

Internet Explorer 7

Uusi selain uudet harmit, uudet hyödyt

Tuoreen selainversion myötä vanhat verkkosivut voivat oikutella. Vääränlainen korjaus saattaa romuttaa sivun, mutta taitava ylläpitäjä välttää karikat ja hyödyntää selaimen uusia piirteitä.

**TEKSTI:
JUKKA K. KORPELA**

Internet Explorerin versio 7 on nyt noin joka neljännellä internetin käyttäjällä. Tämä on niin paljon, että sivuntekijän on syytä ottaa IE 7 huomioon. Toisaalta IE 6 on edelleen enemmistönä.

Niinpä toistaiseksi ei voi rakentaa paljoakaan sen varaan, että IE 7:ssä on korjattu koko joukko IE 6:n virheitä ja puutteita. Se muun muassa noudattaa standardeja edeltäjänsä paremmin.

Verkkosivuista huolehtiville IE 7 merkitsee ensisijaisesti ongelmia ja vasta pitemmällä aikavälillä ratkaisuja. Jotakin IE 7:n uutuuksia voi kuitenkin hyödyntää jo nyt.

Parannukset sivuntekijän kannalta koskevat etenkin laajempaa tyyliohjeiden eli css:n tukea sekä erilaisten virheiden korjauksia. Näissä on kyse piirteistä, jotka ovat jo kauan olleet käytettävissä Firefoxissa ja Operassa mutta joiden käyttö on käytännössä ollut vähäistä IE:n puutteiden takia.

IE 7 leviää osittain automaattisenakin päivityksenä, mutta leviämistä rajoittaa muun muassa se, ettei se edes ole saatavissa eräisiin vielä käytössä oleviin windowsin versioihin.

> **Temput lakkaavat toimimasta**

Monilla verkkosivuilla käytetään erilaisia ovelia keinoja antaa eri selainten käsiteltäväksi erilaista css-koodia. Taustalla on muun muassa se, että sivuntekijät ovat halunneet käyttää ratkaisuja, jotka toimivat hyvin esimerkiksi Firefoxissa ja Operassa mutta sotkisivat ulkoasun IE 5:ssä tai IE 6:ssa.

Tätä varten rakenteet on "suojattu" koodilla, josta tiedetään, että IE 6 ei siitä selviä. Toisaalta on kirjoitettu sellaista css-koodia, joka ei vaikuta mitään standardien mukaan oikein toimivalla selaimella, mutta josta tiedetään, että IE 6 toteuttaa sen.

Tyypillisimpiä tällaisista tempuista on niin sanottu **Holly hack**, jossa syötetään haluttu css-koodi vain IE 6:lle ja IE:n aiemmille versioille seuraavaan tapaan:

```
* html body { width: 500px; }
```

Sääntö viittaa sellaiseen <html>-elementtiin, joka on jonkin muun elementin sisällä. Sellaista ei html-kielessä voi olla, mutta IE:llä oli asiaita ihan oma käsityksensä. IE 7:ssä on (standards-tilassa) korjattu tämä vir-

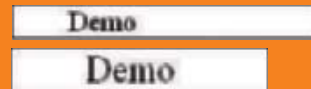
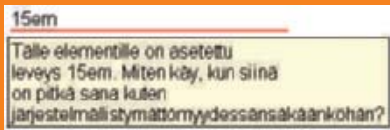
he – mutta ei ehkä niitä virheitä, joiden käsittelemiseksi sääntö kirjoitettiin. Tilanteesta riippuu, miten vakavia seuraukset ovat.

Jos halutaan kirjoittaa css-koodia joillekin määrätuille IE:n versioille, on turvallisempaa käyttää niin sanottua ehdollista kommenttia. Siinä voidaan testata IE:n versionumeroa. Muiden selainten (ja html:n määrittelyn) kannalta kyseessä on kommentti, eli ne eivät näe koko koodia.

Jos sivuilla käytetään selaimentunnistusta, se voi mennä aivan pieleen, jos IE 7:n mahdollisuuttakaan ei ole otettu huomioon. Jotkin tunnistustavat erottelevat esimerkiksi IE 4:n, IE 5:n ja IE 6:n mutta pitävät IE 7:ää "vanhana selaimena".

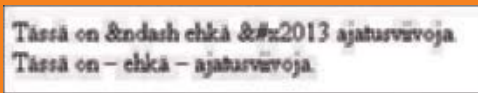
> **Selaimen oikea vai haluttu toiminta**

Suosituimmissa selaimissa on kaksi eri toimintatilaa, **standards**-tila ja **quirks**-tila (oikkutila) taikka Firefoxin suomennoksen termein standarditila ja yhteensopivuustila. Jälkimmäisessä selain pyrkii jäljittelemään selainten vanhojen versioiden, erityisesti IE 4:n ja IE 5:n, erilaisia virheitä. Tämä tun-

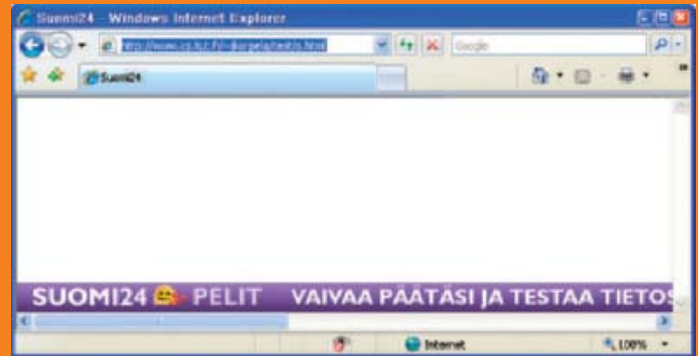


▲ **Ylivuodon erilainen käsittely** IE 6:ssa ja IE 7:ssä näkyy selvästi, jos elementillä on reunaviiva. IE 7 pitää laatikon mitassaan, vaikka sisältö pursuaa.

▲ **Selainten toimintatilojen ero** voi vaikuttaa muun muassa sivun osien kokoon. Tässä on sama elementti standards-tilassa ja quirks-tilassa. Standards-tilassa täytteiden osuus tulee mukaan laatikon leveyteen.



▲ **Puolipisteen puuttuminen** koodista voi estää merkin näkymisen IE 7:ssä.



► **Jos sivun koodiin** vain lisätään doctype-ilmoitus korjaamatta sivun virheitä, saattaa jopa kaikki sisältö hävitä näkyvistä. Tässä on jäänyt jäljelle vain alaosan mainospalkki.

tuu mielettömältä, mutta sitä perustellaan sillä, että monet vanhat sivustot on toteutettu kyseisten virheiden mukaisesti.

Esimerkiksi elementin leveyden asettaminen toimii quirks-tilassa osittain toisin kuin standards-tilassa. Quirks-tilassa leveyteen lasketaan mukaan myös vaakasuora täyte (padding). Tällä voi olla olennainen merkitys sivun kokonaistaitolle. Myös muun muassa fonttikoon ilmoittavilla avainsanoilla (esimerkiksi font-size: large) on IE:ssä eri merkitykset eri tiloissa.

IE 7 on siis tavallaan vanhan IE:n ja melko standardin mukaisen selaimen yhdistelmä. Siksi on tärkeää, kumpaan tilaan verkkosivu saattaa sen. Tila vaikuttaa monin tavoin sivun osien ulottuvuuksiin ja aseteluun mutta myös esimerkiksi fonttiasetuksiin.

Selaimet valitsevat tilan sen mukaan, onko verkkosivun alussa html:n määrittelyn mukainen dokumenttityypin ilmoitus eli doctype-ilmoitus. Jos ei ole, selain menee quirks-tilaan. Tästä seuraa:

● Vanhoille sivuille, joilla doctype-ilmoitus puuttuu (tai joilla se on väärämuotoinen), sitä ei kannata lisätä (tai korjata), ellei sa-

malla muuteta sivua kokonaisuudessaan html:n ja css:n määrittelyjen mukaiseksi.

● Uusilla sivuilla kannattaa alusta alkaen käyttää oikeanlaista doctype-ilmoitusta.

Koodiesimerkeissä seuraavalla sivulla on kaksi määrittelyjen mukaista doctype-ilmoitusta. Selaimen toiminnan kannalta ei ole merkitystä sillä, kumpaa näistä käytetään, mutta sillä on merkitystä sivun merkkauksen tarkistamisessa validaattorilla. Pienikin virhe näissä ilmoituksissa aiheuttaa yleensä quirks-tilan.

IE:hen voi asentaa niin sanotun aktiivisen kirjanmerkin (Suosit-listaan liitettävän pienen javascript-ohjelman), jonka valitsemalla näkee, kummassa tilassa selain on.

► Jos korjaat, korjaa kunnolla

Selaimen kaksi erilaista toimintatila ei välttämättä ole mikään uutuus, mutta IE 7:ssä tilojen eron merkitys on aiempaakin suurempi. Muun muassa osa parannuksista css-tukeen on käytössä vain standards-tilassa. Tällainen on esimerkiksi ominaisuus `max-width` ja asetukset `white-space: pre`.

Jos halutaan hyödyntää IE 7:n tuomia uusia mahdollisuuksia, kannattaakin ensin tarkistaa, että sivu toimii standards-tilassa. Tämä merkitsee kahta asiaa: oikeanlaista dokumenttityypin ilmoitusta ja sitä, että sivu todella noudattaa "standardeja" eli lähinnä html:n ja css:n määrittelyjä. Tämä saattaa vaatia paljonkin, sillä sivu on ehkä tehty periaatteella "tämä toimii IE:ssä", jolloin on nojautunut IE:n vanhojen versioiden toimintatapaan.

Jos verkkosivu hajoaa käsiin, kun sen alkuun lisätään dokumenttityypin määrittely, ei sivua ehkä kannata yrittää korjata eikä kehittää. Se todennäköisesti toimii kohtalaisesti vielä pitkään, ja sisältöä voidaan tietysti päivittää. Sivun rakenteen korjaaminen nykyaikaiseksi saattaa kuitenkin olla isompi työ kuin se, että se suunnitellaan alkaen uudestaan jonkin sopivan uudistuksen yhteydessä.

► Xml-ilmoituksen vaikutus

Jos sivulla käytetään xhtml:ää ja sivun alussa on niin sanottu xml-ilmoitus kuten `<?xml version="1" ?>` (jota xhtml:ssä suositellaan), niin IE 6 toimii aina quirks-tilassa.

Sen sijaan IE 7 toimii standards-tilassa, jos seuraa oikeanlainen doctype-ilmoitus.

Jos tästä seuraa sivun hajoaminen, ensiavuksi voi lisätä sivun alkuun kommentin `<!-->`, jolloin IE 7:kin toimii quirks-tilassa. Sen jälkeen kannattaa harkita, voisiko sivun kunnostaa toimimaan standards-tilassa.

► Ylivuoto-ongelmia

IE 7 noudattaa (toisin kuin IE 6) css:n määrittelyjä silloin, kun elementin sisältö ei mahdu elementille asetettuihin ulottuvuuksiin. Yli menevä sisältö vuotaa elementin ulkopuolelle. IE 6 sen sijaan suurentaa itse elementtiä.

Uusi, standardin mukainen toimintatapa koskee vain standards-tilaa. Jos se sekoittaa sivun taiton, voi hätäratkaisuksi muuttaa sivua niin, että IE 7 toimii quirks-tilassa. Varsinaisena ratkaisuna voi poistaa tai väljentää elementille asetettuja leveys- ja korkeusvaatimuksia.

► Mistä "–" ilmaantui?

Sivuilla, jotka ennen toimivat, saattaa näkyä &-merkillä alkavia enemmän tai vähemmän outoja

merkkijonoja. Näin käy IE 7:n molemmissa toimintatiloissa.

Syynä on, että IE 7 ei tunnista erikoismerkkien esittämiseen käytettyjä entiteetti- ja viittauksia kuten **–** eikä heksadesimaalisia merkkiviittauksia kuten **&x2013;**, ellei niissä ole puolipistettä. Puolipisteen saa html:n sääntöjen mukaan jättää pois viittauksista useissa tilanteissa, mutta IE 7 ei siis toimi tämän mukaan.

Ongelma ei koske kaikkia entiteetti- ja viittauksia, mutta on helppoa käyttää niissä kaikissa puolipistettä. Tällöin entiteetti- ja merkkiviittaukset ovat edelleenkin toimiva tapa esittää erikoismerkkejä.

> Enimmäisleveys kohdalleen

Sivun sisällön leveydelle on yleensä hyvä asettaa jokin yläraja, jotta tekstirivit eivät levähtäisi suhteettoman pitkiksi. Toinen vaihtoehto on kiinteän leveyden asettaminen, mutta silloin sivu toimii huonosti, jos selaimen ikkuna on kapeahko.

Enimmäisleveyden asettamiseen on monia keinoja, mutta siistein niistä on css-ominaisuuden `max-width` asettaminen. Esimerkiksi `body { max-width: 25em; }` vastaa keskimäärin noin 60 merkin leveyttä.

IE 7 samoin kuin muun muassa Firefox ja Opera tuntee `max-width` -ominaisuuden, IE 6 ei. Jos enimmäisleveyden asettaminen myös IE 6:ssa on tärkeää, on siis syytä edelleen käyttää muita tekniikoita. Helpoin tapa on yksisoluisen taulukon, jonka solulle asetetaan leveydeksi haluttu enimmäisleveys.

Eräs kompromissi on asettaa enimmäisleveys mutta saada IE:n vanhat versiot (6:een asti) käyttämään vastaavaa kiinteää leveyttä. Tämä on esitetty koodiesimerkeissä.

IE 7:ssä on joitakin virheitä `max-width` -ominaisuuden toteutuksessa. Sen asettaminen esimerkiksi listalle (`ul`- tai `ol`-elementille) aiheuttaa usein kummallisuksia. Niinpä onkin parempi kirjoittaa lista `div`-elementin sisään ja asettaa enimmäisleveys tälle `div`-elementille.

> Hiireen reagoivia elementtejä

IE 7:ssä voi käyttää niin sanottua yleistettyä `:hover`-pseudoluokkaa, jonka avulla saadaan elementin ulkoasu muuttumaan, kun hiirtä liikuttamalla viedään osoitin elementin päälle. IE:ssä tämä oli aiemmin mahdollista vain linkeille.

Tällä tavoin voidaan helposti toteuttaa efektejä, jotka aiemmin on tehty kömpelösti javascriptillä ja `onmouseover`- ja `onmouseout`-määrittelyillä.

Koodiesimerkki näyttää, miten jonkin kappaleen taustan väri muuttuu osoittimen liikkeen mukaan.

> Uudet selektorit yksinkertaistavat css-koodia

IE 7:ssä on css:n määrittely (CSS 2.1) toteutettu aiempaa paljon laajemmin joskaan ei kattavasti. Parannukset koskevat etenkin selektoreita, joilla valitaan, mihin sivun elementteihin jokin tyyllisääntö kohdistetaan.

IE:n kannalta uudet selektorit on esitetty oheisessa taulukossa. Ne kannattaa nyt ottaa omaan työkalupakkiin.

Koodiesimerkissä näytetään, miten lomakkeen tekstinsyöttökenttien fontiksi asetetaan Courier New muuttamatta muiden kenttien ulkoasua. ■

Koodiesimerkit

Ehdollinen kommentti

IE:n vanhat versiot lukevat tiedoston `vanha-ie.css`, muut selaimet pitävät riviä kommenttina.

```
<!--[if lte IE 6]>
<link rel="stylesheet" type="text/css" href="vanha-ie.css">
<![endif]-->
```

Doctype-ilmoitus

Ensimmäinen esimerkeistä tarkoittaa "puhtaampaa" html:ää.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN"
"http://www.w3.org/TR/html4/strict.dtd">
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
```

Enimmäisleveys

Vanhat IE:t saadaan käyttämään kiinteää leveyttä ja muut selaimet enimmäisleveyttä.

```
<style type="text/css">
body { max-width: 25em; }
</style>
<!--[if lte IE 6]>
<style type="text/css">
body { width: 25em; }
</style>
<![endif]-->
```

Hiireen reagoiva elementti

Kappaleen tausta muuttuu keltaiseksi, kun osoitin viedään sen päälle.

```
p:hover { background: yellow; color: black; }
```

Selektori

Lomakkeen syöttökenttien fontiksi asetetaan Courier New.

```
input[type="text"] { font-family: Courier New, monospace; }
```

Lisätietoa netistä

- www.positioniseverything.net Selainten css-toteutusten virheitä ja niiden kiertämisen tapoja, mm. Holly Hack.
- www.cs.tut.fi/~jkorpela/quirks-mode.html Standards- ja quirks-tilan valinnan vaikutuksia.
- dorward.me.uk/www/bookmarklets/qors Ladattavissa aktiivinen kirjanmerkki, joka näyttää selaimen toimintatilan.
- validator.w3.org Html-koodin tarkistus.
- jigsaw.w3.org/css-validator Css-koodin tarkistus.

IE 7:n uudet selektorit

Malli	Nimitys	Esimerkki	Esimerkin selitys
A + B	seuraaajaselektori	h2 + p	p-elementti, joka heti seuraa h2-elementtiä
A > B	alielementtiselektori	ul + li	li-elementti, joka on suoraan ul-elementin osana
[a]	määriteselektori	[title]	elementti, jolla on title-määrite
[a = "b"]	määritearvoselektori	[type="text"]	elementti, jolla on määrite type="text"
:first-child	esikoisselektori	li:first-child	li-elementti, joka on jonkin listan ensimmäinen