



Modellvezérelt szoftverfejlesztés a gyakorlatban

Doktori értekezés tézisei

Medve Anna

Okleveles programtervező matematikus

Eötvös Loránd Tudományegyetem Informatika Doktori Iskola
Az informatika alapjai és módszertana program

Doktori iskola vezető:

Dr. Benczúr András, egyetemi tanár

Doktori program vezető:

Dr. Demetrovics János, akadémikus

Témavezető:

Dr. Kozma László, habilitált egyetemi docens

Kutatóhely:

Programozáselmélet és Szoftvertechnológiai Tanszék

A kutatás a TÁMOP4.2.1./B-09/1/KMR-2010-0003, Pannontej Zrt., Sonepar Magyarország Zrt. támogatásával valósult meg.

Budapest, 2013.

Bevezetés

Egy új igény kezelése a szoftverfejlesztésben mindig rejtett kockázatot jelent a változáskezelés jóságára és költségeire nézve. Az un. modellvezérelt technológiákkal [OMG12] már komplex rendszerek esetében is magas absztrakciós szinteken kezelhető a szoftverrendszerek elemeinek funkcionális együttműködése, mind az adatcserék, mind a művelet szintű kölcsönhatások eseteiben.

A modellvezérelt technológiák alkalmazásához szükségesek a formális módszerek bevezetése és az áttérés a kód-központú fejlesztésről a modell-központú fejlesztésre, előtérben a modellezés és modelltranszformációk, az architektúra tervezés, a modell-integrálás és a modelltermékek jósága is [Kup01, Béz04, Koz04, Hoo08, Jéz08, Len08, Cha09, Cla09, Ben10, Ist10, Pat10, Bra12, Jéz12].

A Modellvezérelt Architektúra (Model Driven Architecture (MDA) [OMG12]) az Object Management Group (OMG) terméke elvekkkel és szabályozásokkal a szoftverfolyamat és a szoftvertermékek strukturálására konkrét modellvezérelt módszertanok létrehozásához. Az MDA módszertannal [OMG03] a szoftvertermékek elkülönülnek több szintű modellekkel a kód és platform felett az implementációs technológiáktól független formális rendszermodellbe, amely többszintű modelltranszformációkkal már több platformra megvalósítható.

A modellvezérelt fejlesztés (Model-driven Engineering (MDE) [OMG12]) egy modellvezérelt fejlesztési módszertan a szakterületi nyelvekre [Obe10] és MDA elvekre [OMG03] alapozva. Az *MDE elvek a következők*: modell-alapú reprezentáció, aspektusok kezelése, újrafelhasználás, automatizálás.

Célkitűzések

Kutatásunk tárgya a modellvezérelt szoftverfejlesztés folyamata és a technológiák gyakorlati alkalmazása. Kutatásunk fókuszában a fő MDE elvek állnak: az *aspektusok nézőpontjai* az MDA architektúrával, az *újrafelhasználás* a szoftvertervezési mintákkal és szoftverkomponensekkel, valamint a *szakterületi modellezés technológiái*. Ezeknek az eszközöknek az automatizáltsága jelentősen meghatározza az előállítható termékek tulajdonságait, újrafelhasználásuk lehetőségeit, az alkalmazható modellezési technikákat és ezek sorrendjét a folyamatban. Az automatizáltság és újrafelhasználhatóság végett jellegetesen összetett technológiákkal történik a fejlesztés, amely az „egy probléma egy modellezési nyelv” megközelítést alkalmazza minden MDE aspektusra modellintegrálás elveken. Mindezen technológiai jellemzők által az MDE gyakorlatba ültetése függ a szervezeti képességektől.

A modellvezérelt folyamatban a transzformációs technikák bemeneteiként szolgálnak az üzleti-, architektúra- és tervezési minták, a technológiaválasztás, valamint a minőségi követelmények a megfelelés-vizsgálati pontokkal [OMG03]. Többféle modellvezérelt architektúra minta követhető a szakterületi tudásbázisok alapján. Ilyen folyamatban nagyfokú automatizálás szükséges. A minőségi követelmények technológiafüggése azt eredményezi, hogy az üzleti modell élesen beágyazott a platformmodellbe és a követelményfejlesztés nem zárható le az implementációs technológiákról hozott döntések előtt, hanem már a követelményfejlesztés során hatékonyságvizsgálatot kell végezni, és teszteseteket kell előállítani a funkcionális és a minőségi követelmények inspekcijához. A szakterületi tudásbázisok birtokában nem különülnek el élesen a számításfüggetlen és a platform független architektúra rétegek a folyamatban, és nagy általánosságban a platformmodellek is rendelkezésre állnak [Hut11, Jéz12, Bra12].

Motivációnk maga az MDE elvek gyakorlati alkalmazása.

Keressük a lehetőségeket a szoftvertervezési minták hatékony felhasználásához a modellezés folytonosságának biztosítására a szoftverfolyamatban, amelyben az architektúra modell kulcsszerepét vesszük figyelembe [Per92].

Sok lehetőséget látunk az MDE technológia alkalmazására az üzleti területen. Vizsgáljuk az új technológiák alkalmazhatóságát arra, hogy az üzlet szereplői legyenek a jellemzőket meghatározó és szabályozó résztvevők az üzleti folyamatok modellezésében. Ezzel el lehet kerülni a követelmény feltárás és elemzés leggyakoribb hibáit. Az újrafelhasználás és az aspektusok kezelése fokozható a szabványokra alapozott követelményfejlesztéssel. A fejlesztés hatékonyságát növelni lehet a modell alapú teszteléssel és a követelményekből kinyert tesztesetekkel. Vizsgáljuk a módszerek fejlesztését és a technológiai sorok szervezését.

Az értekezés felépítése

Az értekezés egy bevezető részből, négy számozott fejezetből, és egy összefoglaló részből áll. A bevezetésben ismertetjük a motivációnkat és az értekezés téziseit. Az első fejezetben ismertetjük a modellvezérelt fejlesztés alapvető fogalmait, és kifejtjük a modellvezérelt szoftverfejlesztés dimenzióit, valamint az értekezés fejezeteiben tárgyalt technológiákat. Az értekezésben három tézist tárgyalunk. A 2. fejezetben, az 1. Tézissel megadtunk egy sémát technológiai sorok létrehozására, és egy példányosítását az ITU-T System Design Language technológiákkal, majd kibővítettük a Verimag IFx 2.0 Toolbox ismert verifikációs technológiával. A 3.-as fejezetbe foglaltuk be a 2. Tézist, amely eredménye egy módszertan

keretrendszer létrehozására és újrafelhasználására architektúra-, tervezési- és szakterületi minták összeszövésével az MDE folyamat különböző absztrakciós szintjein. A 4.-k fejezetben a 3. Tézis egy módszertant mutatunk be az üzleti folyamatok változáskezeléséhez MDE folyamatban. Fő eleme az üzleti folyamatok modellezése kooperációban az üzleti szereplőkkel, akik megadják folyamataikat az MDE támogatással. A módszertan végkifejlődése alkalmazással történt több Balaton Régióbeli vállalatnál.

Az egyes fejezetek bevezető részében tekintjük át az adott témakörre nézve a motivációnkat és a munkánkhoz kapcsolódó tudományos eredményeket. Az összefoglaló részben ismertetjük a további kutatási irányokat.

Alkalmazott módszerek

Kutatásunk fő módszerei a vonatkozó publikációk felülvizsgálása és hivatkozása (különös tekintettel a felmérő és osztályozó közleményekre); az automatizált fejlesztőkörnyezeteknek kísérleti megfigyelése és tesztelése; valamint saját készítésű módszereink fejlesztése és felhasználása konkrét ipari projektekből a szakterületi folyamatok modellezésére és gyors alkalmazásfejlesztésre.

Saját eredmények bemutatása

Az értekezésben három tézist dolgoztunk ki, amelyekkel megvalósítottuk célkitűzéseinket és ismertettük a kapcsolódásokat a vonatkozó kutatások eredményeivel.

1. Tézis: Technológiai sor modellvezérelt fejlesztéshez

A modellvezérelt folyamatban nélkülözhetetlen a szakterület-specifikus rendszerfejlesztő nyelvek megléte. Szükség van a modellvezérelt technológiákba integrált verifikáló eszközökre és munkakörnyezetekre, amelyekkel a fejlesztés korai szakaszában fontos szerep hárul a specifikációs nyelvekre és eszközökre. Ez irányú vizsgálatunk célja az, hogy a tervezéstől a futtatható kódig adjunk a felhasználónak javaslatot a modellvezérelt szoftverfejlesztés gyakorlatához.

Az *1. Tézis 1.1 alpontjában* egy sémát állítottunk fel technológiai sorok elemeire modellvezérelt fejlesztéshez. A technológiai sor sémában a változó elemek: a szakterületi modellező nyelvek, az egységesített modellező nyelv, programozási nyelvek, szoftver komponensek, valamint a szabványok. A technológiai sor szervezésében a formális módszerek és a nyelvi technológiák szabványait helyeztük előtérbe [Med05, Med08b, Lei07].

Megadtuk az elvi technológiai sor séma példányosítását az ITU-T rendszerfejlesztő nyelvekkel (International Telecommunication Union-Telecommunication Sector (ITU-T)

<http://www.itu.int/rec/T-REC-Z/en>) [ITU12], amelyek szabványosítása harmonizált az OMG specifikációkkal, elterjedtek az iparban, a gyógyeszközfejlesztésben és a tudomány területein.

1. Tézis. Egy sémát adunk meg *technológiai sorok létrehozására*, modellvezérelt fejlesztéshez (Model Driven Engineering (MDE)) és folyamatba integrált formális verifikáláshoz **(Az értekezés 2. fejezete).**

1.1. Séma felállítása a technológiai sorok létrehozására modellvezérelt fejlesztésekhez, *a nemzetközi távközlési egyesület által létrehozott ITU-T System Design Languages rendszerfejlesztő formális nyelvcsalád felhasználásával.*

1.2. Az 1.1 tézisben javasolt séma bővítése ismert verifikációs technológiai eszközökkel, *a vonatkozó szakirodalomból és gyakorlatból.*

Folyamgráfban ábráztuk az ITU-T, Unified Modelling Languages (UML), Domain-Specific Modelling Languages (DSML) nyelvek helyét és kapcsolatát a fejlesztési folyamatban, amelyhez a koncepcionális tervezést javasoljuk az User Requirements Notation (URN) [ITU12] alapú követelménymodellezéssel.

Az **1. Tézis 1.2 alpontjában** javasoljuk beillesztésre az 1.1 tézispontban megadott technológiai sorokba az IFx 2.0 eszköztárat, valamint a modellalapú teszteléshez a TTCN-3 automatizált tesztelési eszközt és az UML-TP elvi eszköztárat [IFx12, Boz4, ITU12, OMG12, Kup01]. A cél az, hogy az elemzés és a részletes tervezés idején kombinálhassuk a statikus elemzőket és az automatikus modellellenőrzőket a modellek formális verifikálására. Ehhez oktatási környezetben folytattunk kísérletet a verifikáló eszközöknek a szoftverfolyamatba integrálásával [Tót07]. Vizsgáltuk az önálló [Bor10] és a kereskedelmi szoftverekbe beépített automatikus modellellenőrzők alkalmazását elemzésekhez és modelltranszformációkhoz, valamint utóbbiak kombinálását az IFx eszköztárral [Orb08].

Kidolgoztuk az IFx 2.0 Toolbox modellezésbe integrálásának folyamatát az IFx platformszolgáltatások eléréséhez a sokfajta verifikációs eszköz és a C++ nyelvű komponensek beemelésére a verifikálás alatti modellbe. Megfogalmazzuk a verifikátor mérnök munkakört, mint szolgáltatást [Med10b], amely tevékenységeire esettanulmányt adtunk a PIM modellek több rétegben történő transzformációjához egyidejű verifikálással [Med11b], amelyeket általánosítottunk a verifikáló eszköztárak kiválasztásához és a verifikátor mérnök munkafolyamat tevékenységeinek szervezéséhez [Med12b].

Az 1. Tézis közleményei: Med08b, Med10b, Med11b, Med12b. További publikációink a témában: Med05, Med07c, Lei07, Bor10, Tót07, Orb08.

2. Tézis: Szoftvertervezési minták alkalmazása MDE folyamatokban

Ebben a tézispontban azt javasoltuk és adtunk hozzá eljárást és esettanulmányt, hogy a szoftvertervezési minták alkalmazásával hozzunk létre keretrendszerbe foglalt mintagyűjteményt, amely modelltermékként alkalmazható újrafelhasználással az MDE folyamat jobbítására és gyorsítására az absztrakciós szinteken belül és a szintek között. Mindezt az architektúra mintanyelvekkel valósítjuk meg modellvezérelt fejlesztésben, amelyre jellemző, hogy az architektúra döntések alapját az ismert szakterületi platform modellek képezik és az architektúrák szempontjából jelentős követelmények [omg03]. A modellalkotásnak ezt a vonatkozását az architektúra döntésekre nézve tárgyaltuk közleményünkben a [Med11b].

2. Tézis Módszerfejlesztés *keretrendszer létrehozására és újrafelhasználására szoftvertervezési minták összeszövésével* modellvezérelt fejlesztésben. **(Az értekezés 3. fejezete)**

2.1 Módszert és esettanulmányt adtunk *keretrendszer építésére és újrafelhasználására, a POSA2 architektúra mintanyelvek és GOF programtervezési minták összeszövésével*, UML alapú modellvezérelt fejlesztésekhez.

2.2 Módszert és esettanulmányt adtunk *keretrendszer és szakterületi mintanyelv építésére, ill. újrafelhasználására, a POSA2 architektúra mintanyelvek, GOF programtervezési minták és SDL Pattern üzenetkezelési minták összeszövésével* az SDL alapú modellvezérelt fejlesztésekhez.

Mind az általános, mind a szakterületi fejlesztésekhez az elosztott rendszerek architektúra mintanyelveit alkalmaztuk, amelyek a Pattern-Oriented Software Architecture for Concurrent and Networked Objects [Sch00, Bus96], röviden POSA2 néven ismertek. Alkalmaztuk a GOF Design Patterns néven ismert programtervezési mintákat [Gam96, Hil12] a POSA2 architektúra minták viselkedésének modellezéséhez. Szakterületi szoftvertervezési mintákat alkalmaztunk az üzenetkezelés szolgáltatások kontrolljához, amelyek az SDL Pattern Pool minták [Got05].

A **2.1 Tézispont** illetve **2.2 Tézispont** kifejtésével módszert és esettanulmányt adtunk meg a POSA2 architektúra minták modellgyűjteményének a létrehozására a GOF programtervezési minták összeszövésével és újrafelhasználására modellvezérelt fejlesztésben UML fejlesztésekhez (*2.1 tézispont*) [Med10, Med07b], illetve SDL fejlesztésekhez is (*2.2 tézispont*) [Med08]. Az újrafelhasználás esettanulmányai (az értekezés 3.2.2, 3.2.3, 3.3.5 fejezetei)

mutatják a modelltermékek átjárhatóságát a 1. Tézis technológiai közötti modelltranszformációkkal.

A **2.3 tézispont** kidolgozásával megmutattuk, hogy szakterületi mintanyelvet fejleszthetünk gyorsfejlesztésekhez a POSA2 architektúra minták és GOF programtervezési minták modellezésével szakterületi nyelveken. A kapott mintanyelvet a részletes tervezésben kibővítjük a szakterületi minták alkalmazásával újabb absztrakciós szintek bevezetésére minőségi követelmények modellezéséhez és változtathatósághoz, ezzel a tesztelhetőség és átjárhatóság növeléséhez. Az alkalmazott technológiák az SDL, MSC, POSA2, GOF és SDL Pattern Pool, amelyekkel létrehoztuk az SDL Macro-Pattern mintanyelvet [Med08, Med06].

A keretrendszerbe foglalt összesített minták átjárható absztrakciós szinteket képviselnek az MDE folyamatban, általuk az elemzés leegyszerűsödik a fogalmi szintű megfeleltetésekre és a generikus modellek specializációjára az alrendszerek funkciói mentén. A módszer felhasználása bármely szakterületen lehetséges a minták kliens-szerver kommunikációs modelljének a specializációjával a szakterület platformosztályának komponensmodelljét követve. A mintagyűjtemény modelljeivel a kezdő fejlesztők is tudnak mintaalkalmazást végezni a mintákba és a modellekbe beépített szakértői tudás alapján.

A 2. Tézis közleményei: Med08, Med10. További publikációink a témában: Med06, Med07b, Med07c.

3. Tézis: Üzleti folyamatok változáskezelése modellvezérelt fejlesztésben

A modellvezérelt technológiák előnyösek az üzleti területen, ahol jellemző, hogy a szervezeti struktúra állandó fejlődésben van az üzleti stratégia mentén, és a szoftverrendszerek számos érdekelt célcsoport napi szükségleteit elégítik ki. Mind az informatikai rendszerek modernizációja, mind az üzleti változások kezelése során, a szoftverfejlesztési projektek leginkább a meglévő szoftverrendszert alakítják át, és a legritkább esetben vonják vissza teljesen azt, mivel a változtatás alatti rendszer mindig nagyon kiterjedt [Min98]. Ezekhez a rendszerszemponthoz a modellvezérelt követelményfejlesztés [Buh98, Liu01, Wei05, Lam09, Amy11, Mus12] eszköztárát alkalmaztuk a **3. Tézis** kidolgozásában. A **3.1 tézispontban** sémát adtunk technológiai sor szervezéséhez a szolgáltatás fókuszú ITU-T és más MDE technológiákkal az üzleti területre [Med07, Med8b].

Az üzleti stratégiák és alkalmazások újratervezéséhez szükséges a jelenlegi és jövőbeni helyzet specifikálása. A **3.2 tézispontban** egy generikus módszertant adtunk meg az üzleti folyamatok kooperatív feltáráshoz és szabályozásához. Az MDE és URN [ITU12] technológiákra alapozva a folyamatgazdák felbecsülik és validálják saját folyamataikat, majd

a fejlesztők specifikálják azokat, és a területfelelős üzleti elemzők és folyamatgazdák inspekció keretében validálják az általuk megadott követelményeket. A módszertanban megadtunk űrlapokat folyamatfeltáráshoz, inspekció lebonyolításához és hibabejelentéshez. Továbbá esettanulmányokat, oktatási segédleteket, módszerleírásokat folyamatok feltáráshoz, ellenőrzéséhez, specifikálásához és verifikálásához, inspekciós stratégiatervezéshez, modelltermékek dokumentálásához. A módszer kifejlesztését több vállalat konkrét szükségletére adtuk meg [Óvá07, Med10c, Med11d].

3. Tézis. *Módszerfejlesztés az üzleti folyamatok változáskezelésére modellvezérelt fejlesztésben. (Az értekezés 4. fejezete)*

3.1. Egy sémát adtunk *technológiai sor létrehozására az üzleti terület modellvezérelt fejlesztéséhez*, a távközlési területen szabványosított ITU-T System Design Languages nyelvcsalád elemeivel az 1.1 és 1.2 tézispontok technológiáira építve.

3.2. Ipari alkalmazással alátámasztott módszertant adtunk *az üzleti folyamatok specifikálására és inspekció lebonyolítására kooperatív munkaszervezéssel* folyamatgazdák, üzleti elemzők, minőségfelelősök és modellfejlesztők együttműködésében a 3.1 tézispont technológiáira építve.

3.3. Ipari alkalmazással alátámasztott módszertant adtunk a 3.1 tézispont technológiáira alapozva *keretrendszer építésére változáskezeléshez*, az üzleti folyamatok és stratégiák együttes kezelésével az érdekelt felek által.

A vállalati rendszerek sajátosan kritikus rendszerek: az emberi tényezőktől és az irányítási módszerektől függenek. A **3.3 tézispontban** URN alapú módszertant adtunk meg a cél- és forgatókönyv modellek összehangolt fejlesztésével üzleti folyamatok és stratégiák fejlesztésére és generikus keretrendszerbe foglalásukra. A felhasználásuk a célmodellekből kiindulva specializációs iterációkkal történik. [Med12, Med12c]. A keretrendszer részeként generikus modelleket dolgoztunk ki az információbiztonság szabvány [ISO12] és a mobilinformatikai alkalmazások vállalati technológiákba integrálására, specializáltuk elektronikus kereskedelemre konkrét vállalati újratervezésben [Med09, Med11, Med11c]. Mintát dolgoztunk ki a mobilinformatikai alkalmazások vállalati integrálásához és esettanulmányt az URN és UML technológiák alkalmazására a használati esetek leírásából kiindulva [Med12c].

3. Tézis közleményei: Med07, Med08b, Med10c, Med11, Med11c, Med11d, Med12, Med12c. **További publikációink a témában:** Óvá07, Med09.

A tézisek alapjául szolgáló közlemények

- Med07 A. Medve: *Enterprise modelling with the joint use of User Requirements Notation (URN) and UML*. Book chapter In: P. Rittgen (Ed.), *Enterprise Modelling and Computation with UML*, Hershey, USA. Development Ed. IDEA Group, p.46-69, 2007.
- Med08 A. Medve, Il. Ober: From Models to Components: Filling the Gap with SDL Macro-patterns, *Int. Conf. Innovation on Software Engineering CIMCA/ISE'08*, Vienna, Austria, IEEE, p.1252-1257, 2008.
- Med08b A. Medve: *Advanced steps with standardized languages in the re-engineering process*, *Journal of Computer Standards & Interfaces*, Elsevier, 30 (5):315-322, 2008.
- Med10 A. Medve, L. Kozma, Il. Ober: General Modelling Approach Based on the Intensive Use of Architectural and Design Patterns, In: H. Weghom, P. Isaias, R. Vasiu (Eds): *Proc. of the IADIS Int. Conf. Applied Computing 2010*, Timisoara, Romania, p 251-256, 2010.
- Med10b A. Medve, Gy. Orbán, L. Kozma: Let's Go the Verification Engineering, In L. Varga, G. Weiss, G. Kuspér (Eds.) *Proc. of the 8th Int. Conf. on Applied Informatics (ICAI2010)*, pp. 417-428, Eger, Hungary, 2010.
- Med10c Medve, A.: ProjektProcess módszertan fejlesztése - Pannontej Zrt., *Kutatásjelentés*, PE-MIK-VIRT-2/10, 2010.
- Med11 A. Medve: Standards-based Framework for Functionally Integrated Engineering of Information Systems. In T. Szmuc, M. Biró (Eds.) *5th IFIP TC2 Central and Eastern European Conf. on Software Engineering Techniques (CEE-SET 2011)*, Debrecen, Hungary, p.52-58, 2011.
- Med11b A. Medve, L. Kozma: MDA Process and Model Refinements Based on Verimag IFx Verification Tools, In: H. F. Pop, A. Bege (Eds.) *Proc. 8th Joint Conf. on Mathematics and Computer Science, MaCS 2010*, Komarno, Slovakia, July 14-17, 2010, Selected Papers, Novadat Ltd., p.323-336, 2011.
- Med11c Medve, A.: Módszertan elektronikus üzleti technológiák integrálására, Sonepar Magyarország Kft. B2B&B2C&URN projekt, *Kutatásjelentés*, PE-MIK-VIRT-3/11, 2011.
- Med11d Medve, A.: UCM2MSC minőség audit támogatás a ProjektProcessz folyamatspecifikációs módszertanban, Pannontej Zrt., *Kutatásjelentés*, PE-MIK-VIRT-1/11, 2011.
- Med12 A. Medve: Model-based Framework for Integrated Evolution of Business and IT Changes. In: S. Hammoudi, M. Sinderen, J. Cordeiro (Eds.) *ICSOF 2012. Proc. of the 7th Int. Conf. on Software Paradigm Trends*, 24 - 27 July 2012, Rome, Italy, SciTePress, p.255-260, 2012.
- Med12b A. Medve: *Towards MDE Improvements from Integrated Formal Verifications*, J. STUDIA Univ. Babeş-Bolyai, LVII(1):89-104, 2012.
- Med12c Medve, A. Mobil informatikai szolgáltatások üzleti folyamatainak modellvezérelt fejlesztése, AEF project - Informatix Zrt., GOP-1.1.1-09/1-2010-0025 *Kutatásjelentés*, PE-MIK-VIRT-2/12, 2012.

A dolgozat témájában megjelent további publikációink jegyzéke

- Bor10 Zs. Borsi, L. Kozma, A. Medve: Component-based Software Engineering Verification Problems, In: L. Varga, G. Weiss, G. Kuspér (Eds), *Proc. of the 8th Int. Conf. on Applied Informatics (ICAI 2010)* Eger, Hungary, January 27-30, 2010, p.391-400, 2010.
- Lei07 F. Leitold, A. Medve, L. Kovács: SIP security problems in NGM Services, In: K. Al-Begain (Ed.) *Proc. Int. Conf. on Next Generation Mobile Applications, Services and Technologies (NGMAST 2007)*, IEEE, p.241-246, 2007.
- Med05 A. Medve: Standardized Formal Languages for Reliable Model Engineering, *Proc. 25rd. microCAD Int. Conf. on Applied Information Engineering*, ITTC, Miskolc, Hungary, N:315-321, 2005.
- Med06 A. Medve, Il. Ober: Design patterns in SDL: The component configuration pattern case study, *Kutatásjelentés*, PE-MIK-IRT-3/06, 2006.
- Med07b Medve, A., Szima, I.: Esettanulmány a POSA2 Szolgáltatáselérés mintanyelv implementálására, *Kutatásjelentés*, PE-MIK-IRT-5/07, 2007.
- Med07c Medve, A.: UML és SDL társulása a valós idejű és egymásba ágyazott rendszerek modellezésére, Magyar-Francia Tét F/16, *Kutatásjelentés*, PE-MIK-IRT-2/07, 2007.
- Med09 Medve, A., Nagy, T., Nagy, R., Nagy, N.: Kísérlet az ISO/IEC 27001 és 27002 modellezésére és az átfedések kezelésére, *Kutatásjelentés*, PE-MIK-VIRT-1/09, 2009.
- Orb08 Orbán, Gy., Medve, A.: IFx-CADP-SPIN kísérlet, *Kutatásjelentés*, PE-MIK-IRT-3/08, 2008.
- Óva07 Óvádi, P., Medve, A.: SAP PM bevezetése, Le Belier Formaöntöde Ajka Magyarország, *Kutatásjelentés*, PE-MIK-IRT-2/2007.
- Szi08 Medve, A., Szima, I.: Esettanulmány a POSA2 Szolgáltatáselérés mintanyelv implementálására, *Kutatásjelentés*, PE-MIK-IRT-5/07, 2007.
- Tóth07 Tóth, L., Medve, A.: VIATRA kísérlet, *Kutatásjelentés*, PE-MIK-IRT-3/07, 2007.

Irodalomjegyzék

- Amy11 D.Amyot, G.Mussbacher: User Requirements Notation: The First Ten Years, The Next Ten Years (Invited paper), *Journal of Software*, 6(5):747-768, Academy Publisher, 2011.
- BBen10 R.Bendraou, J.-M.Jézéquel, M.-P.Gervais, X.Blanc: A Comparison of Six UML-Based Languages for Software Process Modeling. *IEEE Trans. Software Eng.* 36(5): 662-675, 2010.
- Béz04 J.Bézivin: Model Engineering for Software Modernization, In: 11th Working Conf. on Reverse Engineering (WCRE 2004), Delft, The Netherlands. IEEE Computer Society, 2004.
- Boz04 M.Bozga, S.Graf, L.Mounier, Il.Ober, Iu.Ober, J.Sifakis: The IF toolset, Formal Methods for the Design of Real-Time Systems, LNCS 3185, p. 237-267, 2004.
- Bra12 M.Brambilla, J.Cabot, M.Wimmer: Model-Driven Software Engineering in Practice. Morgan, 2012.
- Buh98 R.J.A.Buhr: Use Case Maps as Architectural Entities for Complex Systems, In: *IEEE Transactions on Software Engineering*, 24(12):1131-1155, 1998.
- Bus96 F.Buschmann, R.Meunier, H.Rohnert, P.Sommerlad: Pattern-Oriented Software Architecture, Volume 1: A System of Patterns. John Wiley & Sons, 1996.
- Cha11 A.W.H.Chang, G.Adamis, L.Erös, G.Kovács, T.Csöndes: A New Approach in Model-Based Testing: Designing Test Models in TTCN-3. In: Il.Ober, Iu.Ober (Eds.) *SDL Forum Toulouse*, p.90-105, 2011.
- Cla09 E.M.Clark, E.A.Emerson, J.Sifakis: Model checking: algorithmic verification and debugging, *Communications of the ACM* 52(11):74-84, 2009.
- Gam95 E.Gamm, R.Helm, R.Johnson, J.Vlissides: Design Patterns: Elements of Reusable Object-Oriented Software. Addison-Wesley, Reading, 1995.
- Got03 R.Gotzhein: Consolidating and Applying the SDL-Pattern Approach: A Detailed Case Study, *Information and Software Technology*, 45(11):727-741, Elsevier, 2003.
- Hoo08 J.Hooman, H.Kugler, I.Ober, Y.Yushtein: Supporting UML-based Development of Embedded Systems by Formal Techniques, *Int. Journal of Software and Systems Modeling*, 7(2):131-155, Springer, 2008.
- Hut11 J.Hutchinson, M.Rouncefield, J.Whittle: Model-driven engineering practices in industry. In R. Taylor et al.(Eds.): *Proc. of the 33rd Int. Conf. on Software Engineering, ICSE 2011, ACM*, p. 633-642, 2011.
- Ist10 Z.Istene: Formal Methods in Robot Control Systems, In: L.Varga, G.Weiss, G.ábor Kusper (Eds): *Proceedings of the 8th ICAI 2010, Eger, Hungary, January 27-30, 2010*, p. 391-400, 2010.
- IFx12 IFx-OMEGA Toolset: <http://www.irit.fr/ifx/>; <http://www-if.imag.fr/>; <http://www-omega.imag.fr/>, 2012.
- ISO12 ISO/IEC 27001, ISO/IEC 27002 Information Security Standard, <http://www.27000.org>, 2012.
- ITU12 ITU-T Recommendation Z. Series: Languages and General Software Aspects for Telecommunication Systems, <http://www.itu.int/rec/T-REC-z>.
- Jéz08 J.-M.Jézéquel: Model driven design and aspect weaving. *International Journal of Software and Systems Modeling* 7(2):209-218, Springer, 2008.
- Jéz12 J.M.Jézéquel, B.Combemale, D.Vojtisek: *Ingénierie Dirigée par les Modèles: des concepts à la pratique*, Ellipses, Paris, 2012.
- Koz04 Kozma,L., Varga,L.: *A szoftvertechnológia elméleti kérdései*, ELTE Eötvös Kiadó, Budapest, 2004.
- Kup01 O.Kupferman, M.Y.Vardi, P.Wolper: Module checking. *J. Information and Computation*, Elsevier, 164(2):322-344, 2001.
- Lam09 A.-van-Lamsweerde: Requirements Engineering – From System Goals to UML Models to Software Specifications. Wiley, 2009.
- Len08 L.Lengyel, T.Levendovszky, H.Charaf: Validated model transformation-driven software development, *International Journal of Computer Applications in Technology*, 31(1):106-119, 2008.
- Liu01 L.Liu, E.Yu: From Requirements to Architectural Design - Using Goals and Scenarios, STRAW'2001, Toronto, p.1- 9, <http://www.cin.ufpe.br/~straw01>, 2001.
- Min98 H.Mintzberg: *Structure et dynamique des organisations*. Edition d'Organisation, Paris, 1998.
- Mus12 G.Mussbacher, J.Araújo, A.Moreira, D.Amyot: AoURN-based Modeling and Analysis of Software Product Lines, In: *Software Quality Journal*, 20(3-4):645-687, 2012.
- Obe10 I.Ober, L.Féraud, C.Percebois: Dealing with variability within a family of domain-specific languages: comparative analysis of different techniques. *Innovations in Systems and SE* 6(1-2):21-28, 2010.
- OMG03 OMG Strategy Study Group MDA Guide Version 1.0.1, 2003, <http://www.omg.org/mda>, 2012.
- OMG12 OMG Specifications – http://www.omg.org/technology/documents/spec_catalog.htm;
- Pat10 A.Pataricza. et al.: Workflow-driven Tool Integration Using Model Transformations, In: G.Engels (Ed.) *Graph Transformations and Model Driven Engineering*, LNCS 5765, Springer, p.224-248, 2010.
- Per92 D.E.Perry, A.L.Wolf: Foundations for the study of Software Architecture. *Software Engineering Notes ACM SIGSOFT*, 17(4):40-42, 1992.
- Sch00 D.C.Schmidt, M.Stal, H.Rohnert, F.Buschmann: *Pattern-Oriented Software Architecture., Patterns for Concurrent and Networked Objects* Wiley, (<http://www.cs.wustl.edu/~schmidt/POSA/>) 2000.
- Wei05 M.Weiss, D.Amyot: Business Process Modeling with URN. *J. E-Business Research*, 1(3):63-90, 2005.