStyle(nodes[i], 'padding-
bottom', 'paddingBottom');
    div.style.paddingLeft = getStyle(nodes[i], 'padding-left'
, 'paddingLeft' );
    clone.style.padding = '0';

    div.style.borderTopWidth = getStyle(nodes[i], 'border-top-
width' , 'borderTopWidth' );
    div.style.borderRightWidth = getStyle(nodes[i], 'border-
right-width' , 'borderRightWidth' );
    div.style.borderBottomWidth = getStyle(nodes[i], 'border-
bottom-width', 'borderBottomWidth');
    div.style.borderLeftWidth = getStyle(nodes[i], 'border-left-
width' , 'borderLeftWidth' );

    div.style.borderTopColor = getStyle(nodes[i], 'border-top-
color' , 'borderTopColor' );
    div.style.borderRightColor = getStyle(nodes[i], 'border-
right-color' , 'borderRightColor' );
    div.style.borderBottomColor = getStyle(nodes[i], 'border-
bottom-color', 'borderBottomColor');
    div.style.borderLeftColor = getStyle(nodes[i], 'border-left-
color' , 'borderLeftColor' );

    div.style.borderTopStyle = getStyle(nodes[i], 'border-top-
style' , 'borderTopStyle' );
    div.style.borderRightStyle = getStyle(nodes[i], 'border-
right-style' , 'borderRightStyle' );
    div.style.borderBottomStyle = getStyle(nodes[i], 'border-
bottom-style', 'borderBottomStyle');
    div.style.borderLeftStyle = getStyle(nodes[i], 'border-left-
style' , 'borderLeftStyle' );
    clone.style.border = 'none';

    div.style.width = getStyle(nodes[i], 'width' , 'width'
);
    div.style.height = getStyle(nodes[i], 'height', 'height');

    //Tilføjer clonen til div elementet
    div.appendChild(clone);

    //Indsætter elementet før det fundne element
    nodes[i].parentNode.insertBefore(div, nodes[i]);

```

```
        //Fjerner det funde element, nu hvor det andet er indsat
        nodes[i].parentNode.removeChild(nodes[i]);
    }
}());
</script>
</body></html>
```

Stadig ikke komplet

Desværre er understøttelsen omkring `getComputedStyle` ikke komplet, det betyder at hvis `margin`, `padding`, `border`, `height` og `width` ikke er angivet direkte i `style=""` er det ikke sikkert det virker komplet. I den forbindelse er det selvfølgelig relevant at tale om OOCSS, men det bliver hvis for meget.

Endelig browser support

Den endelig browser understøttelse ser således ud:

Opera : Alt fungere

IE : Alt fungere

WebKit : checkbox-input og radio-input fungere ikke (dog bedre)

Firefox: file-input, checkbox-input og radio-input fungere ikke (dog bedre)

Husk på at udgangspunktet var at radio, checkbox, file og button ikke fulgte Content-box model. Det problem er væsentligt formindsket.

Til sidst

Stavefejl og ligende små rettelser, sætter jeg pris på vil komme som privat-besked.

Kommentar af andreas13_fam d. 15. Jun 2010 | 1

Jeg kan desværre ikke redigere denne guide på grund af at eksperten ikke fjerner mine tags når jeg redigere