

# Webszerver a mobiltelefonon – a 3G tartalommegosztás új formája

DÉVAI ISTVÁN, BÁNYÁSZ GÁBOR, LEVENDOVSKY TIHAMÉR

BME Automatizálási és Alkalmazott Informatikai Tanszék  
{banyasz.gabor, tihamer}@aut.bme.hu

**Kulcsszavak:** 3G, mobil tartalommegosztás, mobil webszerver, mobil peer-to-peer alkalmazások

A 3G hálózatokhoz kapcsolódni képes mobiltelefonok elterjedésével egyre inkább jelentkezik az a felhasználói igény, hogy a készüléket ne csupán böngészésre, hanem tartalommegosztásra és más mobiltelefonokkal való IP alapú együttműködésre is lehessen használni. A fenti célok elérésére számos kutatási és fejlesztési irány létezik (peer-to-peer alkalmazások, azonnali üzenetküldők), azonban a vizsgálat tárgyát képező mobil webszerver koncepció rendkívül könnyű alkalmazásfejlesztést tesz lehetővé. Egy tesztkörnyezet összeállítása és teljesítménymérések elvégzése után a mobil webszerver életképes alternatívának látszik: szerver oldalon felhasználja a hatékony, objektumorientált Python szkriptnyelvet másrészt nem igényel speciális kliens oldali alkalmazást, csupán egy szabványos web böngészőt. Mindezek mellett a gyakorlatban is használható, akár 10-15 felhasználó egyidejű kiszolgálását lehetővé tevő teljesítményt nyújt. Ezért reményeink szerint a közeljövőben a mobil webszerver technológia számos új alkalmazással bővítheti a 3G készülékeket.

## 1. Bevezetés

A már hazánkban is széleskörűen elérhető 3G hálózatok nem egyszerűen gyorsabb internetelérést tesznek lehetővé, hanem teret engednek olyan új hálózati alkalmazások megszületésének, melyek kihasználják a mobiltelefon erőforrásait, a készülék változó földrajzi pozícióját és mobiltelefonok adhoc létrejövő és megszűnő csoportjait. Cikkünkben bemutatjuk, hogy miként lehet egy jelenleg is kapható 3G mobiltelefon-családon webszervert futtatni, és milyen módon lehet elérhetővé tenni ezt a szolgáltatást az internet nagyközönsége részére.

A rendszer technológiai ismertetése után részletezzük, milyen teljesítményre képes a jelenleg még fejlesztési stádiumban lévő szoftver, valamint milyen fejlesztési és tesztkörnyezet szükséges ahhoz, hogy új, web alapú mobiltelefon alkalmazásokat készítsünk. Végül bemutatjuk a jelenleg elérhető alkalmazásokat és valószínű jövőbeni fejlesztési irányokat.

## 2. Hardver és szoftver környezet

Az Apache alapítvány által fejlesztett webszerver [1] a szabadszoftver-mozgalom egyik jelentős eredménye, a

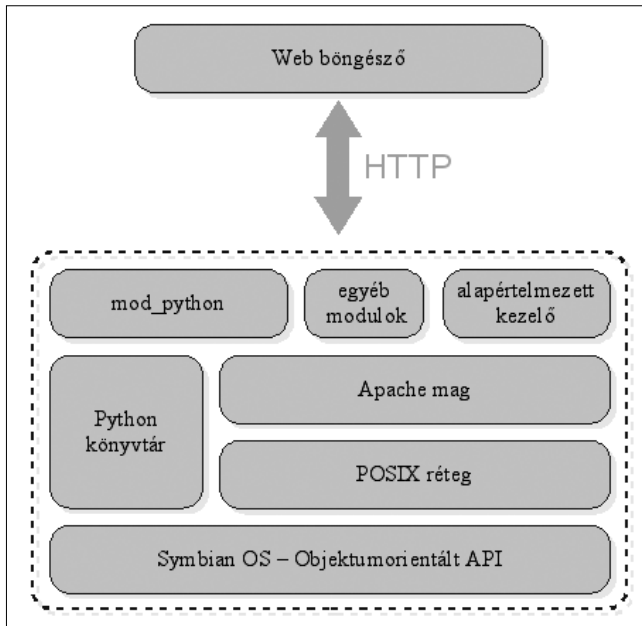
Netcraft adatai [2] szerint jelenleg az internetes oldalak közel 70%-át szolgálja ki. A C nyelvű forráskódot eddig is számos beágyazott rendszerre (többek közt ARM, PowerPC és MIPS) portolták, azonban erőforrásigénye valamint a mobil hálózati sávszéleség hiánya miatt mobiltelefonokon nem volt használható. Szintén probléma volt, hogy a telefonok beágyazott szoftvere nem nyújtotta az Apache számára szükséges POSIX [3] kompatibilis környezetet, ezért a portolás számos kód teljes újrainítását jelentett volna.

A Nokia 3G termékvonala azonban már mind memória, processzor teljesítmény valamint szoftver környezet tekintetében alkalmas arra, hogy az asztali gépeken elterjedt szerverszoftvert futtassuk rajta. Az 1. ábrán látható táblázatot tekintve láthatjuk a közelmúltban megjelent telefonok hardverspecifikációját.

További pozitívum a C és C++ nyelvű programozhatóság és a POSIX szabványt részben megvalósító API elérhetősége. Mivel mobil környezetben is szükséges dinamikus weboldalak generálása, ezért előnyös az is hogy, a Python szkriptnyelvnek [4] már létezik működő portja az S60-as sorozatú telefonokon.

1. ábra A felhasznált készülékek specifikációja

	Nokia 6630	Nokia N80
Processzor	220Mhz ARM architektúra	Texas Instruments OMAP 220Mhz
RAM	10 Megabyte	40 Megabyte
Háttértár	MMC kártya	miniSD kártya
Kijelző	176*208 pixel	352*416 pixel
Hálózat	WCDMA, Bluetooth	WCDMA, Wifi, Bluetooth
Operációs rendszer	Symbian OS V8.0s	Symbian OS V9.1



2. ábra A mobil webszerver szoftverkönyvtára

A Nokia Research Labs által elkészített, „Racoon” [5] nevű, Symbian operációs rendszeren [6] futó Apache port a 2. ábrán láthatóak szerint épül be a mobil szoftverkönyvtárba.

Az ábrán látható szoftverek szerepe:

- *Apache mag*

A webszerver funkciót ellátó kód, mely fogadja a webböngészők TCP/IP alapon érkező HTTP szabványú kéréseit, értelmezi azokat és visszaküldi az igényelt tartalmat (fájlt vagy valamilyen tartalmat előállító kód kimenetét).

- *Apache modulok*

Lehetséges az apache képességeit bővíteni különböző modulokkal. Például beállítható, hogy ha a `http://localhost/camera` oldalra hivatkoznak, akkor ezt ne az alapértelmezett kezelő válaszolja meg (amely a „camera” nevű fájl tartalmát próbálná elküldeni), hanem a `mod_camera` modul kezelje le. Ez a modul futáskor az apache processz részeként fog futni, és egyaránt elérheti az Apache belső API-ját és a Symbian API-t is. Konkrétan a fenti esetben a Symbian API segítségével elkészíthető egy fénykép a mobiltelefonba integrált kamerával, majd az Apache API segítségével továbbíthatjuk az elkészült képet a felhasználó felé.

- *Python csomag*

A Python kódot értelmezni és végrehajtani képes programkönyvtár valamint az azt parancssorban is elérhetővé tevő alkalmazás. A Python Symbian-változata meglepően sok szolgáltatását teszi elérhetővé a telefonnak: szkript fájlokból kezelhető a beépített kamera, készíthető grafikus kezelőfelület, kezdeményezhető hívások, elérhető a teljes telefonkönyv, naptár stb.

- *Mod\_python*

Olyan Apache modul, amely felhasználja a Python programkönyvtárat olyan módon, hogyha a felhasználó egy Python szkript fájlra hivatkozik, akkor a `mod_python` azt előbb futtatja és a futás eredményét küldi a felhasználó felé.

náló felé. Hasonlóan a PHP-hoz, a Python is lehetséges HTML kódba beágyazni és dinamikus webalkalmazások készítésére használni (Python Server Pages).

- *POSIX réteg*

A Symbian OS felhasználói interfészét POSIX kompatibilis programkönyvtárként nyújtó réteg (mely nem a Racoon, hanem a Symbian SDK része). Segítségével például lehetővé válhat, hogy szabvány BSD socketek segítségével valósítsunk meg TCP/IP kapcsolatot stb.

### 3. Az átjárhatóság problémája

A fenti alkalmazásokat lefordítva, majd telepítve azonnal futásra kész a webszerver, elindítása után a telefonon található webböngészőbe a `http://127.0.0.1` oldalon máris láthatóak a kiszolgált oldalak.

Amennyiben 3G kapcsolattal rendelkező mobiltelefonon fut a szoftver, úgy elvárható, hogy az oldalakat az Internet felől is meg lehessen tekinteni, ennek azonban az vet gátat, hogy a szolgáltató a telefonokat az internetes támadásoktól védetten, egy saját belső hálózaton tartja.

Ennek jellemzői általában:

- Belső IP címek (a tűzfal hálózati címfordítást – Network Address Translation, NAT – végez)
- A belső IP címek nem tartoznak egy-egy telefonhoz vagy előfizetéshez, azok időben mindig más és más felhasználókhöz vannak hozzárendelve.
- Még ha ismerjük is egy telefon belső IP címét, a szolgáltató tűzfala nem ad arra lehetőséget (pl. port-forward), hogy a telefontal közvetlen kapcsolatba lépünk TCP/IP segítségével.

Hazai viszonylatban a T-Mobile szolgáltatását tesztelve megerősíthetjük a fenti állításokat hozzátéve, hogy az alkalmazott tűzfal az egyik legszigorúbb, teljesen szimmetrikus [7] típusú. Ennek következménye, hogy a kapcsolatot mindig a telefonnak kell kezdeményezni, hiszen a teljesen szimmetrikus tűzfal nem fog az Internet felől ismeretlen TCP/IP kapcsolatokat a telefonok felé továbbítani. Mivel egy szerveralkalmazásról beszélünk, ezért ebből az is következik, hogy szükség van egy olyan, a tűzfal Internet felőli oldalán található számítógépre, mely az „átjáróállomás” szerepét vállalja és az Internet felől érkező kéréseket egy folyamatosan fenn tartott, a mobiltelefon által kezdeményezett kapcsolaton keresztül továbbítja a telefonkészülék felé.

Az így felépülő rendszer látható a 3. ábrán.

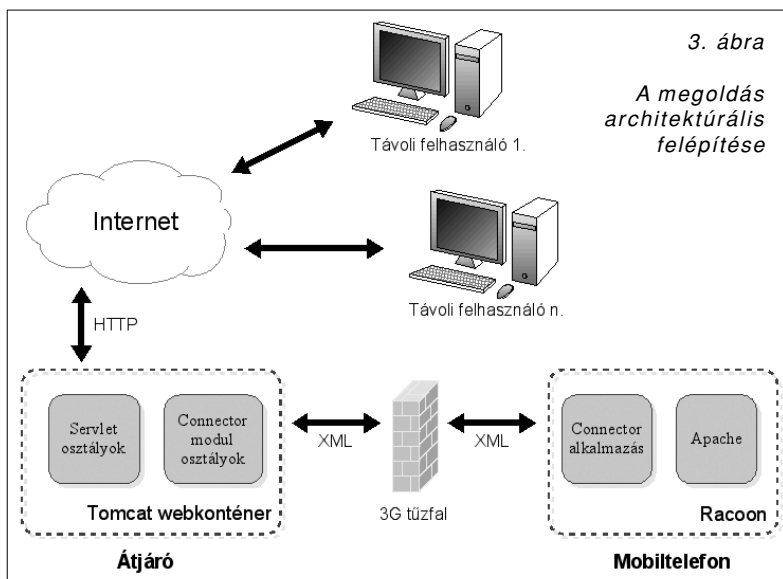
Az architektúra a következőkből áll:

- *Apache webszerver*

Az előzőekben bemutatott Symbianon futó Apache változat illetve a hozzá tartozó modulok, szkript interpreterok stb.

- *Connector Module*

Ez egy Symbian alkalmazás, mely indításkor TCP kapcsolatot kezdeményez az átjáróval majd ezt az átjáró segítségével a működés során folyamatosan fenn is tartja (az átjáró meghatározott időközönként életben tartó „ping” üzeneteket küld a telefon felé). Mivel ezen



socketen kívül még annyi párhuzamos TCP csatornát létesít (3) az átjáróval, amennyi párhuzamos letöltésre lehet számítani.

Ezután ha egy Internet felhasználó a `http://atjaro/~mobilfelhasznalo/` címet szeretné megtekinteni webböngészőjében, először egy szabványos HTTP kérést küld az átjáró felé, amelyet a Tomcat alkalmazásszerverben található webkiszolgáló fog lekezelni. Amennyiben „mobilfelhasznalo” névvel nincs mobiltelefon bejelentkezve, úgy egy „A felhasználó sajnos nincs online” tartalmú oldal jelenik meg a böngészőben.

Amennyiben olyan oldalt kérnek le, amelyhez bejelentkezett felhasználó tartozik, úgy az átjáró a már említett XML alapú protokoll segítségével eljuttatja a kérést (4) a mobiltelefonon futó Connector modulhoz. A modul ezután az XML üzenetből előállítja az eredeti kérést, lefuttatja azt a webserveren és a server választ szintén XML üzenetként az egyik adatcsatornán visszaküldi (5) az átjárónak, ami továbbítja az eredeti kérést feladó Internet felhasználónak.

kapcsolat kétirányú, az átjáró bármikor értesítheti a Connector Module-t, miszerint egy Internet felhasználó bizonyos lapokat szeretne lekérni a webservertől (így valósítható meg az átjárás a tűzfalon). A kommunikáció a Connector Module és az átjáró között egy speciális, XML alapú protokoll segítségével történik.

• **Átjáró**

Olyan Tomcat alkalmazásszerveren [8] futó webalkalmazás, mely a felhasználó szempontjából reverse-proxy-ként működik, azaz a kért weboldalt elkéri a mobiltelefonon futó webservertől, majd ezt a felhasználó számára transzparens módon továbbítja.

A megvalósítás egy kérés kiszolgálása esetén a 4. ábrán vázoltak szerint működik.

Amikor a mobiltelefonon elindítjuk a webservert, a Connector modul egy vezérlőkapcsolatot épít ki az átjáróval (1) TCP protokoll segítségével, XML üzenetek felhasználásával. Az üzenet tartalmazza a felhasználónevet, jelszót, az életbentartó ping üzenet (2) gyakoriságát, valamint azt, hogy a wezserver maximum hány párhuzamos kérés kiszolgálására képes. Ez utóbbi paraméter jelentősége, hogy a Connector modul a vezérlő

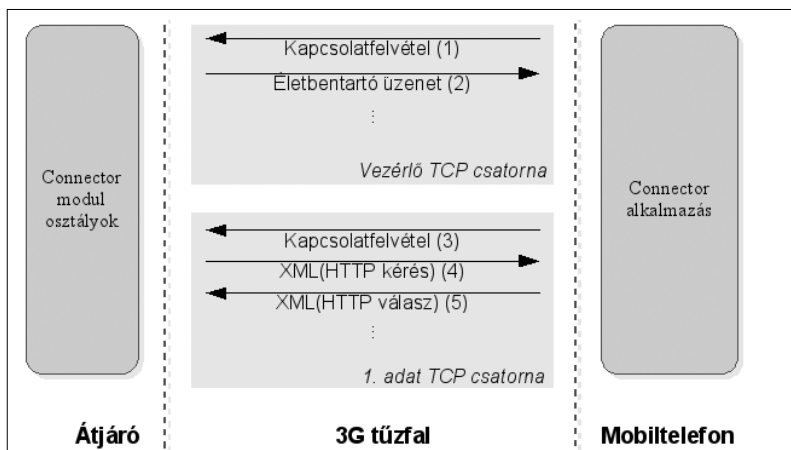
deti kérést, lefuttatja azt a webserveren és a server választ szintén XML üzenetként az egyik adatcsatornán visszaküldi (5) az átjárónak, ami továbbítja az eredeti kérést feladó Internet felhasználónak.

Első pillantásra az átjáró csupán egy kényszermegoldásnak tűnhet, amivel a 3G szolgáltató tűzfalán át lehet jutni, azonban a gyakorlat azt mutatja, hogy számos esetben kifejezetten hasznos egy ilyen közvetítő a telefon és az Internet böngészői között:

- Mivel a telefon erőforrásai igen korlátozottak (lásd később a teljesítményértékeket), mindenképp szükséges a tartalom gyorsítótárazása (caching). Ez csökkenti a processzor és a hálózat terhelését a telefonon, s mivel természetesen a 3G hálózati forgalmat a telefon tulajdonosa fizeti, így csökkenti a mobil wezserver fenntartásának költségeit is.
- Az átjáró segítségével lehetséges a felhasználók földrajzi pozíció szerinti csoportosítása.
- Amennyiben a telefon pillanatnyilag nem kapcsolódik az átjáróhoz, a már említett weboldal jeleníthető meg (vagy a legutolsó tárolt állapot).

**4. Mérési környezet és eredmények**

4. ábra HTTP kérések és válaszok átvitele a 3G tűzfalon

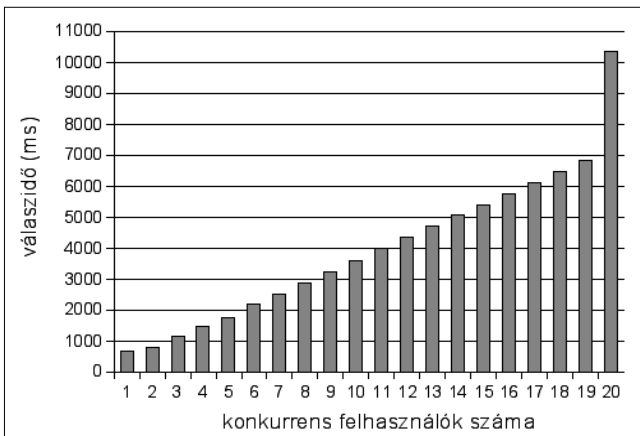


Olyan tesztkörnyezet létrehozása volt célunk, melynek segítségével megismerhető a rendszer működése, teljesítménymérések végezhetők rajta, valamint akár új mobil web alkalmazások fejlesztésére is lehetőség nyílik. Természetesen egy olyan élő SIM kártyával, amelyen engedélyezett a 3G szolgáltatás, azonnal használatba lehet venni a Racoon alkalmazáscsomagot (akár saját átjárót üzemeltetve akár a Nokia által biztosítottat használva), azonban teszteléshez, alkalmazásfejlesztéshez jóval kényelmesebb közvetlen TCP/IP kapcsolatot kialakítani a telefon és a helyileg üzemeltetett átjáró között.

Az általunk használt Nokia 6630 és N80 típusok esetén ezt a következőképpen tudtuk megoldani:

- Nokia 6630-as esetén Bluetooth technológia segítségével lehetséges a közvetlen IP kapcsolat létrehozása. Mivel a gyári alapszoftverek nem támogatják ezt, egy közösségi fejlesztésű GnuBox nevű [9] alkalmazást kell telepíteni, ami a telefonon egy olyan speciális kapcsolatot hoz létre, mely a telefonon futó alkalmazások számára egy vonalkapcsolt internetelésnek látszik. Az internet vagy helyi hálózat felé egy Bluetooth adapterrel felszerelt Windowsos vagy Linuxos PC használható (Windows esetén a közvetlen kábelkapcsolat funkció használatával, Linux-nál PPP szervert futtatva).

- N80-as telefon esetén jóval egyszerűbb a helyzet, hiszen a készülék képes Wifi hálózatokra kapcsolódni, így egy helyileg üzemeltetett Access Point segítségével máris elérhetővé válik a helyi hálózat számára. (Mivel teljesítménymérést is szeretnénk volna végezni, nem egy külön hardveres AP-t, hanem egy PCMCIA WLAN kártya segítségével szoftveres AP-t üzemeltettünk, ezáltal elkerülve, hogy a helyi hálózati forgalom stb. befolyásolja a mérést. Ilyen szoftveres Access Point létrehozására Linux esetén a MadWifi, illetve HostAP driverek [10] segítségével van mód.)



5. ábra A webszerver teljesítménye 3G hálózaton, N80-as készülék esetén

A tesztkörnyezet összeállítása után mindkét telefont közvetlen, Bluetooth, illetve Wifi általi IP kapcsolat segítségével teszteltük. A teljesítménymérést a httpperf [11] illetve apachebenchmark [1] szoftverekkel végeztük. Mivel a helyi hálózat nem jelentett akadályt a sávszélességben, a mobiltelefon folyamatosan képes volt több, mint 10 kérés/sec teljesítményre is (ekkor egy kérés átlagos válaszára kb. 250-300 ms). Az igazi próbát természetesen a 3G hálózaton való tesztelés jelentette, mely esetén a mérési eredmény az 5. ábrán látható.

A mérés során fontos észrevétel volt, hogy a párhuzamos adatkapcsolatok számát célszerű 3-4 között tartani, mivel ez alatt a telefon még nem nyújt optimális teljesítményt (sokat kell a hálózati átvitelre, az átjáró válaszára stb. várni). Ezen érték felett pedig a lineárisnál némileg jobban nő a válaszára, ezáltal a párhuzamos kéréseket célszerű inkább az átjáró által sorosítani, mint újabb adatkapcsolatokat nyitni.

## 5. Jelenlegi alkalmazások

A Raccoon programcsomag már jelenleg is számos mobil webalkalmazást tartalmaz:

- *Képkészítés igény szerint*

Ha egy távoli böngésző a mobil weboldalon a megfelelő gombra kattint, a telefonon a 6. ábrán látható üzenetablak jelenik meg. Amennyiben engedélyezzük a képkészítést, akkor a távoli felhasználó azonnal láthatja az elkészült képet a böngészőben.



6. ábra

*Képkészítés megerősítése a mobiltelefonon*

- *Helyinformáció lekérdezése*

Mobile Network Code, Mobile Country Code, Local Area Code, Cell ID azonosítók.

- *Kik tartózkodnak a közelben?*

Bluetooth hálózaton a telefon képes észlelni, ha egy másik Bluetooth eszköz van a közelben. Amennyiben az eszköz szintén mobil webszerveret futtató telefon, azonnal lekérdezhető a honlap címe és így oldalról oldalra vándorolva megismerhető a környék összes mobil weboldala.

- *Azonnali üzenet küldése*

A 7. ábrán látható üzenetablakban azonnal megjelenik az elküldött üzenet, hasonlóan egy SMS-hez. (Lehetőség van arra is, hogy a postafiókban hagyjunk üzenetet).



7. ábra

*Azonnali üzenet fogadása*

• *Hozzáférhető a telefonon tárolt adatok*

A weben megtekinthető (jelszó ellenében) a teljes telefonkönyv, rendelkezésre állnak a könyvjelzők, a naptár valamint a szöveges üzenetek. (Megléhetősen hasznos szolgáltatás abban az esetben, a ha otthon felejtjük a telefont, vagy ha kényelmesen szeretnénk új bejegyzéseket felvenni.)

## 6. A jövő alkalmazásai

Véleményünk szerint a mobil webalkalmazások egyik legfontosabb feladata, hogy elősegítsék az utóbbi években egyre erőteljesebben jelentkező igényt a mobil tartalomkezelésre. Egyrészt várhatóan számos szolgáltatás építhető a tartalom létrehozása, szűrése és csoportosítása köré: a felhasználók a mobilkészülék segítségével fogják létrehozni a tartalmat (képek, telefonkönyv, naptár, szöveges és hangüzenetek, napló, videoklipek stb.), és ezt mobil webtechnológia segítségével juttatják el az átjáró, majd a felhasználó felé. Másrészt mobil környezetben is megkezdődik az egyes webes tartalmak konvergenciája: a mobil weboldalunkon nincsenek szolgáltató által megszabott korlátozások, bárki küldhet nekünk szöveges, hang vagy videóüzeneteket és válaszképpen mi is bármilyen tartalmat szolgáltatathatunk az oldal segítségével.

Például egy sítúra után nem a felhasználó küldi el barátainak a mobiltelefonnal készített képeit, hanem azok automatikusan letölthetőek lesznek a telefon mobil weboldaláról (mobsite), akár még a sítúra alatt. Ha esetleg valaki elfelejt digitális kamerát vinni a sítúrára, akkor sincs gond, mivel lekérhető a közelben tartózkodó mobil webszervert futtató felhasználók (mobile user – muser) listája, és letölthetők képek a többiek oldaláról. Amennyiben a felhasználó eltéved a sítúráját keresve, beállítható, hogy ha a rendszer egy másik, magyarul beszélő síelőt észlel a közelben, akkor riasszon, így azonnali üzenetben segítség kérhető.

A 3GPP streaming (IP alapú mobil video) már jelenleg is realitás a legtöbb 3G telefon esetében, azonban nemsokára már az ellenkező irányban is lehetséges lesz mindez (hasonlóan a jelenleg is működő videotelefonáláshoz, azzal a különbséggel, hogy a nézőközönség akár a teljes világháló lehet). A lokalitás alkalmazása is számos lehetőséget rejt magában: lekérhető lesz a közelünkben tartózkodó érdekes pontok (Point of Interest, POI) listája és az azokhoz tartozó weboldal (például ha szükségünk van egy könyvre, az összes közelben levő könyvesbolt weboldala elérhető). Azonban nem csupán mi találhatunk meg másokat, hanem mások is minket: egy rendezvény helyszínén tartózkodó tömeg részére a rendezvény szervezője elküldheti a rendezvény képeit, szintén a helyinformációra alapozva.

Természetesen ilyen fokú földrajzi pontossághoz már nem elég csupán a mobiltelefonról lekérdezhető cellainformáció: vagy a szolgáltató oldaláról szükséges támogató vagy a földrajzi pozícióra más módon érdekes következtetni. Például ha mobilkészülékek egy cso-

portja eléri egymást Bluetooth segítségével, akkor az átjáró következtethet arra, hogy a telefonok egymástól maximum 10-100 méterre találhatók. A rendszernek ezenkívül számos ipari alkalmazása is lehetséges, például rendkívül könnyen fejleszthető felületet biztosít különböző mérési adatok gyors publikálására és az átjárón történő aggregálására.

Összefoglalásképpen a mobil weboldalak fő ismérve, hogy interaktív kommunikáció folytatható a weboldal látogatói és tulajdonosa között, kihasználható a földrajzi pozíció, a tartalom a látogatók igénye szerint folyamatosan változtatható. A mobil weboldalak hasznos tulajdonsága, hogy nem csupán explicit hivatkozások vannak az oldalak között, az átjáró számos szempont szerint automatikusan tudja csoportosítani azokat (például a két felhasználónak vannak közös ismerősei a készülék telefonkönyvében).

Szintén rendkívül érdekes alkalmazási területeket ad az, hogy a rendszer segítségével tetszőleges formátumú, szöveges, hang, kép, videóállományok cserélhetők a készülék és a világháló között. Ennek eredménye lehet, hogy számos, jelenleg a mobil hálózat által nyújtott szolgáltatás (pl. rövid szöveges üzenetek) a közeljövőben már IP alapon vehető igénybe, a cikkünkben bemutatott rendszer ehhez visz egy újabb lépéssel közelebb.

## Irodalom

- [1] Apache webszerver  
<http://httpd.apache.org/>
- [2] NetCraft – webszerver statisztika  
[http://news.netcraft.com/archives/web\\_server\\_survey.html](http://news.netcraft.com/archives/web_server_survey.html)
- [3] POSIX –  
IEEE Portable Applications Standards Committee  
<http://www.pasc.org/plato/>
- [4] Python szkriptnyelv  
<http://www.python.org/>
- [5] Racoon szoftvercsomag  
<http://research.nokia.com/research/projects/mobile-web-server/index.html>
- [6] Jo Stichbury,  
Symbian OS Explained: Effective C++ Programming for Smartphones (Symbian Press)  
<http://www.symbian.com/>
- [7] Saikat Guha, Paul Francis,  
Characterization and Measurement of TCP Traversal through NATs and Firewalls, Cornell University  
<http://nutss.gforge.cis.cornell.edu/pub/imc05-tcpnat/>
- [8] Tomcat alkalmazáserver  
<http://tomcat.apache.org/>
- [9] GnuBox  
<http://gnubox.dnsalias.org/gnubox/>
- [10] MadWifi és HostAP eszközmeghajtók  
<http://madwifi.org/>
- [11] Httpperf – webkiszolgálók teljesítménymérő szoftvere  
<http://www.hpl.hp.com/research/linux/httpperf/>