

MAIPAN – Middleware személyi hálózatokban működő alkalmazások összekapcsolására

Rónai Miklós Aurél, Fodor Kristóf, Biczók Gergely, Turányi Zoltán, Valkó András
Ericsson Magyarország Kft, Traffic Lab
{Miklos.Ronai, Kristof.Fodor, Gergely.Biczok, Zoltan.Turanyi,
Andras.Valko}@ericsson.com

Kivonat

Cikkünkben a MAIPAN elnevezésű middleware megoldást javasoljuk dinamikusan változó személyi hálózatok (Personal Area Network – PAN) létrehozására. Ez a middleware eltakarja a PAN szétdaraboltságát és a hálózatot egységesen, egyetlen számítógépnek mutatja az alkalmazások felé. A megoldás lehetővé teszi a PAN-ban több eszközön működő elosztott alkalmazások összekapcsolását, valamint a PAN konfiguráció változásakor az adatfolyamok transzparens átírányítását.

A javasolt middleware a különböző eszközökön futó alkalmazások által nyújtott szolgáltatások ki és bemeneti csatlakozási pontjait virtuális csatornák létrehozásával köti össze. A csatornák szükség esetén újrakonfigurálhatók, ha a hálózat felépítése vagy a felhasználói igények megváltoznak. Az irodalomban található megközelítésekkel szemben mi egy olyan megoldást ajánlunk, ami egy erős és intuitív jogosultság-kezelő mechanizmussal rendelkezik, valamint rugalmas és átruházható kapcsolatrendszer kezelést is tartalmaz.

1 Bevezető

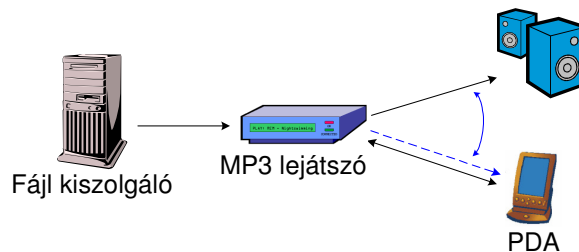
A vezeték nélküli terminálok, mint például okostelefonok, digitális személyi asszisztensek (Personal Digital Assistant – PDA) és laptopok növekvő száma miatt egyre inkább szükség van arra, hogy személyi hálózatokat egyszerű módon lehessen felállítani, valamint be- és újrakonfigurálni.

Cikkünkben azzal foglalkozunk, hogy hogyan lehet személyi hálózatokban egy dinamikusan változtatható, de egységes számítástechnikai környezetet létrehozni, mely kiterjed az egymással összekötött vezetékes és vezeték nélküli, valamint fix és mobil eszközökre. A MAIPAN middleware-t javasoljuk – ami az alkalmazások alatt és a hálózati réteg fölött helyezkedik el –, mely megoldás eltakarja a személyi hálózatban működő különálló eszközöket, és az ezeken futó alkalmazások képességeit úgy mutatja, mintha azok mind ugyanazon a számítógépen lennének elérhetőek. Ezzel egy olyan egységes programozási platformot nyújt, amely lehetővé teszi a személyi hálózatok egyszerű felépítését és a PAN-ban futó alkalmazások dinamikus összekapcsolását, valamint szétválasztását. A megoldás alkalmazás programozói interfésze (API) lehetővé teszi, hogy úgy készítsünk alkalmazásokat a PAN hálózatra, hogy nem kell törődnünk a különböző PAN konfigurációkkal és azok változásaival. Az alkalmazásfejlesztők feltételezhetik bizonyos funkciók (pl. hangszóró) jelenlétét, de figyelmen kívül hagyhatják azt, hogy ezen funkciókat egy bizonyos eszközön futó egyetlen alkalmazás vagy több eszközön futó több alkalmazás halmaza nyújtja. A programozóknak csak a middleware-be kell regisztrálniuk alkalmazási ki- és bemeneteit, és nem kell törődniük

azzal, hogy milyen eszközök vagy alkalmazások fognak ezekhez a csatlakozási pontokhoz kapcsolódni, valamint a programjaikat használni.

A bemutatásra kerülő middleware jogosultság, valamint rugalmas és átruházható kapcsolatrendszer kezelést is tartalmaz. A middleware-ben van néhány intelligens funkció, mely hatékonyabbá teszi az ember számítógép kapcsolatot (Human Computer Interaction – HCI). Elméletben bármely szolgáltatás felderítési, hálózati vagy adatkapcsolati megoldás használható a MAIPAN-nel.

Az 1. ábrán PAN alkalmazásra látató egy példa. Tegyük fel, hogy egy digitális személyi asszisztens tulajdonosa, akinek a készülékén a MAIPAN fut, belép egy szobába ahol hangszórók vannak a sarkokban, melyek szintén MAIPAN-t futtatnak. Ha a felhasználó úgy dönt, hogy zenét szeretne hallgatni, de se mp3 fájljai, se mp3 lejátszója nincsen, akkor bekapcsolva a PDA-ját lekérdezheti, hogy milyen eszközök és szolgáltatások érhetők el a szobában. Ekkor először is – függetlenül MAIPAN-tól – a PDA hálózati rétege kapcsolatokat hoz létre a szobában lévő eszközökkel és a szolgáltatás felderítő protokollja pedig körbekérdez, hogy milyen szolgáltatások működnek a környéken. Ezt követően a felhasználó a MAIPAN vezérlővel bekonfigurálhat egy olyan személyi hálózatot, amely a PDA-t, a sarkokban lévő hangszórókat, az mp3 lejátszót és a hálózaton elérhető fájlservert tartalmazza. MAIPAN létrehozza a szükséges virtuális kapcsolatokat, azaz a fájlservert az mp3 lejátszó bemenetére, az mp3 lejátszó kimenetét a hangszórókra és az mp3 lejátszó vezérlő bemenetét a PDA-ra köti. Ettől kezdve az fájlserveren ki tudja választani azokat a zenéket, amiket hallgatni szeretne és a PDA-jával utasítani tudja az mp3 lejátszót, hogy a zenét a hangszórókon szólaltassa meg. Az ábrán a nyilak az adatáramlás irányát mutatják.



1. ábra: Zenehallgatás

A cikk további részében először a 2. szakaszban áttekintjük az irodalmat majd a 3. szakaszban bemutatjuk a MAIPAN middleware-t, végezetül a 4. szakaszban összefoglaljuk munkánkat.

2 Irodalomfeldolgozás

Mark Weiser a 90-es évek elején alkotta meg [1][2] a mindenütt jelenlévő számítástechnika (ubiquitous computing) koncepcióját. Tíz évvel később, Satyanarayanan Weiser víziójának kihívásaival foglalkozott [3], és bevezette a mindent átható számítástechnika (pervasive computing) fogalmát, mely többek között az elosztott rendszerek, a mobil számítástechnika és az okos terek (smart spaces) keveréke [26].

A middleware-ek lényeges részei mind a mindenütt jelenlévő, mind a mobil számítástechnikának. Mascolo és társai [4] többek között azt feszegetik, hogy a hagyományos middleware rendszerek (mint pl. a CORBA [5]), miért nem annyira

megfelelőek mobil környezetben, valamint arról írnak, hogy hogyan is kell egy mobil számítástechnikai middleware-t megtervezni. A mindent átható számítástechnikai rendszereknek másik kulcs kérdése a biztonság. Chandrasiri és társai [6] a személyi biztonsági területek használatát javasolják – mely területek a felhasználók személyes eszközeiből állnak –, valamint különféle biztonsági aspektusokat elemeznek a PAN-okkal kapcsolatban. Manapság számos projekt foglalkozik ezekkel a kutatási területekkel, ezek közül az alábbiakban kiemelünk néhányat. A területről és az alábbi projektek közül néhányról korábban már bővebben is írtunk [26][27].

Az AURA projekt [7] célja, hogy minden felhasználónak kialakítson egy láthatatlan számítástechnikai aurát. A GAIA [8] projekt egy middleware infrastruktúrát ajánl, hogy okos tereket lehessen létrehozni. Az Oxygen projekt [9] nagyon intelligens, felhasználó barát, könnyen használható mobil eszközök tervezésével foglalkozik. A Portolano projekt [10] adaptív felhasználói interfészek létrehozására fókuszál. A projekt keretében a one.world [11] architektúrát tervezték meg, mely mindenütt jelenlévő számítástechnikai alkalmazások készítését támogatja. Ilyen alkalmazások készítésének elősegítésére az extrovert-Gadgets projekt [12] különféle architektúrákat vizsgált meg és a GAS-OS middleware használatát javasolja szenzorok és működtető szerkezetek (actuator) vezérlésére. A 2WEAR projekt [13] viselhető (wearable) személyi rendszereket analizált, amelyek dinamikusan rakhatók össze különböző eszközök összekapcsolásával. A Cortex projekt [14] olyan új típusú alkalmazásokkal foglalkozik, melyek emberi beavatkozás nélkül, önállóan is tudnak működni. Az EasyLiving projekt [15] célja intelligens otthoni és irodai környezet kialakítása. A Speakeasy megközelítés [16] az alkalmazások közötti minimális interfészek definiálására koncentrál mobil kód és mobil ügynökök használatával. A MobiDesk [17] megoldás egy virtuális számítástechnikai asztal használatát javasolja. A virtuális eszköz létrehozására irányuló koncepciók [18][19][27] a felhasználó környezetében minden autonóm eszközt egy nagy virtuális egységgé alakítanak át.

Hasonlóképpen néhány fent említett megoldáshoz, a MAIPAN a teljes személyi hálózatot az alkalmazások felé egyetlen eszköznek mutatja. A MAIPAN-nal a felhasználók egyszerűen hozhatnak létre és konfigurálhatnak újra személyi hálózatokat. A MAIPAN rendszer alapötleteit angolul a [20]-as cikkben publikáltuk.

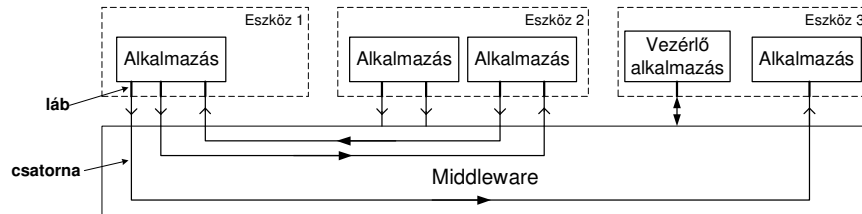
3 A MAIPAN platform

3.1 Alapötletek és definíciók

A MAIPAN *eszközöket, alkalmazásokat és szolgáltatásokat* különböztet meg. Az „eszköz” szó a fizikai eszközt jelenti, amin az „alkalmazás” fut, ami pedig nem más mint az a szoftver amelyik az adott „szolgáltatást” felajánlja. Ezeket az absztrakciókat például az mp3 lejátszó esetében alkalmazva azt mondhatjuk, hogy a „fizikai eszközön” fut az „mp3 lejátszó alkalmazás” ami az „mp3 lejátszó szolgáltatást” nyújtja. A különbségtétel az alkalmazás és a szolgáltatás között azért szükséges, mert szolgáltatást nem csak szoftver nyújthat, hanem hardver is (pl. egér).

A MAIPAN három koncepción alapszik (2. ábra): *láb, csatorna és kapcsolattrendszer*. A szolgáltatások bemeneteit és kimeneteit nevezzük lábaknak – az elnevezést az integrált áramkörök világából kölcsönöztük. A lábak az alkalmazások kapcsolódási pontjai a middleware felé, így a middleware az alkalmazásokat a PAN-ban

be- és kimeneti lábak csoportjainak látja. A lábak típus szerint is megkülönböztethetőek; a típus a láb által generált, illetve elnyelt adat, azaz az adott alkalmazás által kezelt információ fajtáját mutatja meg (pl. egér mozgás, billentyű leütés). A meglévő típusok mellé – szükség szerint – bármikor új típus definiálható. A PAN vezérlő alkalmazás a felelős azért, hogy csak a egyező típusú és megfelelő irányú lábakat lehessen összekötni.



2. ábra: MAIPAN kapcsolatrendszer

A lábak közötti kommunikációhoz a middleware pont-pont kapcsolatokat, azaz csatornákat épít ki. A csatornák azon halmazát, amely egy adott PAN szolgáltatás használatához szükséges, kapcsolatrendszernek nevezzük. Az alkalmazások nem tudnak a csatornákról, sem a kapcsolatrendszerekről, csak saját lábaikról. Például az 1. ábrán a következő csatornák alkotják az mp3 lejátszó kapcsolatrendszerét: van egy csatorna a fájlserver és az mp3 lejátszó, egy másik az mp3 lejátszó és a hangszórók, és egy további csatorna az mp3 lejátszó és a PDA között.

3.2 Biztonság és hozzáférési jogosultság kezelés

A biztonsági és jogosultság kezelési funkciókat eszköz szinten definiáltuk. Ez azt jelenti, hogy a szolgáltatásokhoz (azaz az adott alkalmazás lábaihoz) való hozzáférés az eszközöknek van engedélyezve vagy tiltva. Így ha egy eszköz jogot kap egy szolgáltatás használatára, akkor minden alkalmazás, amely az adott eszközön fut jogosult lesz hozzáférni az adott szolgáltatáshoz. Ez az egyszerűsítés akkor engedhető meg, ha feltételezzük, hogy a PAN-ban kis eszközök vannak, melyek csak egy-két egyszerű szolgáltatást nyújtanak (pl. egér, mp3 lejátszó). Ebben az esetben fölösleges egy bonyolultabb eljárás alkalmazása, ahol a hozzáférést az eszközökön futó alkalmazásoknak egyesével kellene megadni.

A kapcsolatrendszereket a vezérlő entitás szerepét játszó eszközön futó vezérlő alkalmazás tudja felépíteni és újrakonfigurálni. A vezérlő entitásnak kell ellenőriznie a hozzáférési jogosultságok meglétét. Ha ezek hiányoznak, akkor a szolgáltatást nyújtó eszköztől meg kell kérdeznie, hogy a felhasználó használhatja-e az adott szolgáltatást. Központi vezérlő entitás lehet például egy PDA, amelynek elegendő számítástechnikai kapacitása van ahhoz, hogy menedzselni tudjon egy teljes személyi hálózatot. A PAN-ban szereplő minden más eszközt *résztevőnek* nevezünk. Speciális esetekben a résztvevők delegálhatják a hozzáférés engedélyezés döntési jogkörüket más eszközöknek, amelyeket *menedzsereknek* hívunk.

Itt most nem megyünk részletekbe a hitelesítés és engedélyezéssel kapcsolatban, viszont a [6]-ban található megoldások a MAIPAN-nal is alkalmazhatóak a biztonságos kommunikáció megvalósítására.

3.3 Kapcsolatrendszerek áthelyezése

A PAN-ban legalább egy eszköznek kell lennie, mely a vezérlő alkalmazás szerepét játssza. Abban az esetben, ha ez az eszköz eltűnik (mert kikapcsolják, vagy a felhasználó távozik), akkor az általa vezérelt összes kapcsolatrendszer automatikusan lebontódik. Ahhoz, hogy a kapcsolatrendszereket a vezérlő eszköz eltűnése után is fenntarthassuk, MAIPAN lehetőséget ad a működő kapcsolatrendszerek egyik vezérlő alkalmazásról a másikkra való átvitelére.

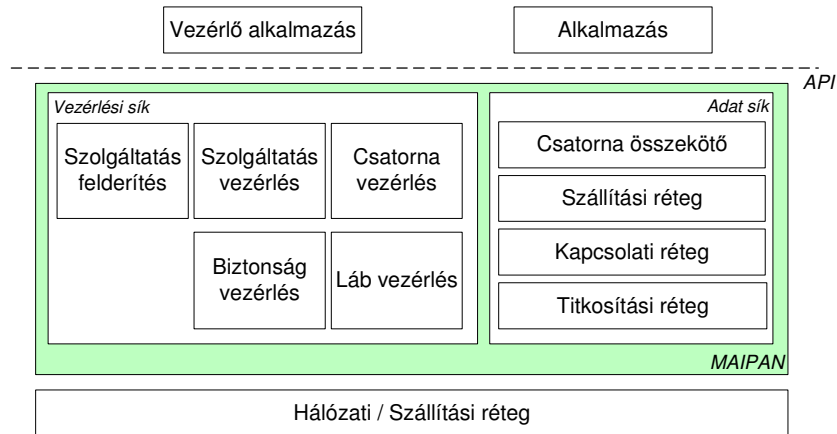
Például, hogy zenét tudjanak hallgatni egy tárgyalóteremben, az egyik felhasználó egy mp3 lejátszó kapcsolatrendszert konfigurál be. Így az adott felhasználó eszköze lesz a vezérlő entitása a kapcsolatrendszernek. Ha a felhasználó el akarja hagyni a szobát, akkor a MAIPAN-nel közli, hogy a kapcsolatrendszert vigye át egy másik vezérlő entitásra, ami például lehet egy másik felhasználó PDA-ja.

3.4 A kapcsolatrendszerek újrakonfigurálása

Abban az esetben, ha egy a PAN-ban résztvevő eszköz eltűnik (pl. a felhasználó elhagyja a termet, vagy az eszköz akkumulátora lemerül), akkor azok a csatornák, melyekben az eszköz részt vett automatikusan lebontódnak és a kapcsolatrendszert újra kell konfigurálni, ha a továbbiakban is használni akarják. Az első lépésben MAIPAN értesíti az érintett vezérlő alkalmazás(oka)t az eltűnés tényéről. A második lépésben a vezérlő alkalmazás eldöntheti, hogy mely más szolgáltatás(oka)t kívánja használni a kieső(k) helyett. Több választási lehetőség esetén a vezérlő alkalmazás kérheti a felhasználót, hogy döntse el ő, mely más szolgáltatásokat szeretné használni, vagy az alkalmazás önmagában is dönthet, ha egyetlen lehetőség van, illetve ha ismerve a felhasználó preferenciáit ki tudja választani a legmegfelelőbb szolgáltatás(oka)t az eltűnt(ek) pótlására. A harmadik lépésben a vezérlő vagy felépíti az új csatornákat vagy lebontja az érintett kapcsolatrendszert.

3.5 Architektúra

A fenti ötleteken alapulva alkottuk meg a MAIPAN middleware-t, melynek architektúrája a 3. ábrán látható. Az alkalmazások a middleware fölött futnak, míg a middleware alatt lévő rétegek végponttól végpontig történő adatkommunikációt biztosítanak. Ahogy az ábrán látható, a protokoll verem függőlegesen két részre van osztva. Az egyik rész az *adat sík*, melynek feladata az alkalmazások közötti hatékony és biztonságos adatátvitel. A *vezérlési sík* a lábak, csatornák és kapcsolatrendszerek menedzseléséért, valamint a kulcs- és jogosultság kezeléséért felel.



3. ábra: A MAIPAN architektúrája

3.5.1 Adat sík

Az alkalmazások a lábaikon keresztül küldik a middleware felé az adatot, amit a *csatorna összekötő réteg* a megfelelő csatornába irányít. A *szállítási réteg* a továbbítandó adatot csomagokra bontja és olyan szolgáltatásokat nyújt, mint például a folyam szabályozás, sorrendezés, automatikus újraküldés, QoS, stb. Csak azok a funkciók kerülnek bekapcsolásra az adott csatornákon, amelyek az adott szolgáltatáshoz szükségesek és hiányoznak a middleware alatti rétegekből. A *kapcsolati réteg* olyan információkat ad a csomagokhoz, melyek a célba éréshez szükségesek: a forrás és cél eszköz címét, valamint a csatorna azonosítóját. Végezetül a *titkosítási réteg* a csomag integritásának ellenőrzéséhez paritás biteket számol és szükség esetén titkosítja a csomagot.

3.5.2 Vezérlési sík

A vezérlési sík tartalmazza azokat a vezérlési funkciókat, melyek a PAN menedzseléséhez szükségesek. A *szolgáltatás vezérlő* rész a helyi szolgáltatások regisztrálását és hozzáférési jogosultságokat kezeli, valamint kommunikál a szolgáltatás felderítő protokollal. Elméletben bármilyen szolgáltatás felderítési protokoll hozzákapcsolható a MAIPAN-hez, úgy mint az SLP, az UPnP vagy a Salutation [21][22]. A kapcsolatrendszereket a vezérlő alkalmazás utasításainak megfelelően a *csatorna vezérlő* hozza létre és konfigurálja újra szükség esetén. A vezérlő alkalmazás igényeinek megfelelően, a csatorna vezérlő kéri meg a PAN-ban résztvevő eszközök *láb vezérlőit*, hogy kívánt lábak között építsék ki a csatornákat. A láb vezérlő utasítja a csatorna összekötő réteget, hogy hozza létre a szükséges csatornát, a szállítási rétegben aktiválja az adott csatornához szükséges szállítási funkciókat és beállítja a csatorna végpontját a kapcsolati rétegben. A *biztonsági vezérlő* kezdeményezi és koordinálja az eszközök közötti azonosítási eljárást, kezeli a szolgáltatás hozzáférési jogosultságokat és tárolja a biztonságos kommunikációhoz szükséges információkat (pl. biztonsági kulcsokat).

3.6 Implementáció

A MAIPAN egy korábbi verzióját Linux operációs rendszerre C-ben implementáltuk [23]. Készítettünk néhány MAIPAN alkalmazást (pl. mp3 lejátszó,

fájlserver) és egy vezérlőt is, hogy vizsgálhassuk a middleware viselkedését [24][25]. Az implementációból tanultakat is beépítettük a middleware most bemutatott verziójába.

4 Összegzés

Cikkünkben bemutattuk a MAIPAN platformot, ami egy olyan middleware, mely személyi hálózatokban működő alkalmazások összekapcsolását teszi lehetővé. E middleware lényege, hogy egy olyan PAN programozási platformot nyújtson, ahol a hardveres és szoftveres erőforrások összeköthetők egymással és a személyi hálózat elosztottsága az alkalmazások számára nem látszik. A javasolt architektúrát használva, az elosztott PAN alkalmazásokat készítő programfejlesztőknek nem kell foglalkozniuk a PAN konfigurálással és dinamikákkal (pl. eszközök eltűnése, illetve megjelenése), és a rendszer által nyújtott egységes alkalmazás programozási interfészt használva egyszerűen készíthetnek PAN szoftvereket.

A MAIPAN ezen a területen újdonságnak számít elsősorban a biztonságos jogosultság kezelési mechanizmusa és a központi vezérlő entitás használata miatt. A MAIPAN jogosultság kezelése egyrészt biztosítja a felhasználó eszközei közötti együttműködést, másrészt biztonságot nyújt a felhasználó eszközeinek más felhasználó eszközeivel szemben és harmadrészt pedig megengedi a különböző felhasználók eszközei közötti ellenőrzött kommunikációt. A MAIPAN a központi vezérlő egység és a vezérlő alkalmazás segítségével kényelmes jogosultság kezelést és PAN konfigurációt biztosít. A megoldás abban is egyedülálló, hogy megengedi a központi entitás cseréjét, azaz a kapcsolatrendszer vezérlési jogai eszközök között szabadon átvihetők

A továbbiakban a jelenlegi rendszer implementációját és teljesítményanalízisét tervezzük elvégezni.

Irodalomjegyzék

- [1] Mark Weiser: „*The Computer for the 21st Century*”, Scientific American, September 1991
- [2] Mark Weiser: „*Some Computer Science Issues in Ubiquitous Computing*”, Communications of the ACM, July 1993
- [3] M. Satyanarayanan: „*Pervasive Computing: Vision and Challenges*”, IEEE Personal Communications, August 2001
- [4] Cecilia Mascolo, Licia Capra and Wolfgang Emmerich: „*Middleware for Mobile Computing (A Survey)*”, In Advanced Lectures in Networking. Editors E. Gregori, G. Anastasi, S. Basagni. Springer. LNCS 2497. 2002
- [5] A. Pope: „*The Corba Reference Guide : Understanding the Common Object Request Broker Architecture*”, Addison-Wesley, Jan. 1998.
- [6] Pubudu Chandrasiri, Ozgur Gurleyen, Yashar Shahabi, Christian Gehrman, Annika Jonsson, Mats Naslund: „*Personal Security Domains*”, Contribution to the 10th WWRP Meeting, New York, October 27-28, 2003
- [7] D. Garlan, D. P. Siewiorek, A. Smailagic, P. Steenkiste: „*Aura: Toward Distraction-Free Pervasive Computing*”, IEEE Pervasive Computing, 2002., <http://www-2.cs.cmu.edu/aura/>
- [8] „*Gaia Project: Active Spaces for Ubiquitous computing*”; <http://gaia.cs.uiuc.edu/index.html>
- [9] „*MIT Project Oxygen*”, Online Documentation, <http://oxygen.lcs.mit.edu/publications/Oxygen.pdf>

- [10] M. Esler, J. Hightower, T. Anderson, G. Borriello: Next Century Challenges: „*Data-Centric Networking for Invisible Computing: The Portolano Project at the University of Washington*”, Mobicom '99., <http://portolano.cs.washington.edu/proposal/>
- [11] Robert Grimm, Janet Davis, Eric Lemar, Adam MacBeth, Steven Swanson, Thomas Anderson, Brian Bershad, Gaetano Borriello, Steven Gribble, and David Wetherall: „*System support for pervasive applications*”, ACM Transactions on Computer Systems, 22(4):421-486, November 2004
- [12] Achilles Kameas, Stephen Bellis, Irene Mavrommati, Kieran Delaney, Martin Colley, Anthony Pounds-Cornish: „*An Architecture that Treats Everyday Objects as Communicating Tangible Components*”, in proceedings of the First IEEE International Conference on Pervasive Computing and Communications (PerCom '03), Fort Worth, Texas, USA, March 23-26, 2003, p. 115
- [13] 2WEAR project: „*A Runtime for Adaptive and Extensible Wireless Wearables*”; <http://2wear.ics.forth.gr>
- [14] „*CORTEX Project: CO-operating Real-time sentient objects: architecture and EXperimental evaluation*”; <http://cortex.di.fc.ul.pt/index.htm>
- [15] Barry Brumitt, Brian Meyers, John Krumm, Amanda Kern, Steven Shafer: „*EasyLiving: Technologies for Intelligent Environments*”, in Proc. of Handheld and Ubiquitous Computing Symposium, (Bristol, England), 2000
- [16] W. K. Edwards, M.W. Newman, J. Sedivy, T. Smith: „*Challenge: Recombinant Computing and the Speakeasy Approach*”, MobiCom'02, September 23-28, 2002, Atlanta, Georgia, USA.
- [17] Ricardo Baratto, Shaya Potter, Gong Su, and Jason Nieh: „*MobiDesk: Mobile Virtual Desktop Computing*”, Proceedings of the Tenth Annual ACM International Conference on Mobile Computing and Networking (MobiCom 2004), Philadelphia, PA, September 26-October 1, 2004
- [18] Jonvik, T.E., Engelstad, P.E., Thanh, D.V.: „*Building a Virtual Device on Personal Area Network*”, Proceedings of 2003 International Conference on Software, Telecommunications and Computer Networks (SoftCOM'2003), Dubrovnik (Croatia) / Ancona, Venice (Italy), October 7-10, 2003
- [19] Jonvik, T.E., Engelstad, P.E., Thanh, D.V.: „*Dynamic PANBased Virtual Device*”, Proceedings of 2nd IASTED International Conference on Communications, Internet and Information Technology (CIIT'2003), 17-19 Nov 2003
- [20] Miklós Aurél Rónai, Kristóf Fodor, Gergely Biczók, Zoltán Turányi, András Valkó: „*MAIPAN: Middleware for Application Interconnection in Personal Area Networks*”, poster at Ubiquitous 2005 conference, San Diego, CA, USA, July 17-21 2005
- [21] Rekesh John: „*UPnP, Jini and Salutation – A look at some popular coordination frameworks for future networked devices*”, California Software Laboratories Inc., Technical Report, June 17 1999
- [22] Feng Zhu, Matt Mutka, and Lionel Ni: „*Classification of Service Discovery in Pervasive Computing Environments*” MSU-CSE-02-24, Michigan State University, EastLansing, 2002
- [23] Kristóf Fodor: „*Implementation of a Protocol Stack for Personal Area Networks*”, diplomamunka, 2003. június
- [24] Fodor Kristóf, Kovács Balázs: „*A Blown-up rendszer megvalósítása*”, HTE-BME diákkonferencia, Budapest, Magyarország, 2003. május
- [25] Balázs Kovács: „*Design and Implementation of Distributed Applications in Ad Hoc Network Environment*”, diplomamunka, 2003. május
- [26] Biczók Gergely, Fodor Kristóf, Kovács Balázs, Szabó Ágoston: *Pervasive computing – rejtett számítástechnika, Híradástechnika*, 2003. március

- [27] Biczók Gergely, Fodor Kristóf, Kovács Balázs, Szabó Ágoston: „*Blown-up rendszer tervezése és megvalósítása*”, első helyezett TDK/OTDK dolgozat, Győr, Magyarország, 2002. november / 2003. április