

Szolgáltatásalapú alkalmazások

Krauth Péter

A vállalatok informatikai infrastruktúrája a következő évtizedben alapvető átalakításokon fog keresztülmenni, amelyek során az eddigieknél jóval rugalmasabb és könnyebben változtatható ún. szolgáltatás-orientált és eseményvezérelt architektúrákra térnek át, és ennek eredményeképp a nagy, monolitikus üzleti alkalmazásaikat moduláris, osztott, jól leírható és könnyen lecserélhető komponensekből (azaz szolgáltatásokból) felépülő szolgáltatásalapú alkalmazások váltják fel.

1. Témakör

Az üzleti élet dinamikája alapvetően megváltozott az internet elterjedésével és a globalizációs erők megjelenésével. A szervezetek a korábbinál jóval gyakrabban kényszerülnek megváltoztatni a folyamataikat, hogy alkalmazkodhassanak a változó körülményekhez. Az ilyen gyors változtatás azonban a gyakorlatban kivihetetlen, amíg e folyamatok hagyományos, nehezen változtatható IT-alkalmazásokkal vannak elválaszthatatlanul összekötve, és miután a szervezetek hatékony működése ma már elképzelhetetlen informatika nélkül.

Sőt, az informatika egyre közvetlenebb kapcsolatba kerül az üzleti tevékenységekkel, és ez magában hordja annak szükségességét, hogy az innovatív, újszerű üzleti megoldások gyorsan, hatékonyan és eredményesen valósulhassanak meg. Ez állt eddig is – és fog állni ezután is – az informatika egyedi (innovatív) alkalmazásai mögött.

Az egyedi üzleti igényeket kielégítő alkalmazások létrehozásának módja azonban manapság *radikálisan átalakul*. Az egyedi szoftverfejlesztés visszaszorul már meglévő komponensek, *rendszerek összeépítésének*, integrációjának rovására, és az integráció egyre inkább kiterjed külső – tehát nem egy adott szervezetben kialakult, hanem szélesebb körű használatra kifejlesztett – komponensek bevonásának képességére is. A kódkészítő programozó helyére az üzletet értő IT-közvetítő (az ún. *integrációs bróker*) lép.

Azonban nemcsak az alkalmazáskészítés *jellege* változik, hanem azok az *elemek* is, amiből az új alkalmazások felépülnek. A nagyon szorosan kapcsolódó – szinte már a saját használhatóságukat is korlátozó – *objektumok* helyett egyre inkább az egymáshoz jóval *lazábban kapcsolódó*, azonban *kifelé is nyitott* komponensek (például ún. webszolgáltatások) terjednek el.¹ Ezek felhasználásának képessége egyre döntőbbé válik a vállalatok informatikai vonatkozású versenyképességében, és ezáltal átalakítja az informatikai infrastruktúrákat: olyan szoftverarchitúrára van szükség, amely biztosítja a dinamikus és hatékony építkezést a webszolgáltatásokból. Az üzleti folyamatokat és megoldásokat ilyen módon speciális ún. *szolgáltatás-orientált architektúrákra* (Service-Oriented Architecture, SOA) alapozva célszerű megtervezni és kifejleszteni.

A SOA az üzleti alkalmazásokat érintő legnagyobb architektúrális változás a kliens-szerver architektúráknak az 1990-es évek elején történt megjelenése óta. Megváltoztatja azt, ahogy az üzleti alkalmazásokat megtervezik és kifejlesztik, és meghatározza azt, ahogy az üzleti alkalmazások jelenlegi modernizálási hulláma megvalósul.

¹ Fontos szerepe van ebben a web 2.0-ás jelenségnek is. Vö. *A web 2.0 (és ami mögötte van)* elemzés.

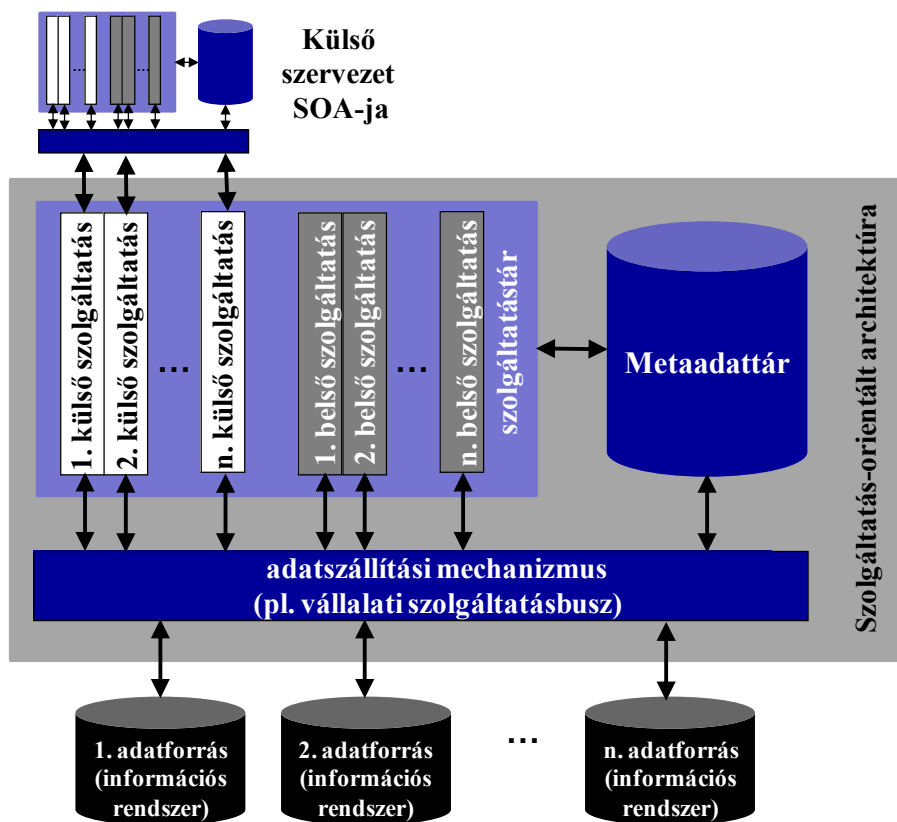
A SOA-t az OASIS² (<http://www.oasis-open.org>) – kicsit absztrakt módon – a következőképpen határozza meg:

„Olyan elosztott erőforrások szervezését és hasznosítását biztosító megközelítés, amelyek különböző tulajdonban lehetnek. Egységes eszközöket nyújt ezeknek az erőforrásoknak a közzétételére, megtalálására, használatára és az egymással való együttműködésre, hogy a kívánt hatásokat mérhető célkitűzésekkel és elvárásokkal összhangban lehessen biztosítani.”

Egyszerűbben fogalmazva: a SOA olyan architektúra, amelynek szolgáltatáskomponenseit újra fel lehet használni, gyorsan ki lehet cserélni és helyettesíteni lehet őket anélkül, hogy ez hatással lenne az üzleti tevékenységre. E megközelítés segítségével olyan szolgáltatásalapú, üzleti alkalmazások készíthetők, amelyek *moduláris* (modular), *osztott* (distributable), *jól leírható* (clearly defined), *könnyen kicserélhető* (swappable) és *újrafelhasználható* (sharable/reusable) „szolgáltatásokból” épülnek fel.³ E szolgáltatások üzleti funkcionalitást nyújtanak, és könnyen összekapcsolhatók, hogy üzleti folyamatokat támogathassanak velük. A SOA-t alkalmazó adatközpontok gyorsan tudnak reagálni a változó üzleti igényekre anélkül, hogy hosszadalmas újrafelvezésen vagy újratervezésen kellene keresztül menni. A SOA költséget takaríthat meg az azt alkalmazó szervezetek számára és jóval agilisabbá teszi őket azzal, hogy kompakt, újrafelhasználható szolgáltatások fejlesztésére ösztönöz, ezért egyre több szervezet állítja át üzleti folyamatait olyan közös „csomagolt szolgáltatásokra”, amelyeket különböző üzleti területeken lehet megosztva használni.

² Az OASIS (Organization for the Advancement of Structured Information Standards) egy nem nyereség-orientált, nemzetközi konzorcium, amely élen jár az elektronikus üzletviteli szabványok kialakításában, harmonizálásában és elfogadtatásában.

³ Külön-külön e tulajdonságok egyike sem újdonság a komponensalapú alkalmazásfejlesztésben, azonban az, hogy a komponensek, mint építőelemek, mind az öt tulajdonsággal egyszerre és együttesen rendelkezzenek, megkülönböztető jegye a szolgáltatásalapú alkalmazásfejlesztésnek.



1. ábra: A szolgáltatás-orientált architektúra fő alkotóelemei

Az 1. ábra mutatja a SOA legfontosabb elemeit. A szolgáltatásokat olyan felületekkel adják meg, amelyek elfedik, hogy például a megvalósításuk csak egy egyszerű, egyedi program, vagy valamilyen komplex integrált folyamat húzódik a háttérben. Ezek a szolgáltatásfelületek, amely ilyen módon függetlenek a megvalósításuktól, és amelyeket dokumentálni kell és nyilvántartásba kell venni, alapvető fontosságúak SOA stratégiai tervében.

Egy jól tervezett szolgáltatást egy újabb modul nem tud kizárólagos módon használni. Egy rosszul tervezett szolgáltatás viszont az alkalmazott nem dokumentált feltételezésekkel és szabályokkal lényegében „hózzáláncolja” magát egy adott, őt használó modulhoz, ami éppen a SOA fő előnyét szünteti meg.

Ha egy különálló programfelület egy teljes üzleti műveletet ír le, és ez a felület olyan módon van nyilvántartásba véve és dokumentálva, hogy egy külső (más szervezeti egység vagy akár más szervezet által használt) alkalmazás meg tudja találni, a leírását fel tudja dolgozni és meg tudja hívni anélkül, hogy belülről ismerné a felület mögötti megvalósítást, akkor ez SOA-szerű szolgáltatás.

A SOA lazán kapcsolt jellegét működés közben egy *adatszállító komponens* (transport component, például vállalati szolgáltatásbusz) és a *szolgáltatástár* biztosítja, amelyek mindegyike fontos eleme a SOA-infrastruktúrának.

A SOA elvben lehetővé teszi, hogy a szoftvereket és komponenseiket szolgáltatásokként különböző gyártóktól, *különböző forrásokból* szerezzék be, hogy aztán ezeket a legkülönbözőbb módon kombinálva jól tudjanak igazodni a változó üzleti követelményekhez.

A gyakorlatban azonban meglehetősen nehéz a különböző forrásokból származó szolgáltatásokat integrált folyamatokba szervezni. Emiatt – jóllehet megvan a lehetőség a szolgáltatások többpartneres ellátására – az igazi üzleti érték mégis inkább a külső szolgáltatók által nyújtott új, innovatív funkciók *gyors beépítéséből* fog származni a vállalatirányítási alaprendszerek (például ERP) „határain”, míg maguk ezek az alaprendszerek akár továbbra is egy (vagy néhány) gyártótól származhatnak. Emellett azon külső szolgáltatók kínálata lesz előnyben, amelyek az alaprendszerek gyártóinak öko-rendszerébe, azaz partneri hálózatába tartoznak. A tanulmány áttekinti a szolgáltatás-orientált architektúrákat éppúgy, mint az ezekre épülő szolgáltatásalapú alkalmazásfejlesztést.

2. Jelenlegi helyzet

A vállalatok *agilitása* (reagáló képessége) nem pusztán a változtatások végrehajtásának gyorsaságától, hanem elsősorban az arra való felkészültségtől függ. Az agilitás az információs rendszereknél nem érhető el a hagyományos architektúrais elvekre épült rendszerek egyszerű felgyorsításával, hanem be kell építeni a rendszerarchitektúrába, és az ezt támogató és kezelő szervezetbe.

Manapság az ún. „szolgáltatások” válnak a szoftverarchitektúra, azaz a szoftverek komponensekre való felosztásának elsődleges egységeivé. Az egymáshoz nyílt és szabványos felületen kapcsolódó szolgáltatásokból a korábbinál jóval rugalmasabb architektúrával rendelkező szoftvereket lehet létrehozni. Ez a nyílt ún. *szolgáltatás-orientált architektúra* (SOA) nagyobb mértékű elmozdulást jelent a ma uralkodó, zárt szoftverarchitektúrákhoz képest, mint amilyen a kliens-szerver vagy a böngésző alapú architektúra-modellekre való áttérés volt korábban a nagyszámítógép-terminál alapú architektúrákról. Ugyanakkor a SOA jól beleillik a számítógépes rendszerarchitektúrák 1960-as években beindult fejlődési vonalába: a kezdeti, *kötegetelt feldolgozást* (batch processing) biztosító architektúrákat az 1970-es évektől kezdve az *online tranzakciófeldolgozásra* (On-Line Transaction Processing, OLTP) felkészített architektúrák váltották fel, később pedig az 1980-as és 1990-es években a kliens-szerver jellegű, két- majd háromrétegű architektúrák jelentek meg. A SOA a 2000-es évek válasza a rendszerarchitektúrákkal szemben felmerülő, üzleti indíttatású igényekre. A vállalatirányítás alaprendszereinél (például ERP) már legalább öt éve tapasztalható a fokozatos elmozdulás a SOA irányába, és ezért ez a terület meglehetősen érettnek tekinthető a SOA szempontjából. Az agilis üzleti szervezetek pedig ma már egyre inkább ún. interaktív⁴ SOA-t használnak, amelyet, ha szükséges, kiegészítenek eseménykezeléssel. A szolgáltatás-orientált és eseményvezérelt architektúrák (*Event-Driven Architecture*, EDA) a vállalati informatikát eredményesebbé és rugalmasabbá teszik, mint a szigetszerűen működő, monolitikus rendszerek.

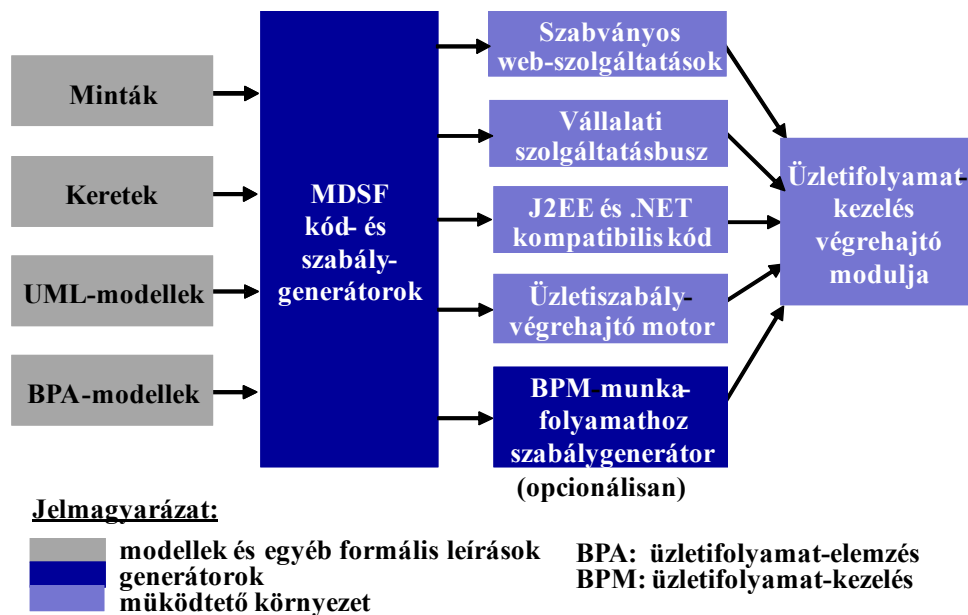
⁴ Nem az emberrel folytatott „párbeszédés üzemmód”, hanem az eredetibb „kölcönhatás” értelemben, azaz a SOA szolgáltatásai időlegesen – a meghívástól a szolgáltatásnyújtás befejezéséig – egymással kölcsönhatásba kerülnek, de a kölcsönhatás kontextusa ezt követően nem őrződik meg, nem befolyásolja a szolgáltatások további működését. Az „interaktív” vagy „kliens-szerver” SOA kifejezés az architektúra állapotmentes működésére utal. Ennek kialakulásában jelentős szerepet játszott a webszolgáltatások elterjedése.

2.1 Az alkalmazáskészítés új megközelítései

A SOA-elveket használva ma a szoftvercégek *szolgáltatásalapú alkalmazásokat* (service-oriented application) fejlesztenek ki ún. *szolgáltatás-orientált alkalmazásfejlesztési* (service-oriented application development) megközelítést használva.⁵

Ez az új megközelítés az alkalmazásokat lazán integrált (és jellemzően már kész) szoftvermodulokból (szolgáltatásokból) építi fel: az „először összeépíteni” elvet követi ellentétben a hagyományos szoftverfejlesztés „először programozni” (és csak azután összeépíteni) megközelítésével. Az ilyen fajta alkalmazásintegráció kulcsszerepet játszik mind a *szervezetten belüli* (hagyományos rendszerek újrafelhasználása), mind a *szervezetek közötti* (elektronikus kereskedelem, B2B) információkezelés összehangolásában.

Valójában az alkalmazásintegráció motivált számos korai SOA-megvalósítást is – azaz olyan elosztott, moduláris jellegű szoftverarchitektúrát, amely lehetővé teszi, hogy egységes felületű modulokat több alkalmazásból is használni lehessen. Sőt, az elmúlt időszakban az alkalmazásintegráció volt a szoftveripar területén az innováció fő hajtóereje, amelynek eredményképp a monolitikus, elszigetelt alkalmazások folyamatosan kiszorulnak a piacról, míg az új rendszerek modularizáltak, elosztottak, és már nemcsak integráltak, de úgy is tervezték őket, hogy integráltak legyenek.



2. ábra: Modellvezérelt szolgáltatásalapú fejlesztési keretrendszer fő elemei

Az alkalmazásintegrációra és az erős architektúratervezésre (*architected development*) alapuló megközelítések – mint amilyen a SOA is – azonban nem az egyedüliek az üzleti,

⁵ Az egyszerűség kedvéért a tanulmány a továbbiakban a „SOA” rövidítést egyaránt használja a *Service-Oriented Architecture*, a *Service-Oriented Application* és a *Service-Oriented Application Development* fogalmak jelölésére, ha a szövegkörnyezetből könnyen kideríthető, hogy melyikről is van szó.

közigazgatási, kormányzati illetve egyéb alkalmazások létrehozása területén. Fontos irányokat jelentenek még a gyorsfejlesztésre (*rapid application development*) és a modellezésre (*model-driven application development*) alapuló megközelítések is. Az eddigi tapasztalatok és vizsgálatok arra utalnak, hogy e négy fő megközelítést *következétesen együtt használva* a szolgáltatásalapú alkalmazásfejlesztés során, jelentős termelékenyséjavulás érhető el. Várható, hogy az ezt biztosító például ún. modellvezérelt szolgáltatásalapú fejlesztési keretrendszerek (*model-driven service frameworks*) akár 80%-át is meg fogják tudni generálni az alkalmazás programkódjának – drasztikusan csökkentve ezzel a programozáshoz szükséges időt és növelve reagáló képességet (agilitást) az üzleti változások megvalósításánál (ld. 2. ábra).

2.2 Hagyományos köztesszoftverek a SOA-hoz

Az alkalmazásintegráció és az ezt támogató SOA az (alkalmazások és az operációs rendszerek közötti) köztesszoftver-infrastruktúrán alapul. Mára az alkalmazásgyártók felismerték, hogy ez a köztes réteg⁶ kritikus a jövő alkalmazási rendszereiben.

A hagyományos köztesszoftverek azonban nem elegendőek olyan gyorsan terjedő alkalmazási architektúrákhoz, mint a szolgáltatás-orientált architektúra vagy az erre épülő, ezt kiegészítő *eseményvezérelt architektúra* (Event-Driven Architecture, EDA):

- A kapcsolat-orientált, hagyományos köztesszoftvernél – ideértve a *távoli eljáráshívást* (Remote Procedure Call, RPC) és *objektumkérés-kezelőket* (Object Request Broker, ORB) – a csatolás túl szoros (*tightly coupled*) a komponensek között, az aszinkron kommunikáció nincs jól támogatva, és a legnagyobb gyártók máig nem egyeztek meg a szabványokban.
- Az ún. *üzenet-orientált köztesszoftvereket* (message-oriented middleware, MOM) magasan kvalifikált fejlesztők ugyan képesek használni SOA- vagy EDA-célokra, de a *kapcsolati metaadatok* hiánya a fejlesztést rendkívül nehézkessé teszi, és a MOM-szabványok szintén nem arattak általános sikert.
- Végül a sima (architekturális támogatás nélkül használt) webszolgáltatások megfelelőek lehetnek egyszerű pont-pont SOA-alkalmazásra, de a hiányzó, egységes adatszállító és eseménykezelési képességeik következtében használatuk hosszabb távon kockázatos.

2.3 Vállalati szolgáltatásbuszok (ESB)

Meghatározó tendencia, hogy ún. *vállalati szolgáltatásbuszok* (Enterprise Service Bus, ESB) alakulnak ki, mint a köztesszoftverek új típusa, amelyek ötvözik a korábbi köztesszoftverekben található hasznos tulajdonságokat. Az ESB olyan webszolgáltatások működtetésére képes infrastruktúra, amely „intelligens” módon támogatja a kommunikáció irányítását és a kapcsolatok közvetítését a lazán csatolt (*loosely coupled*) vagy szétválasztott (*decoupled*) üzleti komponensek között. Ma több mint egy tucat gyártó (köztük a BEA Systems, a Microsoft és az IBM) szállít általános célú, kereskedelmi ESB-termékeket.

⁶ Sok piaci megfigyelőnek volt komoly fenntartása, amikor az IBM az 1990-es vége felé bejelentette, hogy felhagy az *alkalmazáscsomagok* (packaged application) készítésével és forgalmazásával, és ehelyett az informatika üzleti alkalmazását jóval hatékonyabbá tevő *köztesszoftverekre* (middleware) helyezi stratégiájában a hangsúlyt.

Az ESB-ek szolgáltatásregisztrációt és -felkutatást kínálnak, rugalmasságot az adatkommunikációban, és a szolgáltatásminőség magas szintjét. Az ESB-eket ki lehet terjeszteni teljes integrációs keretrendszerre is üzletifolyamat-kezeléssel, B2B-képességekkel és alkalmazási adapterekkel bővítve.

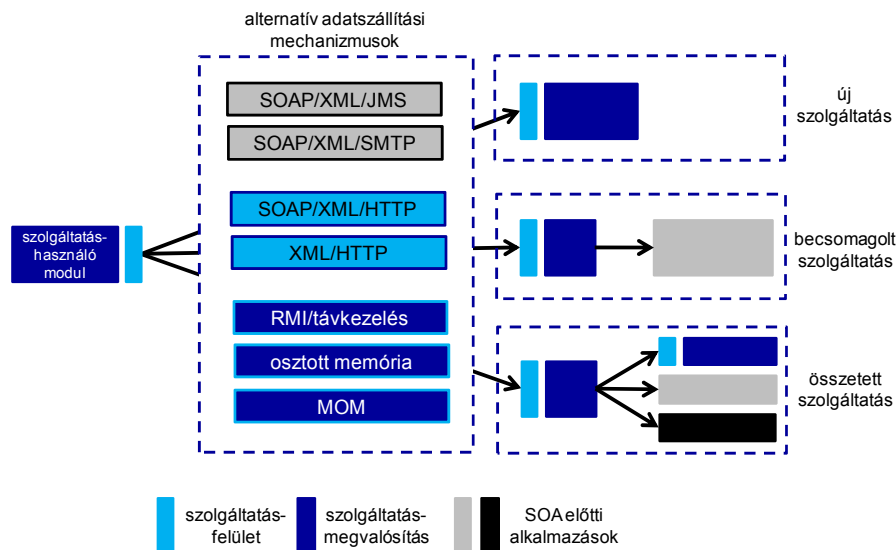
Történetileg visszatekintve a *Progress Software* volt az a cég, aki 2002-ben először használta az „ESB” megjelölést, és készítette el az első kereskedelmi forgalomban kapható ESB-terméket webszolgáltatás-támogatással (*Sonic XQ*, később: *Sonic ESB*). Kevés ESB-szerű, kereskedelmi köztesszoftver-termék létezett ezt megelőzően. Ilyen volt azonban a *Fiorano Tifosi* nevű terméke 1998-ban, amely 2007-re *Fiorano SOA Platform*-má fejlődött: olyan teljes értékű SOA-infrastruktúrává, amely webszolgáltatásokat kezel, valamint működéstervező és -vezénylő (orchestration), illetőleg egyéb korszerű funkcióval rendelkezik.

Szintén úttörő szerepet töltött be a *Candle* 1998-ban bejelentett *Roma* nevű terméke, amelyet később *Pathwai* néven forgalmaztak, amíg az IBM fel nem vásárolta a céget. A *Roma* megvalósította az ESB alapvető elveit (például SOA-kommunikáció egy üzenetkezelő alrendszer felett), de mivel tervezése a webszolgáltatási szabványok megjelenése előtti időre esett, ezért csak korlátozott mértékben fogadták el a későbbiekben.

Az 1990-es években számos nagyvállalat rendelkezett kellően jövőbe látó rendszertervezőkkel és a szükséges műszaki erőforrásokkal, hogy saját köztesszoftvert építsen kommunikációs és integrációs célokra. Ezek többnyire programkönyvtárak voltak, és magasszintű API-kként voltak használhatók valamilyen üzenetkezelő rendszer felett (*Message-Oriented Middleware*, MOM) – általában az *IBM Websphere MQ* (eredetileg: MQSeries), vagy időnként objektumkérés-kezelő rendszerek felett, mint például a *Iona Orbix*-a, esetleg tranzakciókezelő rendszerek felett, mint például a *BEA Tuxedo* terméke.

Mint a modern ESB-eszközök, ezek is „megszabadították” a SOA-alkalmazásokat a kommunikáció, a szabálykövetés, a címezés, a biztonság, a naplózás és a válasz-kérés összerendezés bizonyos vonatkozásainak kezelésétől. Mindazonáltal nem rendelkeztek formális *szolgáltatástárral* (service registry), integrációs képességekkel és metaadatkezelési lehetőségekkel sem.

Az XML 1998-as és a SOAP v1.1 (Simple Object Access Protocol) 2000-es megjelenése után ezeket az egyedi fejlesztésű alrendszerket kiterjesztették a webszolgáltatások szabványainak támogatására, és ezzel lényegében „házilag” készített ESB-ekké váltak.



3. ábra: Szolgáltatástípusok és adatszállítási mechanizmusok

Néhányan még ma is ezt az utat követik, de új, egyedi ESB létrehozása ma már nem nagyon praktikus: se megfelelő szakértelemmel, se hosszabb távú fejlesztési elkötelezettséggel, se olyan különleges üzleti követelményekkel nem rendelkeznek, amely lehetővé és szükségessé tenné egyedi ESB kifejlesztését. Ennek ellenére van egy számottevő kisebbsége a cégeknek, amelyek félig-meddig egyedi ESB-ket készít nyílt forráskódú komponenseket (például nyílt forrású MOM, SOAP-kezelő és transzformációs motor) jogvédett komponensekkel és – bizonyos fokig – egyedi kóddal vegyítve. Emiatt (is) az igény a nyílt forráskódú SOA-infrastruktúra iránt érezhetően növekszik.

2.4 Metaadatkezelés

A SOA-ra való áttérés folyamán az egyik legnagyobb kihívást az üzleti folyamatok és adatok *szemantikai egységesítése* jelenti, amelynek révén ezen az új szinten kell megvalósulnia a folyamatok közötti gyakorlati együttműködésnek.

Egyes korszerű SOA-k elkezdtek támogatni a központi, integrált *metaadattár* koncepcióját, ami az összes integrációval kapcsolatos metaadat egységes kezelését biztosítja. A metaadatok jelentős mértékben elősegíthetik az alkalmazások egységes működését. A metaadattár lehetővé teszi a működési szabályok megosztását is az alkalmazások között, és ezzel a szemantikai együttműködést is elősegítik.

A metaadattár fejlesztését és menedzsmentjét biztosító eszközök a SOA működésének tervezését és menedzsmentjét is javítják. A metaadatok minden új rétege növeli az agilitást is, mivel változtatások at és kiterjesztéseket a metaadatok szintjén gyorsan és alacsony kockázat mellett lehet megvalósítani.

A partneri együttműködésekben a pont-pont kapcsolatok egyre fokozódó elburjánzása már ma is jól érezhető „B2B-integrációs spagetti”-t eredményez. E kapcsolatokra vonatkozó (és természetesen sok egyéb más) metaadat hatékony és minél automatikusabb kezelése ez irányban is megoldást jelent.

3. Folyamatban lévő kutatások, fejlesztések

A SOA-ra való folyamatos áttérést több tényező is jelzi. A technológiai kísérletezésben *élenjáró cégek* sikeres, nagy volumenű bevezetésekről számolnak be. A *szoftvergyártók* rendkívüli mértékben növelik a befektetéseiket a SOA-jellegű köztesszoftverekbe, fejlesztő és menedzsmenteszközökbe. Az *alkalmazáscsomagok gyártói* alkalmazásaikat korszerűsítik vagy úgy, hogy azok képesek legyenek SOA-környezetben működni, vagy úgy, hogy újraírják azokat a valódi SOA-modellnek megfelelően. És egyre több vállalatnál formálódik olyan *kompetenciaközpont*, amelynek feladata a SOA-val és/vagy alkalmazásintegrációval kapcsolatos tevékenységek koordinálása.

3.1 ERP-rendszerek modernizálása

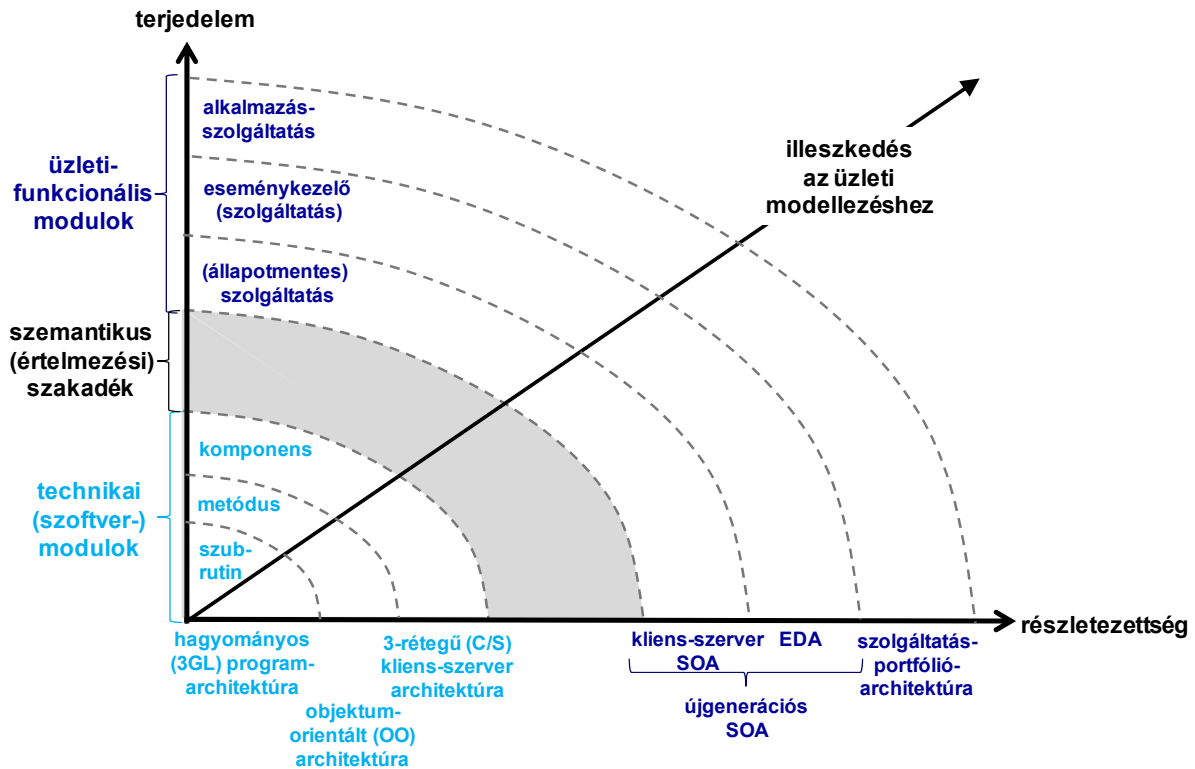
A nagy vállalatiszoftver-gyártók mindegyike néhány éve szinte egyszerre úgy döntött, hogy a SOA-t fogja a saját fejlesztési és partnerépítési megközelítésének alapjává tenni. Az IBM, a Microsoft, az Oracle, az SAP, a Fujitsu (a Software AG-val, mint partnerrel), a BEA, a Tibco, és sok más gyártó egyaránt abban reménykednek, hogy ilyen módon tudják egyszerre mind az ügyfeleik, mind a partnereik bizalmát megtartani.

3.2 Szolgáltatásvirtualizáció

Az olyan szoftvergyártók, amelyek a webszolgáltatások magasabb szintű követelményeit kielégítő technológiával is rendelkeznek, mint pl. a SOA Software (korábban: Digital Evolution) és a Blue Titan, kísérleteket folytatnak virtuális szolgáltatáshálózatok terén a webszolgáltatások minél megfelelőbb menedzsmentjének érdekében. Egyes ESB-gyártók (mint például a Sonic és a Tibco) olyan köztesszoftvert kínálnak, amely kezdeti lépést jelentenek a *szolgáltatásvirtualizáció* irányában.

3.3 Újgenerációs SOA

A SOA újabb generációja (*advanced SOA*) a kliens-szerver elvű (interaktív) SOA és az EDA szerves integrációjára fog épülni (ld. **Hiba! A hivatkozási forrás nem található.** ábra). Mint a múltban oly gyakran, az új architektúratípus (az eseményvezérelt SOA) az elődjének a kiegészítésével és nem a helyettesítésével fog kialakulni. Az EDA tökéletes kiegészítője az interaktív SOA-nak, és ez utóbbi egy lényeges kiegészítője az előbbinek. A szolgáltatások eseményeket váltanak ki, és az eseménykezelők szolgáltatásokat hívnak meg az események feldolgozására. Ahogy a SOA használata fejlődik és egyre érettebbé válik, az EDA követelményei egyre szélesebb körben merülnek fel.



4. ábra: A modularitás üzleti és technikai nézőpontjai

3.4 Egyéb innovatív SOA-technológiák

Az SOA-nak vannak egyéb – jelen pillanatban periférikusnak számító – technológiai és platformjai: bár kockázatosak, még nem teljesen megbízhatóak, de jelentős előnyökkel is járhatnak.

Idetartoznak például az *ágens-alapú integrációs* eszközök (Agent-Based Integration), a *szótár-* (pontosabban: ontológia-) *alapú transzformációs* eszközök (Vocabulary-Based Transformation) és az ún. *vállalati információintegráció* (Enterprise Information Integration, EII). Ezek az újonnan megjelent technológiák hatalmas lehetőségeket rejtnek magukban, és várhatóan itt fog a legtöbb innováció történni a szoftverinfrastruktúrák területén.

3.5 EU által támogatott projektek

Az EU számos projektet támogat a „szoftverek és szolgáltatások” (*software and services*) stratégiai témakörben, ahova a szolgáltatás-orientált megközelítések is tartoznak. Az ennek keretében végzett projektek célja, hogy

- egyrészt a SOA-ra épülő alkalmazások fejlesztéséhez és működtetéséhez szükséges eszközök lehetőleg *nyílt forráskódban* is rendelkezésre álljanak az EU-ban, például:
- módszerek, eszközök és technikák létrehozása *megbízható szolgáltatások* és szolgáltatásalapú alkalmazások költséghatékony fejlesztése és használata céljából

- intelligens keretrendszer kifejlesztése, amely lehetővé teszi, hogy nyílt és bővíthető fejlesztési platformokat hozhassanak létre *webszolgáltatás-alapú alkalmazások* céljából
- másrészt egy-két különösen innovatív megoldás (például szemantikus webszolgáltatás⁷, adaptív szolgáltatásbevonás, heterogén szolgáltatásintegráció) kikísérletezésére lehetőséget biztosítsanak az EU versenyképessége érdekében, például:
- *nyílt fejlesztési platform* kifejlesztése adaptív szolgáltatásmegtalálás, -létrehozás, -integráció és -bevezetés céljából
- szabványalapú, átfogó megoldás biztosítása a *heterogén szolgáltatások integrációjára* modellek, nyelvek és nyílt forrás kódú közteszoftverek kifejlesztésével.

4. A várható fejlődés

A várakozások szerint a szolgáltatás-orientált architektúrák és ezekre épülő szolgáltatásalapú alkalmazáskészítési (tulajdonképpen alkalmazásintegrációs) módszerek néhány éven belül már jelentős hatás váltanak ki, míg a következő évtized végére meghatározóvá válnak.



5. ábra: Várható fejlődés (2008-2018)

4.1 Alkalmazáscsomagok SOA-alapon

Az ERP-rendszer már ma sem csak egy egyszerű tranzakciós rendszer, amely a pénzügyeket kezeli, hanem a vállalat tevékenységének a gerincévé vált. Az ERP egyre kiterjedő, folyamatokra épülő szerepének támogatása érdekében az IT-modernizációs projektek egyik legfontosabb célja a SOA-technológia megvalósítása.

⁷ További információért vö. *Jelentésalapú technológiák* elemzés.

Néhány éven belül az új alkalmazási szoftverekből származó bevételek nagy része SOA-t használó szoftvertermékekből fog realizálódni akár hagyományos licenc-, akár előfizetési díjak formájában. Emellett a szoftverintegrátorok és a szoftvergyártók közötti megkülönböztetés egyre inkább elmosódik, ahogy az alkalmazáscsomagokat (például ERP) részekre bontják, és azokat SOA-elemekként szállítják.

A SOA elterjedése az alkalmazáscsomagok licencvásárlása helyett a szolgáltatások *előfizetése* felé fogja eltolni a bevételeket, valamint a monolitikus szoftvercsaládoktól az *összetett alkalmazások* – azaz több, különböző szolgáltatásból összeépített alkalmazások – irányába.

4.2 Vállalati szolgáltatásbuszok

Az *adatszállítási mechanizmusok* (data transport) a szoftverprojektek köztesszoftver-megoldásainak stratégiai elemévé válik, és hosszabb távon is megteremti az összhangot a szolgáltatások és az események között. A hagyományos – az üzenetek sorbaállítására épülő – köztesszoftvereket ki fogja váltani a *vállalati szolgáltatásbusz* technológiája. Az ESB az adatszállítást magasabb szintre emeli. Az új ESB-gerinc, amely az integrációs termékek új generációjának is utat nyit, radikális javulást hoz a legtöbb szervezet szoftverinfrastruktúrájába.

Nemcsak egyszerűen javítják a szervezetek belső rendszerei közötti információáramlás menedzselhetőségét és teljesítményét, az új ESB-alapú platformok támogatást adnak olyan alapvető fontosságú alkalmazási irányokhoz is, mint pl.

- a többprotokollós szolgáltatás-orientált architektúra (túllépve a webszolgáltatások szolgáltatásminőségi korlátain);
- esemény-vezérelt architektúra;
- koordinált rendszermenedzsment (*coordinated system management*);
- üzleti tevékenység monitorozása (*business activity monitoring*).

4.3 Web 2.0 és SOA együttes alkalmazása

A SOA-ra épülő ERP alapját képezheti a web 2.0-ás innovációk vállalati támogatásának, mint pl. közösségi szoftverek vagy a mobilszámítástechnika és a vállalati montázsalkalmazások (mashup) széleskörű integrációja. Ezen technológiák alkalmazása fontos kiegészítő szerepet fog betölteni a következő években az üzleti teljesítmény növelésében.

Emellett a weben megjelenő ingyenes (de megbízható), illetőleg térítés mellett és szolgáltatásszerűen nyújtott webszolgáltatások jól be fognak épülni, és ki fogják egészíteni a SOA-alapú üzleti alkalmazásokat.

4.4 Metaadatkezelés SOA-ban

A *metaadatkezelés* fejlettsége fogja megkülönböztetni a piaci szereplőket az alkalmazásintegráció tekintetében. Ahogy az integráció egyre fontosabbá válik a tipikus vállalati-intézményi információs rendszerek napi működéséhez, és ahogy az integrációs termékek beépülnek a legtöbb szoftverfejlesztési projektbe, egyre lényegesebb lesz az a képesség, hogy az integrációs eljárásokat változásait következetesen, termelékenyen és integráltan lehessen kézben tartani.

A *metaadattárak* és az ezek körüli eszközök fontos, megkülönböztető tulajdonságai lesznek a vezető platform- és integrációs termékeknek.

4.5 Új tendenciák az alkalmazásintegrációban

A vállalati szolgáltatásbuszok megjelenése, a *B2B-átjárók* (B2B-gateway) növekvő száma és a külső integrációs szolgáltatók piaci jelentkezése csak a legismertebb példái az alkalmazásintegráció területén történő jelentős változásoknak. Az *alkalmazásintegráció* továbbra is meg fogja határozni az innováció menetrendjét, és vezérli az alkalmazási infrastruktúrák és a köztesszoftver-termékek fejlődését.

A soktermékes gyártók valószínűleg kiszorítják az integráció-specialistákat a piac vezető pozícióiról – legalábbis a licenchevételek tekintetében. Ugyanakkor az alkalmazásintegráció piaca továbbra is széttagolt marad: egyetlen gyártó sem lesz képes meghatározó piaci részesedést elérni.

4.6 Üzletkritikus alkalmazások SOA-alapon

Öt év múlva a valós idejű, eseményvezérelt és szolgáltatás-orientált vállalati infrastruktúra nemcsak lehetőség lesz, hanem elemi versenykövetelmény is a legtöbb iparágban és vállalatnál. Az üzletkritikus alkalmazásokat emiatt egyre inkább *összetett, szolgáltatás-orientált architektúrára* építik, és a számuk megháromszorozódik a 2000-es évtized első feléhez képest.

A *szervezetközi hálózatok* külön – adott célt szolgáló – átjárókat és új *értéknövelt hálózatokat* (value-added network, VAN) fognak használni. A VAN-kínálat, mint a webszolgáltatások ipari méretű használatának technológiai platformja az üzletkritikus, szervezetközi környezetekben, a piaci növekedés új szakaszába lép.

4.7 Üzleti folyamatok és SOA integrációja

Az *üzletifolyamat-kezelő eszközök* (Business Process Management Suite, BPMS) ERP-be történő beépülésével az ERP-modernizációs projektek lehetővé fogják tenni a felhasználók számára, hogy *grafikus üzletifolyamat-modelleken* – egyfajta modell-vezérelt alkalmazáson – keresztül változtassanak az ERP-rendszeren, és így jóval közvetlenebb ellenőrzést és befolyást biztosít az üzleti alkalmazások felett, valamint kevesebb hibázásra ad lehetőséget az üzleti igények rendszertervekre való lefordítás során. A folyamatok leírása (tulajdonképpen metaadataik) és megvalósításuk közötti határ egyre inkább elmosódik, ahogy a SOA-fejlesztési és -működési környezetek egységes platformmá integrálódnak, és egyszerre több informatikai és üzleti szerepkört is támogatni képes, hogy szorosan együttműködhessenek a kívánt megoldás létrehozásán. Ennek eredményeképp az *alkalmazások következő generációját* olyan szolgáltatásokból fogják előállítani, amelyek az üzleti folyamatok különböző változataira jól konfigurálhatók. A tranzakciós alkalmazások jelenlegi generációjától eltérően, amelyek a munkafolyamatok szélesebb kontextusán belül lényegében az ismétlődő, mechanikus feladatokat automatizálták, a következő generációs alkalmazások egyre inkább hasonlítanak azokra a teljes folyamat(ok)ra, amely(ek)et megvalósítanak, és ugyanúgy kiterjednek az ember által végzett tevékenységekre (mint például jóváhagyás, döntés, tárgyalás, elemzés, kutatás és együttműködés), mint az automatizált feladatokra.

4.8 Szolgáltatásvirtualizáció elterjedése

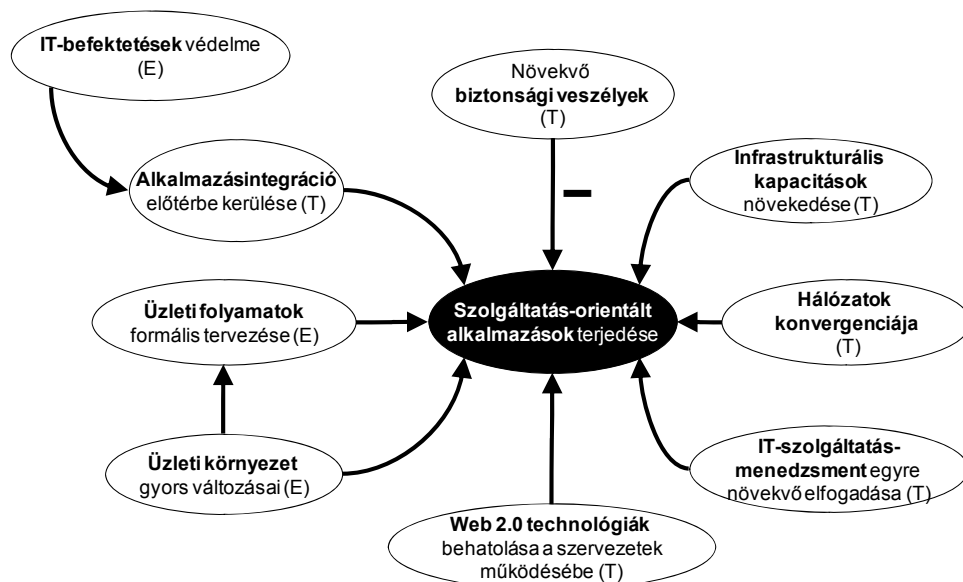
A *szolgáltatásvirtualizáció* lesz a következő logikus lépés a szolgáltatás-orientált és az esemény-vezérelt architektúrák továbbfejlesztésében. A jövő szoftverarchitektúrái meg

fogják követelni, hogy a szolgáltatáskérések megválaszolása a mindenkori legkényelmesebben és leghatékonyabban használható hardvererőforrásokon legyen végrehajtva.

Ezt biztosítja a szolgáltatások virtualizációja (*service virtualisation*), amely egy újabb absztrakciós szint megjelenését jelenti, és amely elválasztja a szolgáltatási funkció kérését egy adott szolgáltatás meghívásától. Ez a virtuális szolgáltatáshozzáférés lehetővé teszi – mint a virtualizált, grid-alapú hardvermegoldásoknál –, hogy az erőforrásokat akkor és úgy lehessen igénybe venni, ahogy arra a szükség van: egy mögöttes virtualizációs mechanizmus határozza meg, hogy mely szolgáltatások – vagy bizonyos típusú szolgáltatásból hány – fog válaszolni egy adott kérésre.⁸

A szolgáltatások ilyen virtuális hálózatai (*virtual networks of services*) néhány éven belül valósággá válnak, jóllehet még további öt évet fog igényelni ennek az architektúrális megközelítésnek az olyan mértékű kiérlelődése a gyakorlatban is, amikor már a virtualizáció a szolgáltatások gridjéhez való hozzáférést megszokottabbá teszi, mint az egyedi szolgáltatásokhoz való hozzáférést.

5. Befolyásoló tényezők



6. ábra: Befolyásoló tényezők

5.1 Technológiai tényezők

A SOA a mainál jóval *nagyobb kapacitásokat* igényel az infrastruktúra terén: az egyes előrejelzések szerint ötévente megtízszereződő infrastrukturális kapacitásokra szükség is lesz – elsősorban teljesítmény, megbízhatóság és ellenálló képesség terén.

⁸ Hasonlítható ez a *tárolóhálózatok* (storage area network, *SAN*) működési elvéhez is, ahol a tárolóberendezéseket hálózatra kötik, és a SAN határozza meg azt a berendezést, ami válaszolni fog egy adott tárolási kérésre. Tárolóberendezéseket el lehet távolítani vagy ki lehet cserélni anélkül, hogy az operációs rendszerek vagy az alkalmazási rendszerek ezt észre vennék.

A SOA megvalósulásához ezért elengedhetetlen, hogy az infrastruktúra (hardver, alap-, köztes- és menedzsmentszoftver) területén a szükséges kapacitásnövekedés (és az ezt lehetővé tevő technológiafejlődés) bekövetkezzen méghozzá olyan áron, amely a SOA-t továbbra is gazdaságilag életképes alternatívaként állíthassa a vállalatok elé.

A SOA ugyanakkor hallgatólagosan feltételezi, hogy a távközlés területén (gyakorlatilag) *korlátlan sávszélesség* lesz elérhető, és megvalósul a (fizikai) *hálózatok konvergenciája*. Ilyen módon egy-egy szervezet fizikai (és jogi) határain kívül lévő szoftverkomponensek valós idejű felhasználása, illetve beintegrálása nem ütközik majd műszaki korlátokba.

Negatívan hatást jelenthet viszont, ha a felmerülő *biztonsági kérdések* nem lesznek folyamatosan és kielégítő szinten kezelve. A SOA és az igényelt sávszélesség növeli a kockázatát az informatikai jellegű támadásoknak, mivel megsokszorozza az *automatikus* (közvetlen emberi kezdeményezés nélkül) létesülő, magas szintű, hálózati kapcsolatokat. E kapcsolatoknál a hitelesség, a letagadhatatlanság és a számonkérhetőség biztosítása a mainál fejlettebb információvédelmi megoldásokat igényel.

A *web 2.0 technológiai* szinte észrevétlenül hatolnak be a vállalatok működésébe. Az alkalmazottak személyes webhasználatára ma már gyakran hatékonyabb és korszerűbb, mint a vállalati rendszereké. Az így megszerzett gyakorlat és ismeretek természetes módon jelennek meg elvárásként a vállalati alkalmazásokkal szemben: a SOA-alkalmazások inkább állnak összhangban a web 2.0 lehetőségeivel, mint a hagyományos vállalatirányítási rendszerek.

Végül nem szabad elfelejtkezni arról sem, hogy az IT-szervezetek menedzsmentje észrevehető átalakuláson ment át az elmúlt öt évben. Elterjedt és egyre elfogadottabb lesz az *IT-szolgáltatásmenedzsment* bevált gyakorlata (például ITIL v2/v3 vagy COBIT), amely olyan *szolgáltatásarchitektúrát* javasol megvalósítani, ahol a szolgáltatások már nem csak az alkalmazásarchitektúra komponenseit jelentik, hanem magukba foglalják az alkalmazások fejlesztéséért, bevezetéséért, üzemeltetéséért és továbbfejlesztéséért felelős tevékenységeket is. Ilyen módon az IT-szolgáltatásmenedzsment ideális környezetet teremt a SOA-fejlesztési projektek megvalósításához és a szolgáltatásminőségi garanciákat nyújtani képes, SOA-üzemeltető tevékenységekhez.

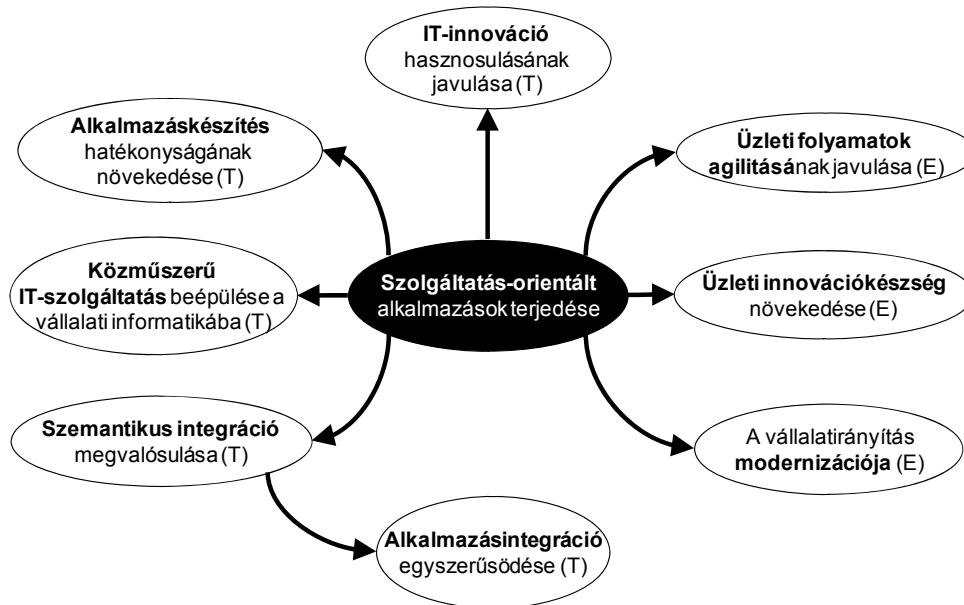
5.2 Társadalmi-gazdasági tényezők

A vállalatok gazdasági tevékenységében alapvetően az *üzleti környezet váratlan és gyors változásai* adják a legnagyobb hajtóerőt a SOA terjedéséhez. A hagyományos architektúrára épülő vállalati rendszereket ugyanis nem lehet elég gyorsan és kellő biztonsággal úgy módosítani, hogy a változó üzleti követelményeknek folyamatosan meg tudjanak felelni. SOA-val viszont üzleti funkciókat ellátó szolgáltatások rugalmas, könnyen változtatható rendszerét lehet létrehozni. Külön előnyt jelent, hogy segítségével az új üzleti igényekre kifejlesztett innovatív megoldásokat a szokásosnál könnyebben be lehet építeni a vállalati alkalmazásokba.

Az üzleti környezet gyakori változásai sok vállalatnál életre segítik az üzleti folyamatok tervezési-modellezési tevékenységét is. Az *üzleti folyamatok formális tervezése* elősegíti, hogy a szolgáltatás-orientált architektúrára épülő alkalmazások a lehető legoptimálisabban támogassák a folyamatokat, és a folyamatok változását a szolgáltatásalapú alkalmazásokkal gyorsan követni lehessen, mert az üzleti modellekből a szükséges szolgáltatásokat automatikusan meg lehet generálni, és csak azokat a szolgáltatásokat kell lecserélni, amelyek érintettek az üzleti változással.

A szervezetek vezetése gyakran úgy tekint az *IT-befektetésekre*, mint átláthatatlan és nehezen – ha egyáltalán – megtérülő, pénzügyileg kockázatos vállalkozásokra. Nem véletlen, hogy a végsőkig igyekeznek védeni a befektetéseket az informatika területén. Az *alkalmazásintegráció* éppen ezért került az elmúlt évtizedben az érdeklődés előterébe, hiszen ilyen módon nem kell feltétlenül teljesen új alkalmazást készíteni, hanem ki lehet indulni a már elkészült és jól működő alkalmazásokból, és nagyrészt meg lehet őrizni azokat. Ezt az alkalmazáskészítési megközelítést, és így az IT-befektetések hatásos védelmét, a SOA-technológiákkal lehet leginkább megvalósítani.

6. Várható hatások



7. ábra: A SOA várható hatásai

6.1 Technológiai hatások

A SOA egyik kedvező hatása, hogy elősegíti egy sor *IT-iparban keletkező innováció* megvalósulását. A SOA első hulláma még azzal a feltételezéssel élt, hogy a szolgáltatáshasználó és -nyújtó modulok majdnem kizárólagosan a hagyományos, távoli eljáráshívásra (remote procedure call) emlékeztető, kérés-válasz jellegű, kommunikációs szabályokat alkalmazhatnak. Ahogy a SOA-fejlesztők újabb és újabb üzleti követelményekkel találkoztak, felismerték, hogy bizonyos helyzetekben egy teljesen aszinkron, eseményvezérelt architektúrára (EDA) van szükség. Ennek hatására a SOA fogalma fokozatosan kibővül az EDA tulajdonságaival (újgenerációs SOA). Az újgenerációs SOA pedig előkészíti a terepet egy másik innováció széles körben való elterjedésének – az egyre komplexebb ún. kontextusérzékeny alkalmazásoknak.⁹ A SOA más olyan IT-trendek kibontakozását is segíteni fogja, amelyek a SOA-alkalmazások

⁹ Az ilyen alkalmazások a kontextusinformációk alapján automatikusan módosítják a működésüket. Gyakran egy konkrét ügyfélre vagy felhasználóra és adott szituációra vannak szabva. Az üzleti alkalmazások régóta gyűjtenek és használnak kontextusinformációkat, de ennek rendszerszerűségét és elosztott jellegét a SOA jelentősen növelni tudja.

moduláris és elosztott jellegét használják ki, mint például a virtuális számítástechnika vagy a modell-vezérelt szoftverfejlesztés nem tudna (vagy nem tudna olyan gyorsan) kialakulni a SOA jótékony hatása nélkül.

Egy másik, figyelemre méltó kapcsolat a jelentésalapú technológiákkal áll fenn. A *szemantikus integráció* egyes komponensek vagy információforrások magas szintű, üzleti fogalmak szintjén létrejövő összekapcsolását ígéri, amelyben kiemelt szerepe van e komponensekről, illetve információforrásokról igen szerteágazó, leíró információkat tárolni és kezelni képes metaadattáraknak. Ez jelentős hatással lesz az alkalmazásintegrációra annak későbbi szakaszában (várhatóan csak az évtized közepén), amikor az alapvető architektúráis változások már lezajlottak, azaz metaadattárakkal felszerelt, szolgáltatás-orientált architektúrák elterjedtek. Az összetett alkalmazások komponensei közötti együttműködés megteremtése ezzel lényegesen leegyszerűsödik, ugyanakkor hatékonysága és eredményessége növekszik.

A SOA elősegíti, hogy a *közműszerű IT-szolgáltatások* a komplex alkalmazási szoftverek területére is kiterjedhessenek. Ehhez az kell, hogy az alkalmazásokat külön-külön – és távolról is – jól használható komponensekre (szolgáltatásokra) bontsák fel, és így ezek SOA-kompatibilisek legyenek. Másrészt arra is pozitív hatással lesz, hogy az ilyen módon egyre növekvő számú szolgáltatás gridbe – tulajdonképpen virtuális hálózatba – szervezhetőek legyenek.

Az egyediség erőltetése ott, ahol már bevált típusmegoldások vannak, az erőforrások pazarlását, végső soron alacsony szintű hatékonyságot von maga után. Az alkalmazásfejlesztés szolgáltatás-orientált átalakulása nagyban elősegítheti, hogy az egyedi *alkalmazáskészítés hatékonysági problémái* megoldódjanak. A SOA kisebb mértékű ráfordítást, illetőleg a ráfordítások hatékonyabb felhasználását ígéri. A helyzetet tovább élezi, hogy az egyedi alkalmazáskészítési tevékenység ma még gyakran nem az innovatív üzleti megoldások területén jelentkezik, azaz nem ott, ahol ténylegesen szükség lenne egyedi megközelítésekre. A SOA-ra épülő alkalmazáskészítés elősegíti az innovációkészséget az üzleti folyamatok informatikai támogatásában is.

6.2 Társadalmi-gazdasági hatások

A SOA legjelentősebb hatása, hogy elősegíti az *üzleti innovációt* termék-, szolgáltatáskínálat, illetőleg üzleti gyakorlat tekintetében. Az üzleti vezetők sokkal gyorsabban tudnak létrehozni vagy módosítani egy SOA-alapú üzleti folyamatot, mint egy olyat, amelyet nem SOA-alapú rendszerek támogatnak – ha egyébként azok egyformák. A SOA teszi lehetővé, hogy egy cég úgy módosíthassa üzleti architektúráját, hogy gyorsabban reagáljon az ügyféligenyekben vagy – szélesebb értelemben véve – a piacon bekövetkező változásokra.

A SOA-ra való áttérés növeli az *agilitást* az üzleti folyamatokban. Az üzleti vezetőknek tisztában kell lenniük azzal, hogy azok a vállalatok, amelyek a következő öt évben megfelelő modell-vezérelt alkalmazásokat alakítanak ki, rugalmasabb üzleti folyamatokkal fognak rendelkezni, a folyamatok gyorsabb megváltoztatásra lesznek képesek és az üzleti tevékenységük összehangoltabb lesz az informatikával.

A nagyvállalatok irányítása jelenleg olyan alkalmazáscsomagok „kezeiben” van, amelyek nehezen változtathatók (ha igen, akkor is csak a gyártók által), csak korlátozott mértékben paraméterezhetőek és gyakorlatilag nem lecserélhetőek. Legfőbb előnyük, hogy beválnak tekintett üzleti gyakorlatot közvetítenek „kevésbé fejlett” vállalatok felé,

ugyanakkor nincs garancia arra, hogy az egyik szervezetnél bevált gyakorlat nem lesz éppen ellenkező hatással egy másik helyen. Ezt ma már e szoftverek gyártói is felismerték, és ezért nem véletlen, hogy alkalmazáscsomagjaikat újabban SOA-alapokra helyezik. Ilyen módon a SOA-alapú vállalati alkalmazások a *vállalatirányítás modernizációjában* élenjáró szerepet fognak betölteni: növelik a vállalatvezetés cselekvési lehetőségeit, fokozzák a reagáló képességét és elősegítik az IT-befektetések racionalizálását.

7. Hazai helyzet

7.1 Jelenlegi helyzet

A vezető magyar alkalmazásfejlesztő cégek (például IQSYS, Fornax, Alerant) szakemberei már a 2000-es évek elején megismerkedtek az *alkalmazásintegráció* alapeszközeivel (például BEA WebLogic, IBM WebSphere), és azóta több sikeres projektben is használták azokat – elsősorban a pénzügyi területen.

Magyarországon azonban az *egyedi alkalmazásfejlesztés* „hagyományosan” aránytalanul nagy piaci részesedéssel rendelkezik összehasonlítva a vezető nyugati országokkal.

Ennek oka a hazai munkaerő relatíve alacsony költsége mellett elsősorban abban rejlik, hogy a hazai vállalatok és intézmények vezetőiben erős a tendencia, hogy saját szervezetüket és annak működését egyedinek, megismételhetetlennek tekintsék.

Nem véletlen az sem, hogy a *vállalatirányítás* szabványos informatikai alaprendszereinek terjedésében elsősorban nem a hazai tulajdonban lévő, hanem a multinacionális vállalatok képviselték a húzóerőt. Ma – amikor ez a piac a telítődés jegyeit mutatja – még kérdéses, hogy a hazai kis és közepes vállalatok képesek lesznek-e típusalkalmazásokat bevezetve korszerűsíteni gazdálkodásukat.

Talán mindezek együttes „eredményeként” is a hazai alkalmazásintegrációs projektek inkább alkalmazások közötti egyedi kapcsolatokat valósítanak meg, mintsem hogy hosszabb távú architektúrais megoldásokkal kísérleteznének. Vezető informatikusok még néhány évvel ezelőtt is igen szkeptikusan nyilatkoztak a SOA hazai megvalósítási lehetőségeiről. Mára csak annyiban változott helyzet, hogy a szoftvergyártók SOA-alapú termékfejlesztései kezdenek az IT-szervezetek látókörébe kerülni, és helyenként használatba is veszik azokat.

A *SOA-ra való áttérés* tekintetében sem jobb a helyzet, mivel itt a vállalati IT-architektúra jelentős átalakításáról van szó, az üzleti modell kevésbé nyilvánvalóan előnyös, különösen nem rövidtávon, és különösen nem a webszolgáltatásoknak a vállalati informatika fő komponenseiként való elfogadása területén. Jelenleg elsősorban csak kísérleti jelleggel próbálkoznak magyar szervezetek kívülről származó webszolgáltatások felhasználásával. Jóval többen tervezik a weben megjelenő szolgáltatásaik webszolgáltatássá való átalakítását (pl. elektronikus áruházak, könyvesboltok), azonban ez még messze van a SOA és az ESB léptékű váltástól.

Megállapítható az is, hogy az *üzleti folyamatok* területén számos vállalat használ modellező eszközt a folyamatok tervezésére és vizsgálatára. Ezek azonban általában nem kapcsolódnak közvetlenül alkalmazásintegrációs megoldásokhoz.

7.2 Folyamatban lévő kutatások, fejlesztések

A SOA az informatika fő fejlődési irányvonalához tartozik, ahol az élenjáró szoftvergyártók jelentős stratégiai befektetéseket tesznek, ezért nem meglepő, hogy

Magyarországon e tekintetben sem folyik konkrét termékfejlesztés. A hazai szoftver- és rendszerintegrációs cégek elsősorban a kedvező piaci perspektívával rendelkező, külföldi eszközök hazai forgalmazásával és üzleti projektekből történő felhasználásával foglalkoznak.

Ennek ellenére több példa is van magyar szervezetek olyan hazai, illetőleg nemzetközi K+F projektekből való részvételére, amelyek a SOA egyes részterületeire irányulnak. Az **IQSYS** (korábban az IQSOFT) a szemantikus információintegráció és metaadattárkezelés szűkebb területén törekszik olyan élenjáró technológia kialakítására és ehhez kapcsolódóan technológiai ismeretek megszerzésére, amelyek felhasználásával javíthatja versenyképességét a hazai integrációs megoldások piacán.

A BME részvételével folyó **FUSION EU**-projekt pedig az üzleti folyamatok szemantikai szintű modellezésével igyekszik a webszolgáltatások integrációjára gyakorlatban is használható eszközöket kifejleszteni. A téma piaci fontosságára utal, hogy a projekt vezetését és koordinálását az SAP látja el. A projekt jól beleilleszkedik az SAP kutatási céljaihoz, amelyet Rainer Ruggaber (SAP Research) a „szolgáltatások internet”-jének (Internet of Services) nevezett komplex szolgáltatási infrastruktúráként fogalmazott meg, és amelynek építőelemei a SOA, a web 2.0 és a szemantika a technológiai oldalon, valamint üzleti modellek és szisztematikus, közösségre alapuló innovációs megközelítések a használati oldalon.

7.3 Várható fejlődés

Várható, hogy a vonzó üzleti modell (létező alkalmazások újrafelhasználása), az integráció eszközkészletének minőségi javulása és a bízható tapasztalatok együttesen az *alkalmazásintegrációt* egyre gyakrabban használt megközelítéssé teszik a jelenleginél jóval szélesebb alkalmazási területen (például távközlés, gyártó ipar) a nagyvállalatok körében.

A hazai kis és középvállalkozások azonban döntően a web 2.0 webszolgáltatásait, szoftverintegrációs megoldásait (mashup), nyílt forráskódú eszközeit és szemantikai megközelítéseit fogják használni rugalmas és gyorsan reagáló IT-támogatás megszervezésére üzleti tevékenységükhöz. Méreteikből és piaci helyzetükből adódóan ez a természetesen kialakuló, ad hoc irányvonal teljes mértékben le tudja fedni a piacaik által megkövetelt agilitást.

A valódi SOA terén azonban csak az évtized második felére várható, hogy néhány hazai, tőkeerős és erős piaci versenyre kényszerülő vállalat érett SOA-termékekre alapozva elkezd fokozatosan átalakítani az üzleti alkalmazásaik alatti szoftverarchitektúrát a SOA következetes és üzleti indíttatású megvalósítása érdekében. Náluk a web 2.0 technológiai csak kiegészítő jelleggel jelennek meg. Emellett ezeknél a szervezeteknél a SOA-megközelítés párosulni fog az IT-szolgáltatásmenedzsment élenjáró alkalmazásával is. Mindkét hazai vállalat kategóriában tipikus lesz a közműszerűen nyújtott szoftverszolgáltatások beépítése mind a web 2.0, mind a SOA környezetben. Sőt, jelentős eltolódás várható a saját, belső infrastruktúrán, illetőleg a külső szolgáltatók távoli adatközpontjaiban üzemeltetett szolgáltatások egymáshoz viszonyított arányában – az utóbbi megközelítés javára.

8. Összefoglalás

Az alkalmazáskészítés és -integrálás egyre szorosabban összefonódik a szoftverarchitektúrákkal és az ezeken működő szoftverekhez való hozzáférés szolgáltatásszerű és szolgáltatásként történő biztosításával. A készítendő szoftverrel szemben nemcsak az az elvárás, hogy szolgáltatásszerűen legyen működtethető („*a szoftver, mint szolgáltatás*”), hanem hogy kifejezetten úgy legyen elkészítve, amely a távolról való – mind emberi, mind gépi úton történő – használatát biztosítja (*a szoftver, mint egy szolgáltatás-orientált architektúra komponense*).

A „semmitől egy új” alkalmazást készítő programozó helyét fokozatosan a létező alkalmazásokat mindenféle más célokra is felhasználni tudó *integrációs bróker* tevékenysége veszi át, aki jól meg tudja érteni egyrészt az üzleti igényeket, másrészt hatékonyan képes az adott igényekhez alkalmazható, már létező (külső vagy belső) komponenseket megtalálni és ezeket összeépíteni.

Amennyire „védi” azonban a létező alkalmazásokat az alkalmazásintegráció, legalább annyira fogja felforgatni a szervezetek informatikai infrastruktúráját és szoftverarchitektúráját a szabványos webszolgáltatásokra építő szolgáltatás-orientált architektúra és az ezt támogató vállalati szoftverbusz, mint korszerű köztesszoftver-megoldás. A SOA egy folyamat kezdete és nem a befejezése, amely megadja a keretet további innovatív megoldások számára (például eseményvezérelt architektúra, kontextusérzékeny alkalmazás).

A ma még (kívülről) csak paraméterezzhető alkalmazási csomagokat külön-külön is jól integrálható, szabványos szolgáltatáskomponensekre bontják, és így is forgalmazzák őket. Egyre egyértelműbben jelenik meg az igény a ma még elkülönülő alkalmazások és információforrások *szemantikai szinten* is megvalósítandó együttműködésére, amelyben kiemelt szerepet kapnak a szabványos ontológiákat is kezelni tudó *metaadattárak*.

A hazai közép vállalkozások versenyképességének biztosításában az általuk alkalmazott informatikai megoldásoknak is növekvő szerepe lesz. Számukra különösen fontos, hogy a legkorszerűbb technológiákat hazai IT-szolgáltatók is képesek legyenek megbízhatóan és költséghatékonyan szállítani.

Félő azonban, hogy a szoftverarchitektúrák és alkalmazáskészítési megközelítések területén bekövetkező változások könnyen olyan helyzetet teremthetnek, amelyben versenyképességük (például információellátottság, üzleti rugalmasság, partnerkapcsolat-kezelési képesség) megtartása már csak előrelátó és következetes IT-beruházási stratégiával lenne elérhető, ugyanakkor az ehhez szükséges finanszírozási és szolgáltatói háttér csak részben lesz biztosítható.

Ajánlott irodalom

- Barry, Douglas K.: *Web Services and Service-Oriented Architecture: The Savvy Manager's Guide*. Morgan Kaufmann, 2003.
- Durvasula, Surekha et al.: *SOA Practitioners' Guide Part 2: SOA Reference Architecture*. September 2006. (<http://www.soablueprint.com/whitepapers/SOAPGPart2.pdf>)
- Hurwitz, J. – Bloor, Robin – Baroudi, C. – Kaufman, N.: *Service Oriented Architecture For Dummies*. Wiley Publishing, 2007.
- Natis, Yefim V.: *Applied SOA: Transforming Fundamental Principles Into Best Practices*. April 2007, Gartner.

- *Reference Architecture for Service-Oriented Architecture Version 1.0*. OASIS, April 2008. (<http://docs.oasis-open.org/soa-rm/soa-ra/v1.0/soa-ra-pr-01.pdf>)
- Christoph Schroth, Christoph – Janner, Till: *Web 2.0 and SOA: Converging Concepts Enabling the Internet of Services*. June 2007, IT Pro, IEEE Computer Society. (<http://www.alexandria.unisg.ch/EXPORT/DL/42537.pdf>)