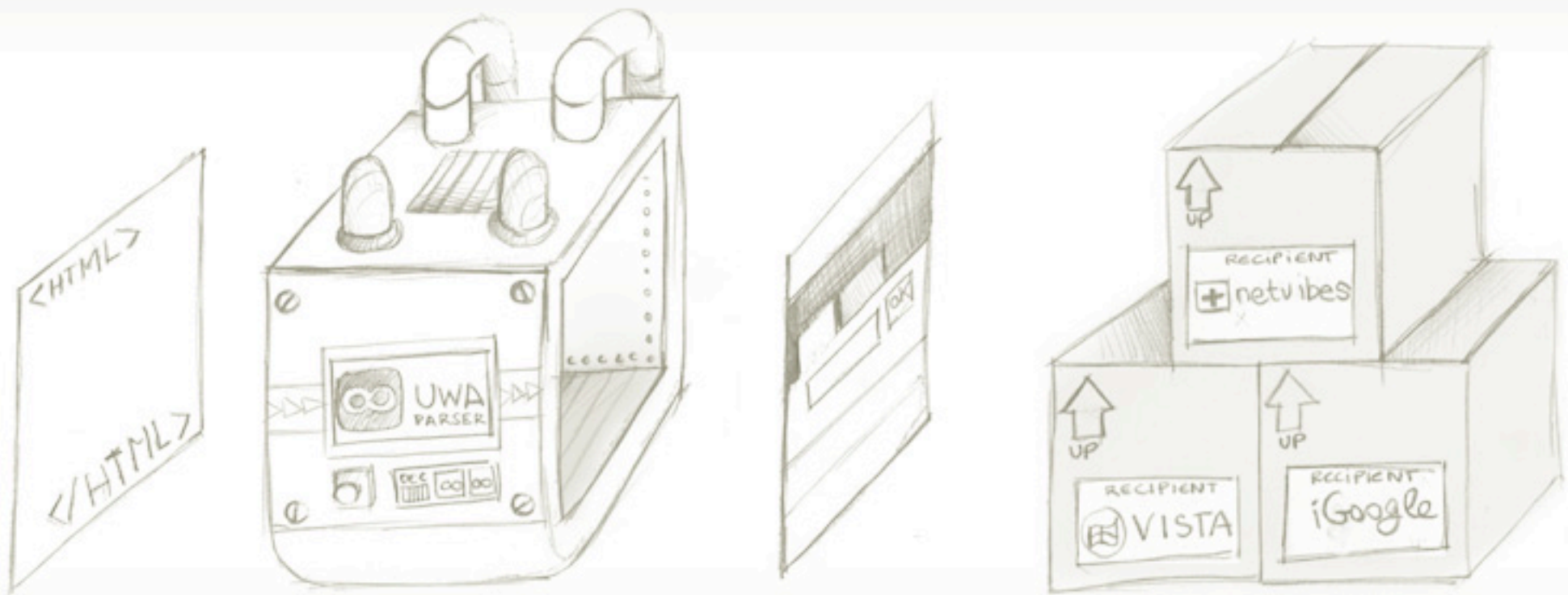


Netvibes Universal Widget API



Bártházi András - <http://dev.netvibes.com/>
Web Konferencia - 2008. április 25. szombat

Ez mi ez?

- Netvibes:
személyre szabható információs központ
- UWA:
univerzális formátum widgetek leírására, avagy
egyszer elkészíted, “minden” platformon
működik

Mit mutatok be?

- Miből áll egy UWA widget?
- Hogyan működik egy UWA widget?
- Egy konkrét widget lefejlesztése:
IndaFotó fotómegosztó szolgáltatás

Mik azok a widgetek?

- Kicsi, specializált programok
- Elérhetőek böngészőn keresztül:
Netvibes, iGoogle, Live.com, YourMinis...
- Elérhetőek valamilyen telepített motorral: Vista Gadgets, Apple Dashboard, Yahoo! Widgets, Opera...

Miért született meg az UWA?

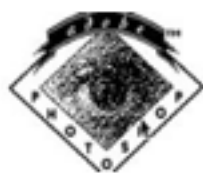
- Nincs de-facto standard
- Minden oldal, widget motor a saját formátumát használja
- A W3C specifikáció fejlesztés alatt, de nem használható még
- Most kell valami megoldás

Mit ígér az UWA?

- Minden népszerűbb platformon működik, az UWA modul bármilyen módosítása nélkül
- Ez ma: Netvibes, iGoogle, Live.com, Opera, Apple Dashboard, Windows Vista, Yahoo! Widgets, iPhone
- Hasonlít a MiniAPI-hoz, csak szigorúbb
- Könnyen tanulható, hiszen ismert, bizonyított standard eszközöket használ:
XHTML/XML, JavaScript/Ajax, CSS



Web | Gasztro | Marketing | Miner Blog



Webisztán: designerek

A Webszemé Készítsetek e legrégebbi - 1

MS Fejlesztői Portál: AJAX MaskedEditExtender - 6 hours ago

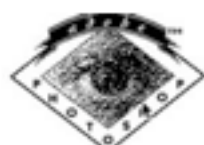
Ebben a cikkben a MaskedEdit r megismerkedni, illetve utána já

Pixelszabászat: Egyszerű r ellenőrzés - 7 hours ago

Egy új szolgáltatás, az inursite.com premise is simple; enter a few



Web | Gasztro | Marketing | Miner Blog



Webisztán: Retróverseny web designereknek - 5 hours ago

A Webszemét blog versenyt hirdetett: Készítsetek egy web oldalt a lehető

MS Fejlesztői Portál: AJAX Control Toolkit XVII. - MaskedEditExtender - 6 hours ago

Ebben a cikkben a MaskedEdit nevű illetve utána járunk annak is,...

Pixelszabászat: Egyszerű red/green frontend ellenőrzés - 7 hours ago

Egy új szolgáltatás, az inursite.com simple; enter a few of...

powered by [netvibes](#) uwa



Web | Gasztro | Marketing | Miner Blog



Webisztán: Retróverseny web designereknek - 5 hours ago

A Webszemét blog versenyt hirdetett: Készítsetek egy web oldalt a lehető legrégebbi - web oldal...

MS Fejlesztői Portál: AJAX Control Toolkit XVII. - MaskedEditExtender - 6 hours ago

Ebben a cikkben a MaskedEdit nevű extender-rel fogunk megismerkedni, illetve utána járunk annak is,...

Pixelszabászat: Egyszerű red/green frontend ellenőrzés - 7 hours ago

Egy új szolgáltatás, az inursite.com a következőket tudja: The premise is simple; enter a few of...

UWA widget

- Egy statikus XHTML fájl, valid XML
- UTF-8 kódolású
- Netvibes névtérrel egészül ki az XHTML:
xmlns:widget=“<http://www.netvibes.com/ns/>”

UWA widget

- Egy UWA widget:
 - HTML váz
 - meta leíró elemek
 - widget:preferences
 - stílus definíciók
 - script
 - alaphelyzeti HTML tartalom

HTML váz

- Mint egy weboldal:

```
1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "h
2 <html xmlns="http://www.w3.org/1999/xhtml" xmlns:widget="ht
3 <head>
4
5 <title>IndaFotó</title>
6
7 <style type="text/css">/*! [CDATA[*
8 /*]]>*/</style>
9
10 <script type="text/javascript">/*! [CDATA[
11 // ]]></script>
12
13 </head>
14 <body>
15 </body>
16 </html>
```

Meta leíró elemek

- általános HTML és egyedi elemek is:

```
7 <meta name="author" content="András Bárházi" />
8 <meta name="version" content="1.0" />
9
10 <meta name="apiVersion" content="1.0" />
11 <meta name="inline" content="true" />
12 <meta name="autoRefresh" content="20" />
13
14 <link rel="icon" type="image/x-icon" href="http://indafoto
15
16 <link rel="stylesheet" type="text/css" href="http://www.net
17 <script type="text/javascript" src="http://www.netvibes.com
```

widget:preferences

- egyedi, a widget beállítási lehetőségeit leíró UWA elemek:

```
19 <widget:preferences>
20   <preference name="title" type="text" label="Cím" defa
21   <preference name="tabs" type="text" label="Elemek" de
22   <preference name="nbTitles" type="range" label="Db" d
23 </widget:preferences>
```

stílus definíciók

- teljesen szokványos stílus leírók:

```
25 <style type="text/css"> /**/<br/>26     .nv-tabContent ul,<br/>27     .nv-tabContent ul li {<br/>28         display: block;<br/>29         margin: 5px;<br/>30         padding: 0;<br/>31         background: none;<br/>32     }<br/>33     .nv-tabContent ul li {<br/>34         background: #000;<br/>35         padding: 1px;<br/>36         float: left;<br/>37     }<br/>38 /*]]&gt;*/&lt;/style&gt;</pre></div>
```

script

- A konkrét működést leíró, megszokott JavaScript:

```
40 <script type="text/javascript">
41
42 var IndaFoto = {};
43
44 IndaFoto.data = [];
45
46 IndaFoto.display = function(feed) {
47
48     var tabContent = IndaFoto.tabs.getTabContent(IndaFoto.data.currentId);
49     if (!feed) {
50         tabContent.innerHTML = '&lt;p&gt;Nem létező azonosító, vagy egyéb hiba.&lt;/p&gt;';
51     } else {
52         tabContent.innerHTML = '';
53         var pager = new UWA.Controls.Pager({
54             module: widget,
55             limit: widget.getValue('nbTitles'),
56             offset: IndaFoto.data.offset,
57             dataArray: feed.items
58         });
59     }
60 }</pre></div>
```

alaphelyzeti HTML tartalom

- ez is teljesen szimpla:

```
145 </head>  
146 <body>  
147 <p>Loading ...</p>  
148 </body>  
149 </html>
```

Hogyan működik egy (UWA) widget?

- Információ beszerzése, kommunikáció
- Információ feldolgozása
- Információ megjelenítése

Információ beszerzése, kommunikáció

- http://dev.netvibes.com/doc/uwa_specification/using_javascript_and_ajax
 - UWA.Data.getFeed(url, callback)
 - UWA.Data.getXml(url, callback)
 - UWA.Data.getJson(url, callback)
 - UWA.Data.getText(url, callback)
 - UWA.Data.request(url, request)

Információ feldolgozása

- JavaScript alapokon, “semmi különös”

Információ megjelenítése

- A dokumentum HTML tartalmának manipulációja alapvetően DOM-mal:
 - `widget.body` = a widget befoglaló HTML eleme
 - `widget.createElement("div")`
 - `widget.body.appendChild(item)`
 - `widget.body.innerHTML = "Szöveg"; !!!`

Információ megjelenítése

- Template-ekkel, Control-okkal:
http://dev.netvibes.com/doc/uwa_templates
- Data grid, E-mail list, Feed list, Rich list, Thumbnailed list, Pager
- TabView
- Title Highlight

Események lekezelése

- http://dev.netvibes.com/doc/uwa_specification/using_javascript_and_ajax
- `widget.* = function() {}`
 - `onLoad, onRefresh, onResize, onSearch, onResetSearch, onKeyboardAction`

Tippek és trükkök

- Olvasd el a teljes doksit, dev.blogot az elejétől a végéig!
- Készülj fel a különböző platformokra, használj szabványos és platformfüggetlen JS-t, CSS-t
- Minél több dolog kerüljön a widgetbe (gyors betöltődés)
- Ne használj id-t, mert lehet, hogy egy oldalon többször is megjelenik egy widgeted!

Tippek és trükkök

- A függvényeidet tegyed egy a widget neve alapján elnevezett objektumba:
`var Widgetem = {};`
`Widgetem.display = function(){};`
- Kódolj angolul, ha segítséget kell kérned, jóval könnyebben fogják megérteni a kódod külföldiek

Tippek és trükkök

- Fejlessz élesben a Netvibes.com-on egy üres fülön.
- Fejlesztéskor inline, cache nélkül add hozzá az oldalhoz a widgetet.
- Amíg nem implementálsz a widget.onRefresh függvényt, a widget refresh gombja újratölti a kódod, nem kell a teljes oldalt újratölteni

IndaFotó widget

- Cél #1 - beállítást célzó üzenet megjelenítése

IndaFotó Edit



A widget tartalmát a Beállítások között tudod beállítani. Az "Elemek" értékhez IndaFotó felhasználók azonosítóját írhatod be, vesszővel elválasztva.
Például: **boogie,eszpee,blumi,darthwalk,tschoppy,dtamas**
Fedezd fel az [IndaFotó](#)t további azonosítókat illetően!

.showMessage()

- Létrehozunk egy (tetszőleges nevű) függvényt.
- `widget.body.innerHTML`-lel felépítjük az üzenetet

```
var IndaFoto = {};  
  
IndaFoto.showMessage = function() {  
    widget.body.innerHTML =  
        '<p><a href="http://indafoto.hu/boogie/image/85596-84fb74c7" target="_blank">' +  
        'Például: <strong>boogie,eszpee,blumi,darthwalk,tschoppy,dtamas</strong></p>' +  
        '<p>Fedezd fel az <a href="http://indafoto.hu/">IndaFotót</a> további azonosítókat' +  
        '}'
```

widget.onLoad

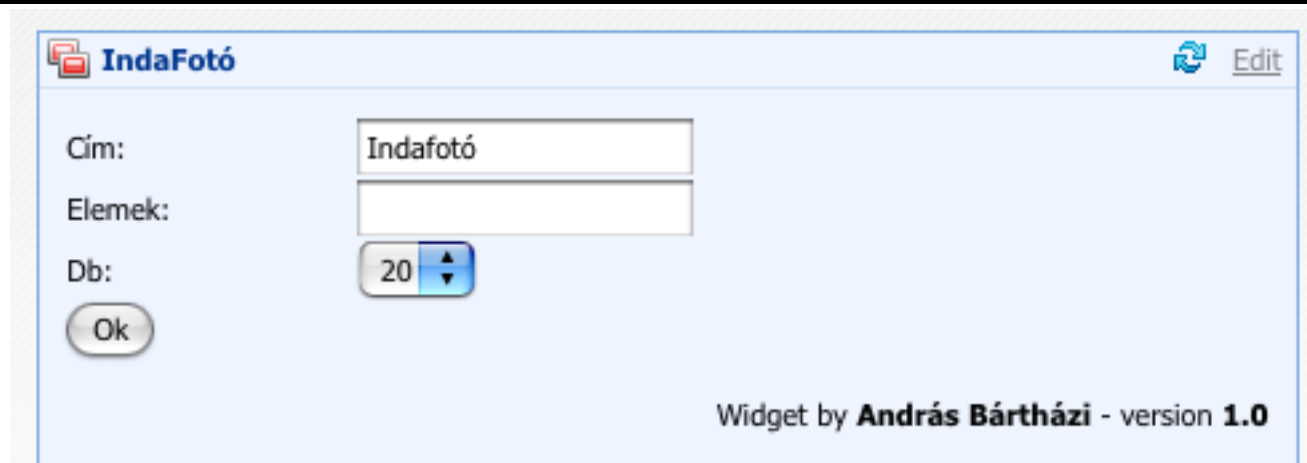
- Definiáljuk a widget.onLoad függvényt, és meghívjuk innen függvényünket:

```
// general widget code  
widget.onLoad = function() {  
    IndaFoto.showMessage();  
}
```

widget:preferences

- Létrehozunk egy widget:preferences blokkot a fájl elején:

```
<widget:preferences>  
  <preference name="title" type="text" label="Cím" defaultValue="Indafotó" />  
  <preference name="tabs" type="text" label="Elemek" defaultValue="" />  
  <preference name="nbTitles" type="range" label="Db" defaultValue="10"  
    step="1" min="1" max="30" />  
</widget:preferences>
```



The screenshot shows a dialog box titled "IndaFotó" with an "Edit" button in the top right corner. The dialog contains three preference fields:

- "Cím:" with a text input field containing "Indafotó".
- "Elemek:" with an empty text input field.
- "Db:" with a range slider set to 20.

An "Ok" button is located at the bottom left. At the bottom right, the text "Widget by András Bárházi - version 1.0" is displayed.

onLoad

- Ha már ki van töltve az “Elemek” rész, akkor nem az üzenetet kell megjeleníteni. Egyből dolgozzuk is fel és tároljuk le az “Elemek”-et.

```
widget.onLoad = function() {  
  
    if (widget.getValue('tabs') == '') {  
        IndaFoto.showMessage();  
    } else {  
        IndaFoto.ids = widget.getValue('tabs').split(/\s*,\s*/);  
        IndaFoto.initTabs();  
    }  
  
}
```

.initTabs()

- Írjuk meg az előbb meghívott fül inicializáló részt. Ennek is bármi lehetett volna a neve.

```
IndaFoto.initTabs = function() {
    widget.body.innerHTML = '';
    IndaFoto.tabs = new UWA.Controls.TabView({
        dataKey: "id",
        autohideDropdowns: true,
        softPadding: true
    });
    for (var i=0; i<IndaFoto.ids.length; i++) {
        IndaFoto.tabs.addTab(i, {text: IndaFoto.ids[i]});
    }
    IndaFoto.tabs.observe("activeTabChange",
        IndaFoto.onTabChange);
    IndaFoto.tabs.selectTab(0);
    IndaFoto.tabs.appendTo(widget.body);
}
```

.onTabChange()

- Ha változik a tab (vagy ha épp nekiindulunk), azt le kell kezelni:

```
IndaFoto.onTabChange = function(tabId, data) {
  IndaFoto.data.currentId = tabId;
  IndaFoto.data.offset = 0;
  widget.setTitle(widget.getValue('title') + ": " +
    IndaFoto.ids[tabId]);
  if (!widget.data[tabId]) {
    UWA.Data.getFeed("http://feed.indafoto.hu/" + IndaFoto.ids[tabId] +
      "/feed", function(data) {
        widget.data[tabId] = data;
        IndaFoto.display(data);
      })
  } else {
    IndaFoto.display(widget.data[tabId]);
  }
}
```

.display()

- És ha megvan a feed, akkor meg is kell jeleníteni azt az aktuális tab tartalmaként - melyik div is az?

```
IndaFoto.display = function(feed) {  
  
    var tabContent = IndaFoto.tabs.getTabContent(IndaFoto.data.currentId);  
    if (!feed) {  
        tabContent.innerHTML = '<p>Nem létező azonosító, vagy egyéb hiba.</p>'  
    } else {  
        tabContent.innerHTML = '';  
    }  
}
```


.display()

- Lapozás template:

```
var pager = new UWA.Controls.Pager({
  module: widget,
  limit: widget.getValue('nbTitles'),
  offset: IndaFoto.data.offset,
  dataArray: feed.items
});
pager.onChange = function(newOffset) {
  IndaFoto.data.offset = newOffset;
  IndaFoto.display(widget.data[IndaFoto.data.currentId]);
}
```

.display()

- Képek listája:

```
var feedList = widget.createElement('ul');
var j = 0;
for (var i = IndaFoto.data.offset; i < feed.items.length; i++) {

    if (j++>=widget.getValue('nbTitles')) break;

    var item = feed.items[i];
    var li = widget.createElement('li');
    var a = widget.createElement('a');
    a.href = item.link;
    a.target = '_blank';
    a.desc = item.content.stripTags().truncate(255);
    a.onmouseover = function() {
        UWA.Utils.setTooltip(this, this.desc, 250);
    }
    var img = widget.createElement('img');
    img.src = item.enclosures[0].url;
    a.appendChild(img);
    li.appendChild(a);
    feedList.appendChild(li);
}
```

.display()

- És a maradék, adjuk hozzá a tab tartalmához ezeket:

```
tabContent.appendChild(pager.getContent());  
tabContent.appendChild(feedList);  
tabContent.appendChild(pager.getContent());  
}  
}
```

widget.onRefresh()

- Kezeljük le a frissítést, esetünkben elég egyszerűen:

```
widget.onRefresh = widget.onLoad;
```

CSS

- hogy jól nézzenek ki a képek, balra floatoljuk a listájukat, és adunk egy fekete keretet mindegyiknek:

```
<style type="text/css">/*! [CDATA[*/
  .nv-tabContent ul,
  .nv-tabContent ul li {
    display: block;
    margin: 5px;
    padding: 0;
    background: none;
  }
  .nv-tabContent ul li {
    background: #000;
    padding: 1px;
    float: left;
  }
/*! ]]>*/</style>
```

További információk

- <http://dev.netvibes.com>
- <http://dev.netvibes.com/doc/>
- <http://dev.netvibes.com/forum/>
- <http://dev.netvibes.com/blog/>
- <http://eco.netvibes.com>
- Munka? :)
<http://dev.netvibes.com/doc/jobs>

További információk

- <http://barthazi.hu/jaccoter/indafoto/>
- <http://webakademia.hu/>
- <http://www.slideshare.net/carsonsystems/tariq-krim-fowa-feb-07>
- <http://www.slideshare.net/netvibes/netvibes-uwa-workshop-at-parisweb-2007>

Jövő

- Már most újdonságok a Ginger kapcsán:
 - `widget.readOnly` - ha publikus oldalon jelenik meg a widget, ne lehessen módosítani
 - `widget.addStar` - bármilyen link + szöveg pár hozzáadható a Netvibes hírfolyamhoz is
 - bővebben a dev.blogban:
<http://dev.netvibes.com/blog/>

Jövő

- Jön, jön, jön:
 - OpenSocial
 - és egyéb meglepetések

Köszönöm!

- Kérdések?
- Keress meg emailben:
 - andras kukac netvibes pont com