

# Szemantikus protokollt alkalmazó mobil Peer-to-Peer kliensszoftver

FORSTNER BERTALAN, KELÉNYI IMRE

BME Automatizálási és Alkalmazott Informatikai Tanszék  
{forstner.bertalan, kelenyi.imre}@aut.bme.hu

**Kulcsszavak:** szemantikus Peer-to-Peer információ-visszakeresés, mobil szoftverfejlesztés

*A korszerű okostelefonok egyre inkább hasonlítanak a kisméretű számítógépekre. Felhasználóik már nemcsak telefonálni szeretnének segítségükkel, hanem zenehallgatásra, fényképek készítésére és megosztására, információk olvasására is zsebkészüléküket használják. A szélessávú vezeték nélküli távközlés elterjedésével olyan hálózati alkalmazások költöznek át mobil készülékekre, amelyek eddig csak asztali környezetben voltak megszokottak. Sokoldalú felhasználási lehetőségei miatt egy ilyen nagyon lényeges szoftver lehet a teljesen elosztott hálózatra épülő információmegosztó alkalmazás, vagyis a Peer-to-Peer (P2P) szoftver. A fajlagosan magasabb távközlési költségek ugyanakkor azt igénylik, hogy az elosztott információ-visszakeresés során fejlett, kisebb adatforgalmat generáló, ugyanakkor a kliensek hálózati fluktuációját toleráló protokollt használjunk. Ebben a cikkben bemutatjuk az első, Symbian mobil operációs rendszerre fejlesztett Peer-to-Peer kliensszoftvert, valamint az adott célra kifejlesztett szemantikus protokollt.*

## 1. Bevezetés

Az információk, különböző fájlok, dokumentumok felkutatása a kezdetektől fogva nagy kihívása az informatika tudományának. A hálózatba kötött számítógépek és más eszközök (például mobiltelefonok) számának növekedésével a feladat csak tovább bonyolódott. A központosított megoldások mellett hamar kialakultak a teljesen elosztott információ-visszakereső alkalmazások, a különböző Peer-to-Peer rendszerek. Ezek lényege, hogy az egyes résztvevők azonos szerepkörrel vesznek részt a kommunikációban. Tipikus példa lehet egy kép- vagy zenemegosztó szoftver, ahol az egyes felhasználók egymás számára elérhetővé teszik saját tartalmaikat. A Peer-to-Peer alkalmazás összekapcsolja ezeket a felhasználókat és irányítja a fájlok keresését.

Az okostelefonok, vagyis smartphone-ok lehetővé teszik, hogy az említett fájljainkat (képek, zenék) magunkkal vigyük. Ezek a készülékek ugyanis – egy megszokott mobiltelefontól eltérően – fejlett funkcionalitással, komoly számítási teljesítménnyel, nagyobb háttértár kapacitással és viszonylag nagy kijelzővel rendelkeznek. Fejlesztői szempontból azonban még fontosabb, hogy a rendelkezésre álló környezet esetén komplexebb saját programok futtatására is alkalmasak lehetnek. A kizárólag a Java mobilra optimalizált verziójának futtatókörnyezetét (J2ME [1]) tartalmazó készülékektől eltérően az okostelefonok komolyabb operációs rendszerén natív alkalmazásokat is képesek futtatni. Néhány ismertebb smartphone operációs rendszer (Microsoft Windows CE [2], Linux egyes mobil disztribúciói [3]) előtt magasan őrzi piacvezető szerepét [4] a Symbian rendszere [5].

A Symbian céget 1998 júniusában alapította a Nokia, a Motorola, a Psion, valamint az Ericsson, és 1999-ben csatlakozott hozzájuk a Matsushita, majd 2002-ben a Sony-Ericsson és a Siemens. Ezen cégek célja egy

mobil eszközökre szánt és a mobil környezetből adódó összes körülményt és követelményt figyelembe vevő operációs rendszer megalkotása volt. A kezdeményezés sikerét jelzi az egyre újabb verziókkal megjelenő operációs rendszer töretlen népszerűsége mind a készülégyártók, mind a felhasználók körében. Legfőbb hátrányként talán a konkurens technológiákhoz képest bonyolultabb programozói felületet lehet felhozni. Mindemellert mégis ez a platform tűnt a legcélszerűbbnek arra, hogy egy olyan összetettebb alkalmazást, mint a Peer-to-Peer kliens, mobil környezetbe ültessünk.

Önmagukban az okostelefonok leginkább a különböző típusú tartalmak tetszőleges helyen történő megjelenítésére, lejátszására nyújtanak a felhasználó számára csábító lehetőséget. A technológia egy másik irányának fejlődése, vagyis a mobil szélessávú hálózatok elterjedése teszi azonban lehetővé, hogy ezeket az információkat az asztali számítógépes környezetben megszokott módon osszunk meg másokkal. Miután a le- és feltöltési sebesség már nem okoz gondot, érdemessé vált elgondolkodnunk a Peer-to-Peer alkalmazások mobil környezetbe ültetéséről.

A kérdés vizsgálata során számos olyan körülményre bukkantunk, amelyek az eredeti fájlmegosztó protokollok hatékonyabb változatainak kifejlesztését tették szükségessé. Mivel a mobil kommunikáció költségei magasabbak a vezeték nélküli kommunikációénál, a keresés adatforgalmát minimalizálni kell. A korszerű protokollok szemantikus úton próbálják a találati arányt növelni, az eredetileg véletlenszerű Peer-to-Peer hálózatot különböző megfontolások alapján strukturálni. Ezek a megfontolások azonban nem feltétlenül életképesek mobil környezetben, hiszen a mobil felhasználók fluktuációja viszonylag nagy, vagyis az egyes kliensek gyakran csatlakoznak, illetve hagyják el a hálózatot. Ennek oka nemcsak a hálózati forgalom, és így a költségek, illetve az

energiafogyasztás csökkentésének igényére vezethető vissza, hanem a mobiltelefonok sajátosságaiból is adódhatnak (például télerő hiánya, akkumulátor lemerülése stb.) Emiatt egy olyan szemantikus hálózati réteget dolgoztunk ki, amely egy adott csomópont (vagyis mobiltelefon) társhálózati kapcsolatait a lehető leggyorsabban igyekszik a felhasználó érdeklődési területei szerint átalakítani, és amely számára egy-egy csomópont kiesése a többi kliens keresési hatékonyságát, illetve eredményességét tekintve nem kritikus.

## 2. A Gnutella bemutatása

A Peer-to-Peer egyáltalán nem nevezhető új technológiának, hiszen a kezdetekben maga az Internetet is egy olyan decentralizált, Peer-to-Peer elven működő hálózatnak indult, melynek célja a számítási teljesítmény megosztása volt. Ez idővel, a fejlesztések hatására, fokozatosan megváltozott és eltolódott a kliens/szerver architektúra irányába, azonban a mai tendenciák azt mutatják, hogy kezdenek visszatérni az eredeti elképzelésekhez.

A P2P hálózatoknak számos felhasználási módja ismert, ilyenek többek között az információmegosztás és az információ visszakeresése. Az ilyen típusú hálózati protokollok közül az egyik legkiforrottabb a Gnutella. A protokollra épülő hálózatok célja a résztvevők erőforrásainak megosztása, amely gyakorlatilag bármi lehet (például kriptográfiai kulcsok), azonban a továbbiakban csak fájlok megosztásával foglalkozunk.

Fontos megjegyezni, hogy a modern Gnutella verziók már nem tisztán P2P rendszerek, a hálózatban a centralizált és a decentralizált topológia egyaránt megfigyelhető: bizonyos csomópontok kiemelt szerepkörrel rendelkeznek. Az ilyen csomópontok kinevezése azonban továbbra is felsőbb hatóság irányítása nélkül történik, így megmarad a függetlenség. A hálózat biztonsága szempontjából is nagyon előnyös a decentralizált topológia, hisz nem lehet egy, a központi szerver iránt intézett, támadással lebénítani a hálózatot. Továbbá fontos kiemelni, hogy a Gnutella felépítése moduláris: könnyen illeszthetők a protokollhoz kiegészítések, így módon a szemantikus réteget támogató funkciókat is be tudtuk építeni a rendszerbe.

A Gnutella kliens első feladata, hogy további Gnutella csomópontokat találjon, majd ezekkel kommunikálva, felépítse a kapcsolatot a hálózathoz. Minden csomópont további csomópontokkal áll kapcsolatban (ezek száma általában 2-3), és így a kliens rajtuk keresztül éri el a hálózat többi részét. A Gnutella hálózat forgalma legnagyobb részt keresési kérésekből és válaszokból, illetve hálózati adminisztrációs üzenetekből áll.

Az üzenetek továbbküldése (útvonalválasztás) egy igen egyszerű sémára épül. A Gnutella hálózaton a szervereknek (servent – server and client, egy Gnutella csomópont) nincsen „címe” (ennek, ebben a kontextusban, nem is lenne értelme), így egy üzenet célba juttatásának egyetlen módja valamilyen elárasztásos módszer.

Amikor egy servent megkap egy üzenetet, akkor feldolgozza annak tartalmát, majd továbbküldi a szomszédos csomópontoknak. Minden üzenet fejléce tartalmaz egy értéket (TTL – Time To Live), amely meghatározza, hogy az üzenet „milyen messzire” kerülhet a küldőjétől. Minden alkalommal, mikor egy servent továbbküldi az üzenetet, csökkenti annak TTL értékét, így szabályozható, hogy az üzenetek a hálózat milyen mélységéig jussanak el. Ha a TTL eléri a 0-át, a servent nem küldi tovább az üzenetet. Ez különösen fontos lehet például egy keresési kérésnél, ahol minél több csomópont kapja meg az üzenetet, annál több válasza számíthatunk. Természetesen nagyobb TTL esetén a hálózat terheltsége jelentősen megnőhet, így a legtöbb kliens maximum 7-8 értékű TTL-el küldi az üzeneteket.

## 3. A Symella terve

A mobil készülékek alacsony számítási teljesítménye és korlátozott háttértár-kapacitása egészen eddig nem tette lehetővé egy teljes értékű P2P kliens megvalósítását, ám az okostelefonok elterjedésével a helyzet megváltozott. Elsődleges célunk egy olyan hordozható készülékeken futó Gnutella kliens megalkotása volt, mely alkalmazkodik a mobil eszközök sajátosságaihoz, bármilyen más platformon futó Gnutella klienssel képes kapcsolatok kiépítésére, továbbá modularitása révén könnyen bővíthető: az új kutatási eredmények, implementálás után, a gyakorlatban is tesztelhetőek.

A kliens alapjául a jelenleg elérhető legkiforrottabb és egyben legelterjedtebb mobil operációs rendszert, a már említett Symbian OS-t választottuk, az alkalmazás ezért kapta a Symella nevet (a Symbian és a Gnutella szavak összekapcsolásából).

Bár a szoftver implementálása előtt már számos PC-s Gnutella kliens is a rendelkezésünkre állt, ezek a verziók egészen más szempontokat vettek figyelembe, mint amikre egy mobil alkalmazás készítésekor figyelni kell, így a legtöbb problémára új megoldást kellett találnunk. A kliens képes egy fájl egyszerre több szálon is letölteni, mely jelentősen felgyorsítja az adatcserét, hisz a sávszélesség megoszlik a csomópontok között.

## 4. A SemPeer protokoll

A bevezetésben említett módon a Gnutella protokollra egy szemantikus réteget terveztünk, amelyet SemPeer-nek neveztünk el. Ennek a rétegnek a működését egy, a mindennapi életben megfigyelt jelenségből lehet a legkönnyebben megérteni. Eszerint a valós világban az emberek kapcsolatai különböző szálak mentén szövődnek [6]. Ha például valakinek a munkája a francia irodalommal kapcsolatos, akkor szakmai kapcsolatai is ilyen érdeklődésű emberek köréből alakul ki, hiszen a témával kapcsolatos kérdéseit leginkább ezek a kollégái tudják megválaszolni, nem pedig véletlenszerűen választott emberek. Hasonló megfontolással, ha valaki

a 19. századi klasszikus zenét kedveli, akkor – egy fájl-megosztó hálózatot tekintve – hasonló fájlokkal rendelkező felhasználók gyűjteményében tud a legnagyobb valószínűséggel ilyen zenét fellelni.

Az új szemantikus réteg tehát ez alapján, oly módon igyekszik egy adott csomópont kapcsolatait átalakítani, hogy az egyes szomszédok a szóban forgó csomópont különböző érdeklődési területei közül valamelyikkel rendelkezzenek. Megvalósított esettanulmányunkban a zenei stílusok képviselik ezt a fajta megkülönböztető dimenziót.

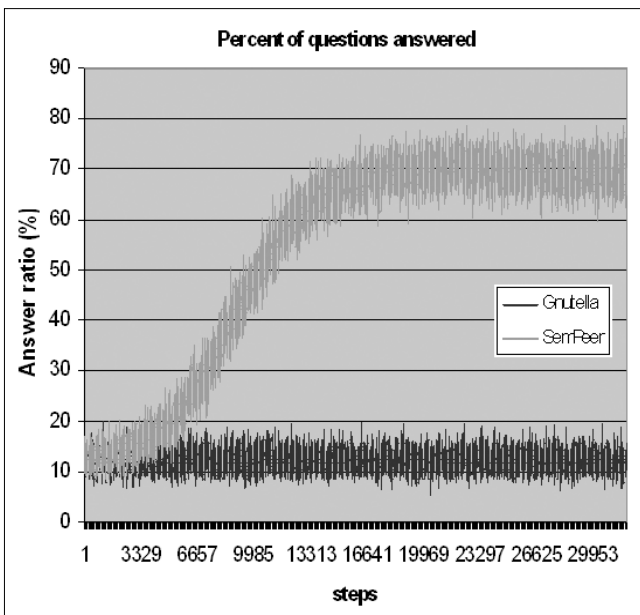
Az érdeklődési területek összehasonlítása a csomópontok által tárolt adatok alapján felépített [7,8] szemantikus profilok egybevetésével zajlik [9].

A protokoll mohó olyan értelemben, hogy a nagyobb hasonlóságot mutató kapcsolatokat részesíti előnyben, vagyis véges számú szemantikus kapcsolatai közül a kisebb hasonlóságú csomópontokat lecseréli a keresés során talált, nagyobb hasonlóságot felmutató csomópontokra. Mindehhez egyre mélyebb, konkrétabb szinten végzi az összehasonlítást. Ezáltal az idők folyamán egyre előnyösebb szomszédos csomópontokat képes kiválasztani, amellett, hogy már a kezdeti stádiumban is képes a véletlenszerű kapcsolatoknál jobb csomópontokat találni.

A protokoll az alap Gnutellához hasonlóan továbbra is strukturálatlan hálózatot alakít ki abban az értelemben, hogy a csomópontok a kereséshez, információ szerzéséhez lokális információkat használnak csak fel, így egy csomópont kiesése (ami mobil környezetben gyakorinak számít) nem okoz fennakadást a továbbra is megosztott adatok felkutatásában, elérhetőségében.

Egy további nehézség, amellyel a SemPeer protokoll kialakítása közben meg kellett küzdeni, a csoportképződés volt. Az algoritmus mohósága miatt ugyanis

1. ábra  
A Gnutella és a SemPeer protokoll találati arányainak összehasonlítása



gyakran kialakulhatott az a helyzet, hogy egy keresőkérdés olyan csomóponthoz jutott vissza, amelyik már feldolgozta azt, hiszen az azonos érdeklődésű szomszédok kapcsolatai is logikusan az effajta profilú kliensek közül kerültek ki.

Ez nemcsak fölösleges hálózati forgalmat jelentett, hanem az egy keresőkérdés által elért csomópontok számának jelentős esését, így a keresett adat fellelési esélyének jelentős csökkenését is okozta. Ezt a hátrányt előnnyé kovácsolva ezért olyan mechanizmust is a protokollba kellett építenünk, amely a szemantikus réteg megfelelő, csoportképződést messzemenőig támogató topológiáját biztosította [10].

## 5. A protokoll teljesítménye

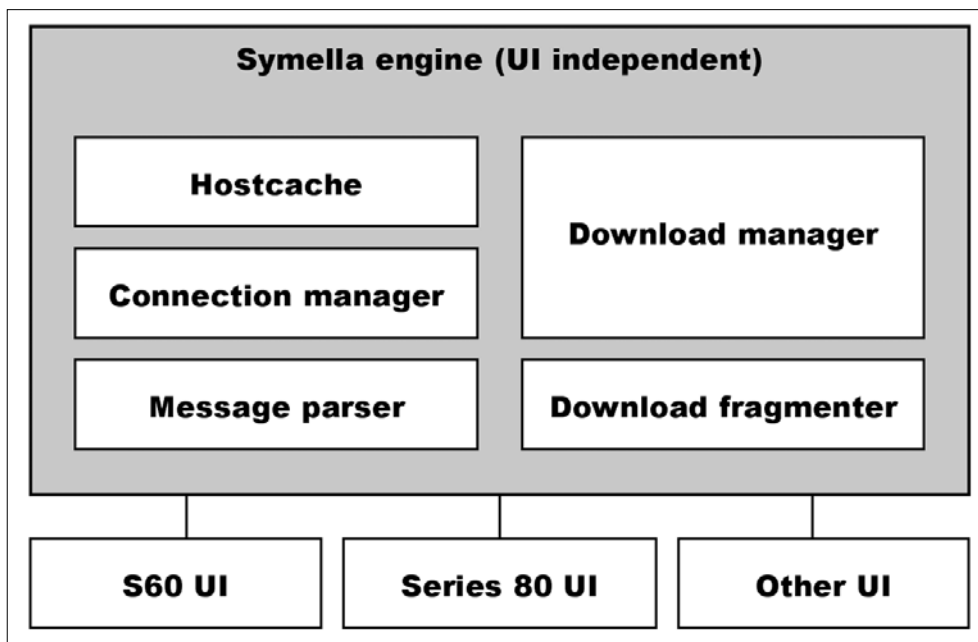
Ahhoz, hogy a protokoll teljesítményét megvizsgáljuk, egy szimulációs környezethez [11] illesztettük a saját protokoll-bővítvényünket. A méréseket különböző tipikus paraméterekkel végezve megállapíthattuk, hogy milyen esetekben számíthatunk jelentős teljesítménymelkedésre a protokollunk alkalmazásától.

Érdekesképpen vizsgáljunk meg egy esetet, amikor a kliensek erősen egy témakörre koncentrált érdeklődési területtel rendelkeznek, és viszonylag kevés fájl képesek tárolni. Ez tipikusan a kis memóriakapacitású okos mobil eszközök esete, ahol még egy albumnyi zenei anyag sem fér el a memóriakártyán. A szimulált hálózatban 16.000 csomópont található 4-7 kapcsolattal, amelyek átlagosan 5 fájl képesek tárolni a saját zenei ízlésükből. Tíz főbb zenei ízlést különböztettünk meg. Meglepően jó eredményt kapunk, amennyiben egy csomópont a saját érdeklődési területének megfelelő zenéket keres: átlag 5-15 keresőkérdés eredményének visszaérkezésével a szemantikus protokoll olyan réteget hozott létre a véletlenszerű kapcsolódású alap Peer-to-Peer hálózat fölött, amelyben a találati arány az alap protokoll hétszeresét is elérte (0.11-ről 0.69-re emelkedett) (1. ábra).

Természetesen amennyiben a keresőkérdések, vagy a tárolt dokumentumok tematikája nem ennyire homogén, akkor az átlagos találati arány ennél csak kisebb mértékben növekszik, azonban a teljesítménynövekedés minden esetben megfigyelhető.

Érdekes még megemlítenünk, hogy a szemantikus protokoll által elérhető elméleti maximális találati arány kiszámítására egy matematikai modellt alkottunk. A SemPeer protokoll finomítása során elértük, hogy ezt az elméleti határt minél inkább megközelítsük.

A [12] referencia alatt található cikkünkben összevetettük a modell által jelzett találati arányt a szimulált eredményekkel különböző helyzetekben (különböző érdeklődési területtel rendelkező csomópontok, érdeklődési körbe nem tartozó tárolt dokumentumokkal, keresőkérdésekkel stb.) Azt tapasztaltuk, hogy a kialakított protokoll a dokumentumok rendelkezésre állása esetén képes volt minimális eltéréssel megközelíteni az ideális értéket.



2. ábra A Symella architektúrája

### 6. A Symella architektúrája

A következőkben bemutatjuk azt az alkalmazás-architektúrát, amely a fent részletezett felismert követelményeknek és a Symbian operációs rendszer elveinek leginkább megfelel (2. ábra). A Symella motorja felelős a kapcsolatok kiépítéséért és karbantartásáért, a keresési kérések feldolgozásáért, illetve a letöltések felügyeletéért. A Gnutella csomópontok adatait egy intelligens Host cache tárolja, mely folyamatosan gyűjti és osztályozza a különböző forrásokból beérkező címeket, hogy a Connection manager mindig a leoptimálisabb kapcsolatokat tudja kiépíteni. Az osztályozás alapja a kapcsolatok kiépítéséhez szükséges idő, ezen kívül a hoszszú ideje aktív, megbízható kapcsolatokat külön eltárolja a rendszer. Az üzenetek feldolgozásáért és küldéséért a Message parser felelős.

A letöltésnél a sávszélesség minél jobb kihasználása kiemelt fontosságú szempont volt a tervezésénél, ezért is döntöttünk a többszálú letöltés támogatása mellett. Ennek lényege, hogy a letöltés elkezdése előtt a fájlt logikailag több apró darabra bontjuk, ezt végzi el a Download fragmenter (daraboló) modul, majd ezeket a darabokat több különböző csomóponttól, egymással párhuzamosan töltjük le. A letöltési alrendszer másik fel-

adata a keresési találatok begyűjtése és a megegyező fájlokra hivatkozó válaszok összevonása.

A Symbian csak alapját képezi a mai készülékek operációs rendszerének. Erre ráépül még a felhasználói felület (User Interface, UI) rétege, amely készülék-típusonként nagyon különböző lehet. Egy széles képernyővel és teljes QWERTY billentyűzettel rendelkező Nokia Communicator felhasználói felületét teljesen máshogyan kell programozni, mint mondjuk egy Sony Ericsson P900-ét. Mivel egyik rendszer mellett sem szerettük volna elkö-

telezi magunkat, így a fejlesztés folyamán fontos szempont volt a platformfüggő és -független részek szétválasztása. Az alkalmazás jelenleg két UI-t is támogat: a legelterjedtebb S60-at, melyet a hagyományos mobiltelefon-billentyűzettel és kis képernyővel rendelkező készülékek használnak, illetve a Series 80-at, mely szabványa teljes alfanumerikus billentyűzetet és nagyobb kijelzőt ír elő.

A megfelelő architektúrának köszönhetően a mobil szoftver válaszüzeje igen jó lett. A 3. ábrán összehasonlítottuk a Symella fontosabb válaszüzejeit az elterjedtebb rendszerek megfelelő adataival.

### 7. Összefoglalás

Az általunk tervezett és implementált szoftver az első Symbian operációs rendszer alatt futó, natív nyelven írt Peer-to-Peer fájlcsere alkalmazás. Jelenlegi formájában már most több ezren használják szerte a világon, a regisztrált letöltések száma pedig folyamatosan növekszik.

A tartalomszolgáltatók szemszögéből nézve, egy decentralizált, elosztott módon működő információcsere rendszer sok lehetséges jövőbeli felhasználási módot rejt. Az adatok szabadon tárolhatók asztali, illetve mobil eszközökön, a mobil kliens az egységes protokoll ré-

3. ábra A Symella és néhány ismertebb Gnutella kliens válaszüzejének összehasonlítása

	Symella	BearShare	LimeWire	Gnucleus
<b>Első induláskor a kapcsolódásig eltelt idő</b>	10 mp	25 mp	7 mp	79 mp
<b>További indulásokkor ugyanez</b>	3 mp	7 mp	3 mp	15 mp
<b>Első keresésre a találatok beérkezési ideje</b>	45 mp	35 mp	30 mp	29 mp
<b>További keresésekre ugyanez</b>	40 mp	26 mp	30 mp	30 mp

vén képes bármelyikkel kapcsolatba lépni. A mobil készülékek erőforrás-kapacitása jelentősen kisebb, mint egy asztali számítógépé, de sokkal nagyobb számban állnak rendelkezésre, így a redundancia révén nagy megbízhatóságú hálózatok építhetők ki belőlük.

A kliens jelenleg is megbízhatóan, és a mobilkészülékek képességeihez mérten viszonylag gyorsan működik, továbbfejlesztési lehetőségekben azonban nincsen hiány. A Gnutella szabvány számos olyan funkciót tartalmaz, amelyek implementálása tovább növelheti a kliens működésének hatékonyságát. Ilyen például az üzenetek tartalmának tömörítése, amely így jóval kisebb üzenetméretet és nagyobb szabad sáv szélességet eredményez. A szemantikus réteg továbbfejlesztésével pedig nagyságrenddel csökkenthetjük a keresések lefutási idejét.

Az alapszoftver forráskódját a GNU GPL licenc keretében bárki szabadon használhatja [13]. A kutatási eredmények összefoglalása után a SemPeer protokollkiegészítés szintén publikusan elérhető lesz.

### Köszönetnyilvánítás

A cikkben bemutatott projekt egyes részei a Mobil Innovációs Központ támogatásával valósultak meg.

### Irodalom

- [1] J2ME hivatalos weboldal,  
<http://java.sun.com/javame/index.jsp>
- [2] Windows CE hivatalos weboldal,  
<http://msdn.microsoft.com/embedded/windowsce/default.aspx>
- [3] Montavista Mobilinux weboldal,  
<http://www.mvista.com/products/mobilinux/>
- [4] Gartner Research  
(Ben Wood, Carolina Milanesi, Ann Liang, Hugues J. De La Vergne, Tuong Huy Nguyen, Kobita Desai, Sauk-Hun Song, Nahoko Mitsuyama):  
Market Share: Mobile Terminals, Worldwide, 1Q05 (2005. május 24.)
- [5] Symbian hivatalos weboldala,  
<http://www.symbian.com>

- [6] Mark S. Granovetter,  
"The Strength of Weak Ties",  
American Journal of Sociology, 78 (1973),  
pp.1360–1380.
- [7] H. Assadi,  
"Construction of a Regional Ontology from Text and Its Use within a Documentary System  
International Conference on Formal Ontology and Information Systems", FOIS-98,  
IOS Press, Amsterdam (WebDB-2000),  
Springer-Verlag, Berlin, 2000,  
pp.60–71.
- [8] J.-U. Kietz, A. Maedche and R. Volz,  
"Semi-Automatic Ontology Acquisition from a Corporate Intranet",  
Proc. Learning Language in Logic Workshop (LLL),  
ACL, New Brunswick, N.J., 2000,  
pp.31–43.
- [9] Resnik, P.,  
"Semantic similarity in taxonomy:  
An information-based measure and its application  
problems of ambiguity in natural language",  
Journal of Artificial Intelligence Research, 11 (1999),  
pp.95–130.
- [10] B. Forstner, R. Kereskényi, Dr. H. Charaf,  
"Eliminating Clustering in the Propagation Tree of Semantic Peer-to-Peer Networks",  
IASTED Conference on Parallel and Distributed Computing And Networks, Innsbruck, Austria,  
February 14-16, 2006.
- [11] B. Forstner, G. Csúcs, K. Marossy, H. Charaf,  
"Evaluating Performance of Peer-To-Peer Protocols with an Advanced Simulator",  
Conference on Parallel And Distributed Computing And Networks, Innsbruck, Austria,  
February 15-17, 2005.
- [12] B. Forstner,  
"An Analytic Model for Peer-to-Peer Systems with Semantic Overlay Network", AACS'06 Workshop,  
2006, Budapest, Hungary
- [13] A Symella elérhetősége:  
<http://symella.aut.bme.hu>