

Architektúra a počítanie orientované na služby a multiagentné systémy (SOA, SOC a MAS)

SOA architektúra je prístup pre voľne viazané, od protokolu nezávislé a na štandardoch založené distribuované počítanie (Computing), kde sú použiteľné zdroje k dispozícii v sieti (internet) vo forme služieb (webových). Trendy v oblasti podnikových informačných systémov sa týkajú systému zostaveného z voľne viazaných súborov webových služieb. Táto oblasť sa javí ako prirodzené prostredie na použitie agentových technológií. Integrácie agentových technológií s inými strategickými technológiami sa týkajú najmä: webových internetových služieb a workflow, čo je tok informácií v podnikovom procese a ich automatizovaného riadenia. Agenty a služby (services) sú dve koncepcie, ktoré sú stále viac integrované a navzájom využívajú svoje výhody. Servisne orientované MAS (multiagentné systémy) budú poskytovať nový prístup pri konštruovaní komplexných aplikácií, pričom vývojár sa bude môcť koncentrovať na vyššie úrovne abstrakcie vývojových fáz a sústrediť na kľúčové konceptuálne štruktúry.

1. Architektúra a počítanie orientované na služby (SOA a SOC)

SOA (Service Oriented Architecture) architektúra je prístup pre voľne viazané, od protokolu nezávislé a na štandardoch založené distribuované počítanie (Computing), kde sú použiteľné zdroje k dispozícii v sieti (internet) vo forme služieb (webových). SOA sa stáva prístupom riešenia podnikovej technológie umožňujúcej zvýšenie agility a flexibility z hľadiska používateľov. Programové komponenty sú služby založené na štandardných protokoloch. SOA používa tzv. zrnitosť služieb pre možnosť efektívneho skladania, zapuzdrenia a ovládania služieb. SOA ponúka tzv. hrubozrnné podnikové služby a tiež tzv. jemne zrnité programové služby. Je alternatívnym modelom tradične užšie viazaných objektovo orientovaných modelov, ako CORBA (Common Object Request Broker Architecture) a DCOM (Distributed Component Object Model). V SOA môžu byť jednotlivé služby konštruované ako objektovo orientované, ale celkový návrh je orientovaný na služby (Services). Architektúra orientovaná na služby – servisne orientovaná architektúra (SOA) oddeluje úlohy softvérového inžinierstva od úloh z oblasti programovania a vytvára softvérových inžinierov bez nutnej potreby byť najprv dobrým programátorom.

1.1 Čo znamenajú a predstavujú SOA a SOC?

Počítanie – výpočty orientované na služby – servisne orientované počítanie SOC (Service Oriented Computing) zodpovedá množine koncepčných prístupov, princípov a metód, ktoré reprezentujú výpočtový proces počítania (Computing) v servisne orientovanej architektúre SOA (Service Oriented Architecture), v ktorej sú programové aplikácie konštruované na základe systému služieb (Services) s nezávislými komponentmi so štandardnými rozhraniami (interfaces). Hlavná idea SOC a SOA je explicitne oddeliť softvérové, programové inžinierstvo (Software Engineering) od programovania s dôrazom na softvérové inžinierstvo a so znížením dôrazu na programovanie. SOC rozdeľuje programový vývoj do 3 častí, ktoré sú reprezentované nasledujúcimi profesiami [15]:

- **Konštruktéri, stavitelia aplikácií (Application Builders)**, ktoré sú vytvárané softvérovými inžiniermi.
- **Poskytovatelia servisu, služieb (Service Providers)**, vytvárajúcich programátormi.

- **Sprostredkovatelia servisu, služieb (Service Brokers)**, ktorí registrujú služby a sprostredkujú aktivity spoločností, organizácií, organizácií počítačového priemyslu a organizácií pre legislatívu.

Poskytovatelia servisu, služby používajú tradičné programovacie jazyky, ako Java, C++ a iné na návrh a programovanie programových komponentov. Všetky komponenty sú „zabalené“ (wrapped) do zodpovedajúceho rozhrania (interfaces) otvorených štandardov a nazývajú sa službami, servismi (services), alebo webovými službami, ak sú služby použiteľné v rámci siete internet. A tak môžu aj **konštruktéri, stavitelia aplikácií** (Application Builders) jednoducho používať služby bez nutnosti ďalšej komunikácie s poskytovateľom služby. Sprostredkovatelia servisu, služby (Service Brokers) umožňujú službám registrovať sa a byť verejne prístupným. Pomáhajú konštruktérom aplikácií nájsť službu, ktorú potrebujú. Konštruktéri aplikácií reprezentujú konečných používateľov, ktorí špecifikujú aplikačnú logiku v „špecifikačnom jazyku vyššej úrovne“ (v žiadnom prípade len s pomocou základných programovacích jazykov) použitím štandardných služieb, čiže použitím komponentov. Aplikační konštruktéri – konštruktéri aplikácií sú vlastne softvéroví inžinieri, ktorí dobre rozumejú programovej architektúre a okruhu problematiky aplikácií. Zrejme sa bude zvyšovať potreba aplikačných konštruktérov a môže prevyšovať i potrebu programátorov.

1.2 Kto používa SOA a SOC?

Väčšina veľkých spoločností pôsobiach v oblasti IS/IT, ako BEA, IBM, ORACLE, MICROSOFT, HP, SAP, CISCO, JUNIPER, SUN Microsystems, SYBASE a i. rad svojich produktov orientujú smerom k SOC a SOA. Jazyky, protokoly a štandardy sú vyvíjané na podporu a ovládanie aplikácií typu SOC. SOC tiež používajú napr. používatelia z oblasti internetových bankových služieb, aerolínií (internetové zaistovanie leteniek), cestovných agentúr a i.

Aké konkrétne produkty ponúka napr. spoločnosť Oracle v oblasti SOA?

Oracle SOA Suite [13] je kompletný súbor servisných infraštruktúrnych komponentov na výstavbu, rozmiestnenie a ovládanie nejakej SOA architektúry. Obsahuje Oracle JDeveloper 10g na vývoj Java aplikácie v rámci SOA architektúry, Oracle Business Rules na modifikáciu procesnej a aplikačnej logiky bez nutnej znalosti programovania, Orac-

le BPEL Process Manager umožňujúci podnikom a spoločnostiam usporiadať rôznorodé aplikácie a webové služby do podnikových procesov, Oracle Business Activity Monitoring, ktorý je komplexným riešením na výstavbu kompletných obrazoviek a alarmov v reálnom čase a zobrazuje a monitoruje podnikové procesy a služby, Oracle Web Services Manager, Oracle Enterprise Service Bus, ktorý je kľúčovým komponentom SOA architektúry na integráciu na báze štandardu, dát a podnikových aplikácií s organizačnou štruktúrou podniku a tradičnými partnermi.

2. Architektúra podniku v súvislosti s orientáciou na služby

Podnik orientovaný na služby – servisne orientovaný podnik (SOE – Services Oriented Enterprise) spája v omnoho väčšej miere podnikové procesy v horizontálnej úrovni. V rámci tejto podnikovej SOA architektúry poskytujú táto infraštruktúra podnikovým architektom a architektom dozeraujúcim na bezpečnú schopnosť behu (run) týchto služieb konzistentnosť v rámci (pozdĺž podniku). SOE a SOA nemožno uzavrieť do jedného balíka, nejde o jednu technológiu a nedajú sa nimi riešiť všetky problémy. Na obr. 1 je zachytená architektúra podniku, spoločnosti v súvislosti s orientáciou na služby. Následne je uvedený význam vybraných pojmov z obr. 1, ktoré ešte neboli uvedené alebo nie sú všeobecne známe.

„Choreografie“, skladanie služieb (Choreography of Services). WSCI (Web Services Choreography Interface) je jazyk založený na XML, opisujúci tok správ, ktoré si vymieňajú v procese zúčastnené webové služby v rámci interakcií s inými službami. Jazyk WSCI opisuje správanie webových služieb. Tie sú vyjadrené v termínoch okamžitých a logických závislostí medzi vymieňajúcimi sa správami, znakmi sekvencných pravidiel (rule), korelácií, ovládání chybových stavov (Exceptional Handling), transakcií a i.

Prijatie paradigma služieb (SPA – Services Paradigm Adoption) – paradigma služieb v rámci podniku znamená prijatie do podnikových štruktúr termínov (a realizácií toho, čo sa pod termínom skrýva) používaných v rámci služieb. To znamená, že podnikové informačné systémy, technológie, infraštruktúra majú byť dekomponované do termínov a vyjadrené v termínoch fungujúcich prevádzkových služieb.

„Zrornosť“ služieb (GoS – Granularity of Services) – zodpovedá rozsahu platnosti, dosahu funkcionality uvažovanej služby. Dobre, jemne (fine) „zrnatá“ služba (fine-grained services) poskytuje relatívne malú časť, malé množstvo plne použiteľného podnikového procesu napr. prístup k dátam. Hrubo-zrnné služby (coarse-grained services) sú konštruované z jemne zrnatých služieb.

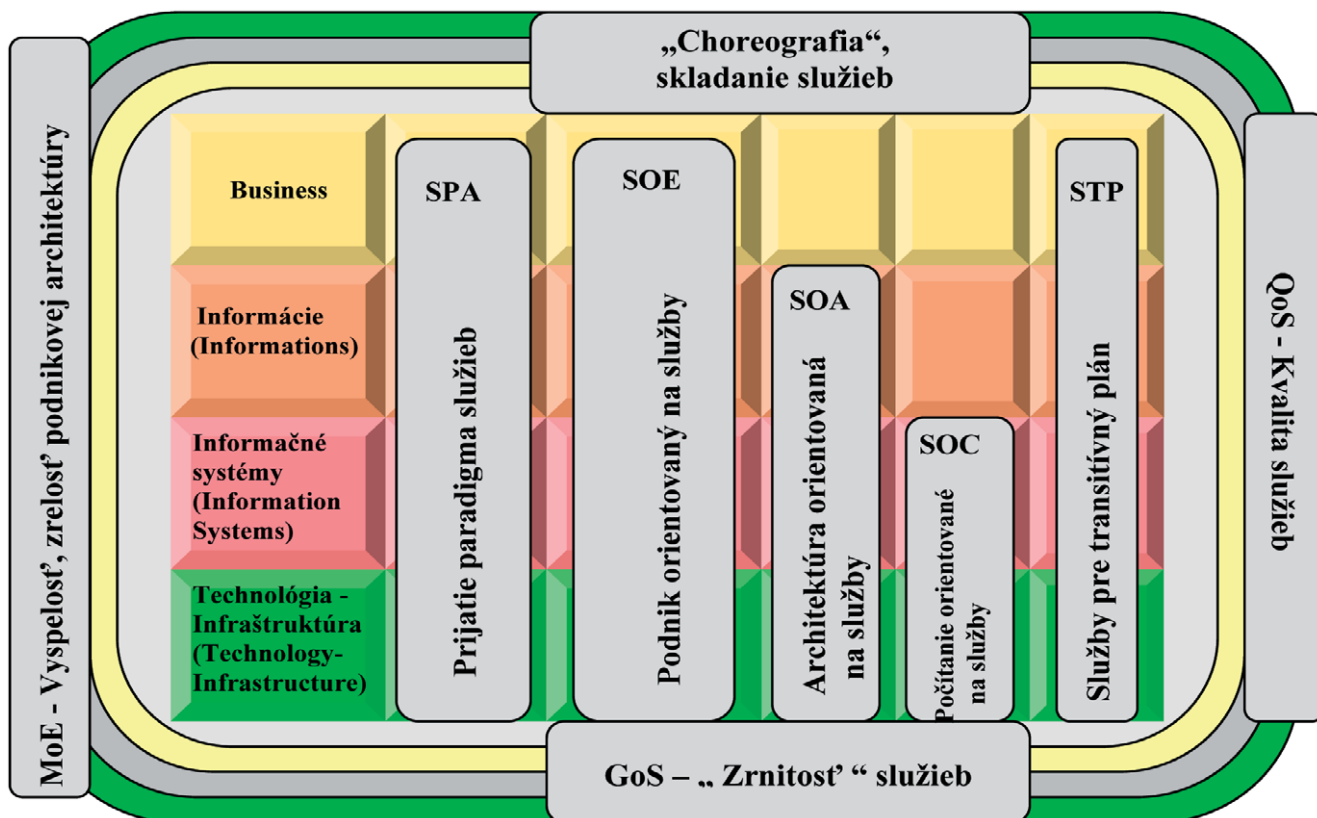
Vypelost', znalosť podnikovej architektúry (MoE – Maturity of the Enterprise). Model vypelosti SOA podniku poskytuje základ na diskusiu medzi odborníkmi IT – informačných technológií podniku a podnikovými používateľmi o aplikovateľnosti a prínose SOA pozdĺž (across) piatich úrovni vypelosti (maturity). Ide o úrovne:

1. základné služby,
2. architektúra služieb,
3. obchodné služby, spolupracujúce služby,
4. meranie podnikových služieb,
5. optimalizácie podnikových služieb.

Informačný systém podniku obsahuje databázové systémy, operačné systémy a siete. SOC je použiteľná pre všetky tieto 3 úrovne. Na služby orientovaný podnik je štandardizovaný odporúčaniami organizáciou OASIS (Organization for the Advancement of Structured Information Standards; <http://www.oasis-open.org>). Typické aplikácie sú implementované ako podnikové webové služby.

3. Agentovo založená, servisne orientovaná architektúra

Obchodné a priemyselné spoločnosti sa stále viac zaujímajú o aplikácie vyhovujúce požiadavkám spolupráce, flexibility, adaptability priemyselných, podnikových a obchodných procesov. Trendy v oblasti podnikových informačných systémov sa týkajú systémov zostavených z voľne viazaných súborov webových služieb. Táto oblasť sa javí ako prirodzené prostredie na použitie agentových technológií [10]. Agent je aktívna, trvalá programová entita, ktorá môže vnímať, správať sa „rozumne“ voči svojmu okoliu, pôsobiť naň a môže komunikovať s inými agentmi. Integrácie agentových technológií s inými strategickými technológiami sa týkajú najmä webových internetových služieb a workflow, čiže toku informácií v podnikovom procese a ich automatizovaného riadenia; workflow znamená automatizáciu celého podnikového procesu, alebo



Obr.1 Architektúra podniku, spoločnosti v súvislosti s orientáciou na služby podľa [9]

jeho časti, počas ktorého sa dokumenty, informácie alebo úlohy (Tasks) odovzdávajú od jedného účastníka procesu druhému podľa procedurálnych pravidiel tak, aby sa dosiahlo, alebo prispelo k plneniu celkových, globálnych systémových pravidiel (Rule engine) a sémantického webu. Agentová technológia sa viac uvažuje ako jedna z najzaujímavejších technológií pre úspešnú podporu SOA a SOC.

3.1 Agenty a služby

Rad úloh riešených obvykle klasickým algoritmickým postupom je riešiteľný jednoduchšie pomocou interakcií. Paradigmatický posun je zdôrazňovaný trendmi smerom k použitiu SOC a MAS. Tieto trendy sú vzájomne komplementárne, SOC potrebuje vlastnosti proaktívneho správania a schopnosť vyjednávania, čo poskytujú agenty, zatiaľ čo agenty potrebujú schopnosť pre vzájomné porozumenie v zmysle sémantického webu vo forme ontológií objektov a procesov [10].

Ontológie sú používané v oblasti umelej inteligencie, sémantického webu (Semantic Web; <http://www.w3c.org>), softvérového inžinierstva a architektúry informačných systémov ako forma reprezentácie znalosti danej reality alebo časti tejto reality.

Agentovo založené webové služby. Typické agentové architektúry majú mnoho spoločných rysov s webovými službami. Agentová architektúra poskytuje adresár žltých (yellow) a bielych (white) stránok (podobne ako UDDI pre webové služby), kde agenty ponúkajú svoje jednoznačné, typické funkcie a iné agenty vyhľadávajú agenty s potrebnými funkciami. Avšak agenty rozširujú webové služby v niekoľkých oblastiach.

UDDI (Universal Description, Discovery and Integration) [17] je špecifikácia pre distribuované, webovo založené registre webových služieb. UDDI je tiež verejne prístupná s možnosťou implementovať a registrovať informácie o webových službách a následne ich ponúknuť iným entitám (businesses). UDDI je založená na takých štandardoch, ako jazyk XML a protokol SOAP (Simple Object Access Protocol). Poskytuje štandardizovaný mechanizmus na lokalizáciu webových služieb.

Keď porovnáme agentové technológie s nejakým ideálnym mechanizmom na implementáciu komplexných systémov, agentová technológia sa ukazuje ako veľmi vhodná pre aplikácie vyžadujúce komunikácie, aplikácie založené na distribuovanom počítaní a informačných systémoch a vyžadujúce autonómne komponenty adaptabilné na zmeny. „Manželstvo“ medzi agentovou technológiou a ontológiou je druhom technológiou, ktorá mení tvár podnikového programového vybavenia. Ontológia by mohli dovŕšiť riešenie problému: ako efektívne objaviť webové služby a ako urobiť možnou vzájomnú spoluprácu heterogénnych webových služieb. V tomto scenári agenty reprezentujú chrbticu (backbone) [11].

Inžinierstvo SOC aplikácií. Spojenie myšlienok SOC a MAS (Multiagentných systémov) môže priniesť schopnosť výstavby systémov na báze konvenčných informačných technológií a štandardizovať spôsob, ktorý môže pomôcť štandardizácii praktického vývoja, monitorovanie a ovládanie rozsiahlych systémov a použiť na to štandardizované vývojové nástroje [15].

Architektúru nejakej aplikácie špecifikujeme pomocou workflow, kauzálneho a procesného modelu, cieľového grafu, podgrafu a i. Toto musíme realizovať pomocou zostavenia, zloženia z použiteľných služieb. Tieto služby môžeme nájsť v lokálnych skladoch (repository) alebo môžu byť uložené v rámci siete internet. Inžinierstvo nejakého SOC systému je potom proces objavovania a spájania (matching) vhodných služieb, plánovania ich interakcií a skladania pre ich beh v čase (runtime). Tento proces je ilustrovaný na obr. 2. Systémová architektúra by mohla byť organizovaná v rámci komunit rôznych druhov agentov: poskytovatelia služieb, asistenti personálnych služieb a agenty typu sprostredkovatelia služieb, manažér používateľských profilov, manažér workflow a i. Na dosiahnutie cieľov by tieto autonómne agenty mohli byť schopné realizovať úlohy kooperácií a súťaží s inými agentmi a spolupracovať s inými entitami a systémami (softvérovými systémami spro-

stredkujúcimi odporúčaniami odborných organizácií, legislatívne predpisy a i.). Mohli by podporovať zmeny v dynamickom správaní modifikovateľnom na základe procesných pravidiel. Konečne by mohli byť schopné výstavby workflow, skladať webové služby a monitorovať ich realizáciu.

V [14] je prezentované agentovo založené prostredie nazývané MASE (Multiagent System Environment), ktoré dovoľuje dynamické zostavovanie webových služieb. Jeho architektúra je založená na spoločnosti agentov založených a zložených z dvoch druhov agentov: komponentový manažér a workflow manažér. Každý komponentový manažér je asociovaný, združený s jednou, alebo niekoľkými webovými službami a je zodpovedný za interakcie s nimi prostredníctvom nástroja WSIG JADE add-on [14]. JADE je prostredie na vývoj multiagentných systémov podľa špecifikácie organizácie FIPA – <http://www.fipa.org>. Návod na prácu s nástrojom WSIG JADE add-on – JADE WSIG (Web Services Integration Gateway) Guide je k dispozícii na adrese http://jade.tilab.com/doc/tutorials/JADE_WSIG_Guide.pdf. Komponentový manažeri sú schopní iniciovať webové služby, previesť správu z jazyka ACL (Agent Communication Language, pozri <http://www.fipa.org>) do opisu v jazyku WSDL (Web Services Description Language) ai.

Workflow manažeri majú za cieľ podporiť používateľov v procese výstavby workflow skladaním kompozícií externých webových služieb a monitorovať ich realizáciu. Na zaistenie týchto komplexných aktivít poskytujú Workflow manažeri používateľom dva alternatívne automatické procesy, a to preddefinovaný workflow, ktorý je extrahovaný zo skladu (repository) štandardov a spoločných šablón, dynamické workflow, keď Workflow manažér podľa požiadaviek používateľa vytvorí nové workflow zložené zo základných, atomických (jemne „zrntých“) služieb použiteľných v systéme. Systém MASE tiež ponúka používateľom možnosť „ručnej“ výstavby workflow.

3.2 Workflow, WS-BPEL, služby a agenty

Zdokonalenie SOA, vplyv siete internet a sémantický opis služieb nám dovoľujú predpokladať, že webové služby sa perspektívne stanú a niekde už stávajú základnými podnikovými komponentmi. Tradičné interakcie medzi ľuďmi a programami budú v nových generáciách podnikového programového vybavenia nahradené interakciami hlavne medzi úlohami (tasks), ktoré sú dynamické a flexibilné [3].

V počítačovej vede sa aplikácie obvykle definujú:

aplikácie = algoritmy + dáta.

Dnes môže byť táto definícia nahradená:

aplikácie = počítanie + koordinácie.

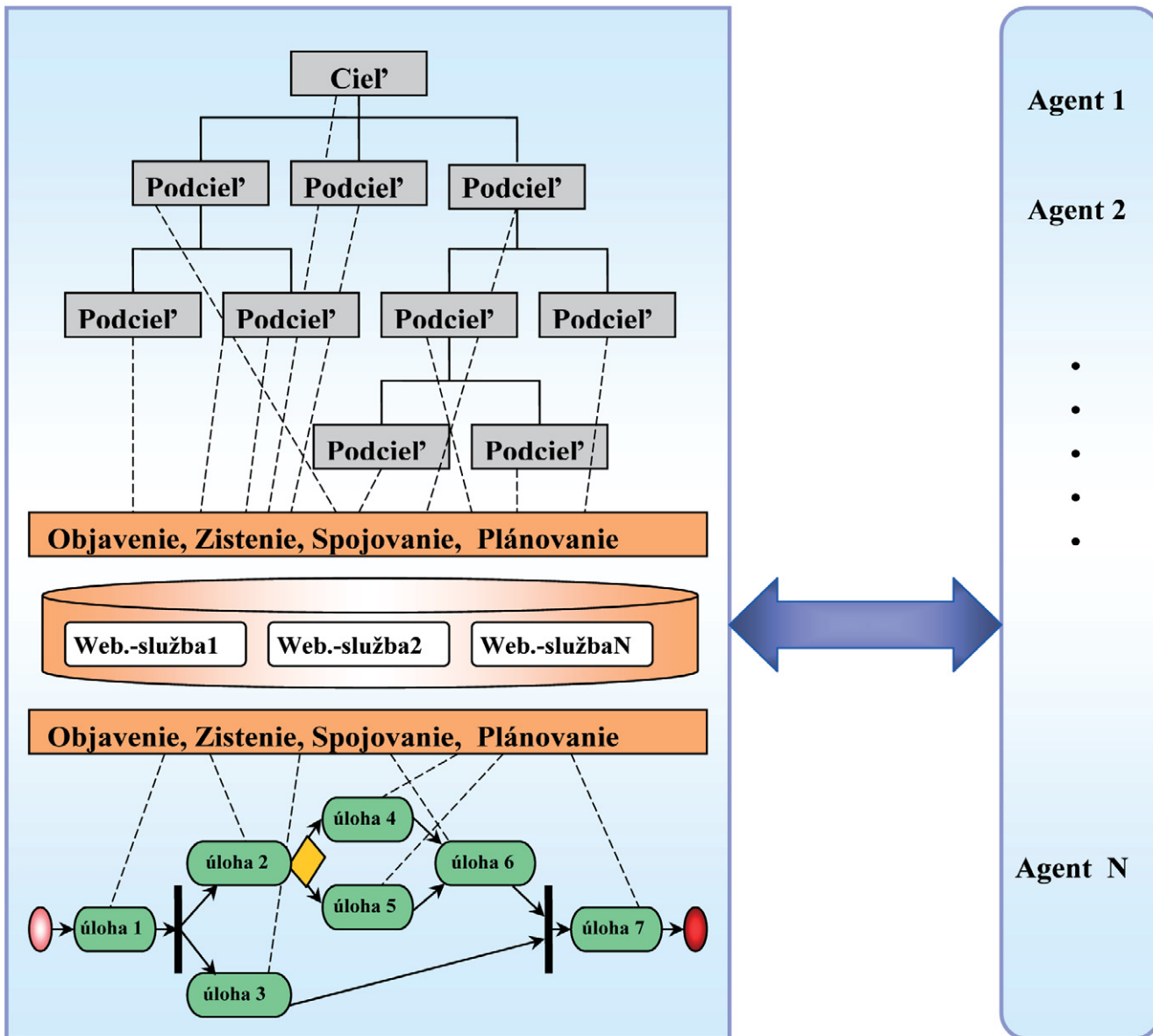
Podľa [3] jazyky opisujúce workflow a s nimi združené nástroje na návrh mohli by byť použité na špecifikácie multiagentných systémov. Jazyk BPEL (Business Process Execution Language) pre webové služby by mohol byť použitý na vyjadrenie začiatočného návrhu usporiadania MAS, ktorý sa potom môže inteligentne adaptovať na meniace sa podmienky prostredia.

BPEL4WS (pôvodne vzniklo v roku 2002 ako zlúčenie odporúčaní spol. IBM, Microsoft a BEA) je v podstate štandardom na báze jazyka XML na vyjadrenie workflow skladajúceho webové služby. BPEL4WS je dnes známe pod označením WS BPEL. Vzhľadom na to, že BPEL4WS (Business Process Execution Language for Web Services) opisuje vzťahy medzi webovými službami vo workflow, agenty reprezentujúce webové služby by mohli poznať ich vzťahy apriori.

Systémy na ovládanie workflow nie sú často navrhnuté na rešpektovanie dynamických zmien v prostredí, ktoré vyžaduje adaptívnu reakciu. Adaptívny workflow [1] sa často požaduje v rámci internetových obchodných činností typu B2B i B2C:

adaptívny workflow = webové služby + agenty.

Súčasná špecifikácia WS-BPEL definuje jazyk založený na odporúčaní XML na formálny opis podnikových procesov založených na „orchestrácii“ (Orchestration), resp. „choreografii“ (Choreography) webových služieb. Ide o otvorený štandard podľa organizácie OASIS.



Obr.2 SOA, inžinierstvo SOC, proces objavovania, skladania a plánovania vhodných služieb a začlenenie MAS (multiagentného systému), obr. upravený podľa [10]

WS-BPEL workflow je štruktúrovaný XML dokument zložený z troch hlavných častí: (i) z definície procesných atribútov, (ii) z definície súhry realizácie, (iii) z opisu aktivít, ktoré majú byť realizované.

V priemyselných podnikoch vzrastá záujem o ovládanie podnikových procesov a jazyk WS-BPEL je v širokom rozsahu prijímaný ako jazyk na použitie v definícii workflow. Na obr. 2 je vyjadrený vo forme cieľového grafu (Goal graph – hore), workflow (dole), so začlenením MAS, ako architektúra a počítanie orientované na služby – SOA a SOC vytvárajú procesy objavovania a skladania vhodných služieb uspokojujúcich požadované špecifikácie, ktoré sú práve vyjadrené vo forme uvedeného cieľového grafu a workflow.

4. Záver – servisne orientované multiagentné systémy a ich rozvoj v spolupráci so SOA a SOC

Servisne orientované MAS budú poskytovať nový prístup pri konštruovaní komplexných aplikácií, pričom vývojár sa bude môcť koncentrovať na vyššie úrovne abstrakcie vývojových fáz a sústrediť sa na kľúčové konceptuálne štruktúry [12]. Vývoj rozsiahlych MAS volá po nových inžinierskych metódach. Takéto systémy musia byť samoorganizujúce a musia podporovať rekonfigurovateľnosť a nový návrh systému v reálnom čase.

Predikcie správania rozsiahlych MAS sú náročné. Niekedy môžu pomôcť nástroje a techniky zo štatistickej fyziky, teórie sietí a i.

Služby (Services) by mohli podporovať MAS zahrnujúci:

- (i) ovládanie heterogénnych informačných podnikových systémov,
- (ii) vedecké výpočty s rozsiahlymi dynamicky rekonfigurovateľnými zdrojmi,
- (iii) výpočty a informácie poskytnuté používateľovi do mobilných zariadení („mobilné počítanie“) v reálnom čase.

Za 10 rokov sa bude rozsah aplikácií MAS a SOC zvyšovať, napr. v oblasti inteligentných domov, štandardných podnikových ERP systémov, geografických systémov a i.

Za 15 rokov môžeme očakávať kompletne včlenenie MAS do výrobných systémov s možnosťami „uvedomiť si“ sám seba a byť schopný detegovať a opravovať svoje chybné správanie a rozhodnutie.

Táto práca bola vypracovaná s podporou programu č. MSM 6046137306 MŠMT ČR.

Literatúra

- [1] BUHLER, P. A., VIDAL, J. M., VERHAGEN, H.: Adaptive Workflow = Web Services + Agents. College of Charleston, University of South Carolina, Stockholm University. <http://jmvidal.cse.sc.edu/papers/buhler03c.pdf> (2008).
- [2] BUHLER, P. A., VIDAL, J. M.: Enacting BPEL4WS specified Workflows with Multiagent Systems. <http://www.agentus.com/WSABE2004/program/02-Buhler.pdf> (2008).

- [3] BUHLER, P. A., VIDAL J. M.: Towards Adaptive Workflow Enactment Using Multiagent Systems. <http://portal.acm.org>, (2008).
- [4] BURIAN, P.: Metodologie a multiagentní system Jadex – návrh informačního systému pomocí multiagentního systému. Systems Integration 2007 – 15th International Conference, Prague, CZ, June 10 – 12, 2007.
- [5] CABRI, G., LEONARDI, L., PUVIANI M.: Service-Oriented Agent Methodologies. Università di Modena, <http://www.agentgroup.unimo.it/MOON/papers/pdf/wetice07.pdf> (2008).
- [6] CICORTAS, A., IORDAN, V.: A Multi-Agent Framework for Execution of Complex Applications. Acta Polytechnica Hungarica, Vol. 3, No. 3, 2006.
- [7] CHEN, Y.: Service-Oriented Architecture: Concept, Programming, and Applications. ASU Workshop on Service-Oriented Architecture Education, Research, and Applications Arizona State University, Tempe, Arizona, May 15 – 16, 2006.
- [8] Enterprise service bus: http://en.wikipedia.org/wiki/Enterprise_service_bus#ESB_Architecture
- [9] Institute For Enterprise Architec. Develop.: <http://www.enterprise-architecture.info> (2008)
- [10] HUHNS, M. N.: Software Development with Objects, Agents, and Services. University of South Carolina. <http://www.open.org.au/Conferences/oopsla2004/PapersAO/Keynote-Huhns.pdf>. (2008).
- [11] HUHNS, M., SINGH M. P. (editors): Service-Oriented Computing: Key Concepts and Principles. JAN., FEB. 2005, IEEE Internet Comp., Publish. by the IEEE Comp. Soc. 1089-7801/05/. <http://www.cse.sc.edu/~huhns/journalpapers/V9N1soc.pdf>, (2007).
- [12] HUHNS, M. N., SINGH, M. P. (editors): Research Directions for Service-Oriented Multiagent Systems. IEEE Internet Comp. 1089-7801/05/, IEEE Computer Society NOV., DEC. 2005.
- <http://csdl.computer.org/dl/mags/ic/2005/06/w6065.pdf>, (2008).
- [13] Oracle SOA Suite Component: <http://www.oracle.com/technologies/soa/soa-suite.html>, (2008).
- [14] POGGI, A., TOMAIUOLO, M., TURCI, P.: An Agent-Based Service Oriented Architecture. DII, University of Parma, Parco Area delle Scienze 181A, 43100, Parma, <http://woa07.disi.unige.it/papers/PoggiSOA.pdf>, (2008).
- [15] TSAI, W. T.: Service-Oriented System Engineering: A New Paradigm. Proceedings of the 2005 IEEE International Workshop on Service-Oriented System Engineering (SOSE '05), 0-7695-2438-9/05 2005 IEEE. <http://csdl.computer.org/dl/proceedings/sose/2005/2438/00/24380003.pdf>, (2008).
- [16] TSAI, W. T., CHEN, Y., BITTER, G., MIRON, D.: Introduction to Service-Oriented Computing. Computer Science and Engineering Department, Arizona State University, <http://www.public.asu.edu/~ychen10/activities/SOAWorkshop/Background.pdf>, (2008).
- [17] UDDI: <http://www.uddi.org> (2008).

Ing. Pavel Burian, CSc.

**VŠCHT v Prahe
Fakulta chemicko-inženýrská
Ústav počítačové a radiacej techniky
e-mail: burianp@vscht.cz**

36